POLITECNICO DI MILANO

Scuola di Ingegneria Industriale e dell’Informazione

Corso di Laurea Magistrale in Ingegneria Biomedica

A modified Faster R-CNN for pulmonary nodule detection: preliminary investigation of hyperparameters and network structure

Relatore: Prof. Guido Baroni

Correlatori: Ing. Noemi Garau
Dott. Ing. Chiara Paganelli

Tesi di Laurea Magistrale di:
Lorenzo Marelli
Matricola: 904968

Anno Accademico 2019 – 2020

Index

Figure index

Table index

List of abbreviations

CAD: Computer Aided Detection

CNN: Convolutional Neural Network

COSMOS: Continuous Observation of Smoking Subjects

CT: Computed Tomography

FROC: Free-Response ROC

Fuzzy K-NN: fuzzy K-Nearest Neighbor

FPN: Feature Pyramid Network

GAP: Global Average Pooling

GGO: Ground Grass Opacity

LDCT: Low Dose Computed Tomography

Lung RADS: Lung CT Screening Reporting and Data System

MRI: Magnetic resonance imaging

MTANN: Massive-Training Neural Network

NLST: National Lung Screening Trial

NMS: Non-Maximum Suppression

R-CNN: Regional Convolutional Neural Network

ROI: Region Of Interest

RPN: Region Proposal Network

SVM: Support Vector Machine

VOC: Visual Object Detection

Abstract

Lung cancer is one of the major causes of cancer-related deaths due to delayed detections at advanced stages (Freddie Bray BSc, 2018). Early detection of lung cancer can significantly increase the chances of survival of high-risk patients. Generally, computerized tomography (CT), able to produce high resolution images and a 3D reconstruction of the anatomy, is used in case of suspicious pulmonary lesions; however, especially at early stages, benign and malignant nodules show very close resemblance to each other and the erroneous detection of benign lesions can lead to additional unnecessary diagnostic analysis.

Low-dose computed tomography (LDCT) has emerged as a promising mass screening method for the early diagnosis of lung neoplasms. During the last 15 years , several observational and randomized control trials have confirmed a high sensitivity of LDCT in early stage and have demonstrated the effectiveness of these prevention programs in reducing lung cancer mortality (Team, 2011) (Giulia Veronesi, 2014) (Giulia Veronesi, 2014) (U. Pastorino, 2015) (Maurizio Infante, 2015).

However , even if these studies obtained encouraging results, the risk of over-diagnosis is still high in lung screening.

Lung nodules show different characteristic in dimension, shape and growth rates. Published recommendations for the clinical management of pulmonary nodules differentiate among solid nodules that completely obscure the lung parenchyma and sub-solid nodules that incompletely obscure the lung parenchyma, combining nodules with ground-glass attenuation and partly solid nodules into one category.

Accurate identification and characterization of malignant lung nodules and development of clear algorithms for their management, remains a challenge. The guidelines published by the Fleischner Society along with the Lung CT Screening Reporting and Data System (Lung-RADS) (Radiology) represent the two most important and considered as reference by radiologists to manage indeterminate pulmonary nodules. Most of the high quality evidence for nodule management comes from screening studies that only include patients at high risk of lung cancer, and there is an acknowledged paucity of evidence for guiding nodule management in patients with a lower background risk of cancer.

In a large-scale screening context, radiologists are faced with the challenging task of identifying subtle abnormalities on a very noisy background. Moreover, they are required to review a large number of images for each patient. In this context, the integration of a computer aided detection (CAD) systems for lung nodule detection have been demonstrated to be extremely useful, achieving a higher sensitivity with respect to other proposed methods such as double reading of LDCT (Rubin GD, 2005).

Different studies have highlighted the contribution of CAD in the detection of pulmonary nodules, providing radiologists a second opinion on early decisions diagnosis and also highlighting complementary information hidden in biomedical images. Several CAD system algorithms have been proposed for the detection of the pulmonary nodules, generally characterized by five fundamental steps: *Data acquisition, Preprocessing, Segmentation, nodule detection* and *false positive reduction.*

In the last years many researchers have investigated the development of CAD systems for lung nodule detection. Starting from very basic workflows, more and more sophisticated systems have been implemented, able to detect different type of nodule with great performance. Recently, deep learning based approaches have shown impressive results outperforming conventional methods. These findings increased the curiosity of the researchers that are implementing different deep learning models to increase the performance of CAD systems in lung cancer screening with LDCT. Among several deep-learning approaches, Convolutional neural network (CNN) gained a lot of prominence for its simplicity and performance dealing with imaging data. Different innovations were brought to CNN structure and parameters, from simple 2D CNN to more complicated 3D CNN and finally, in the context of object detection, to regional convolutional neural network(R-CNN).

R-CNNs were introduced by Ross Girshick et al. in 2014 (2014). The proposed object detection approach consist in the definition of 2000 regions that are passed into a CNN. The feature vector obtained is then passed to (i) support vector machine(SVM) for classification of the object and to (ii) a bounding box regressor for localization. The main problem of this first version of RCNN is that the algorithm needs to classify 2000 region proposals for each image. For this reason, it takes a lot of time to train the network.

To overcome these limitations, Fast R-CNN was proposed by the same author (2015), where instead of feeding 2000 regions to the CNN, the entire input image is processed

only one time by the CNN generating a convolutional feature map. On the convolutional feature map, the region proposals are defined and warped into equal size squares by means of a ROI pooling layer. From the ROI feature vector, softmax layer is used to predict the class of the proposed region and also the offset values for the bounding box.

Both R-CNN and Fast R-CNN use the selective search algorithm (J.R.R. Uijlings, 2012) to find out region proposals which is slow and time consuming. Therefore, Shaoqing Ren et al. (2017)proposed an object detection algorithm, called Faster R-CNN, that replace the selective search algorithm with a region proposal network (RPN) able to distinguish foreground from background regions.

Faster R-CNN represent a progress in terms of efficiency and performance with respect to the other aforementioned R-CNN based network. This architecture , originally designed for object detection in natural images, in the last years have been exploited for lung nodule detection with several important innovations and changes (Jia Ding, 2017).

Our work aims to evaluate the original implementation of Faster R-CNN, proposed by Shaoqing Ren (2017), in order to set up an automatic detection model for lung nodule able to identify the position of lesion in LDCT scan. We revisited the original structure of Faster R-CNN and at the same time we investigated the parameters of the network in order to be suitable for lung nodule detection, a small object detection task(Figure 0).

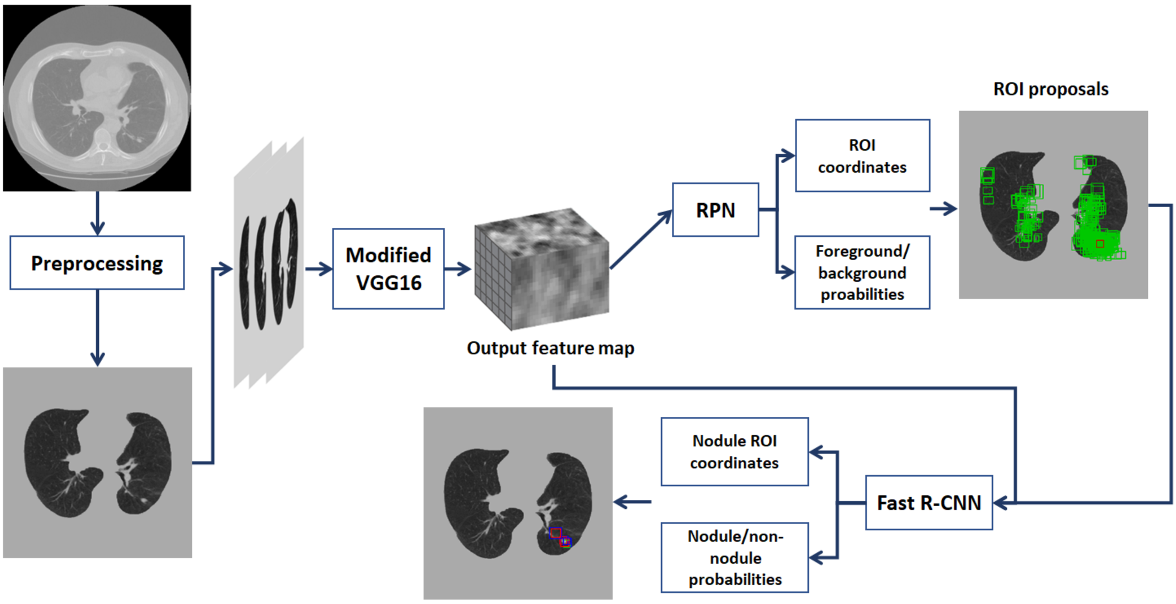


Figure 0. *Workflow of implemented lung nodule detection system*

For the development of our project, a subset of the COSMOS study (Giulia Veronesi, 2014) was considered. From the cohort of 286 patients a total number of 639 CT-scans was collected.

A CT image contains not only the lung but also other tissues, and some of them may have spherical shapes and looks like nodule. In order to overcome this issue, the original CT scans were processed according to an algorithm proposed in literature in the field of pulmonary nodule detection (Fangzhou Liao, 2015). Specifically, to rule out those aforementioned distractors, as preprocessing step, we extracted the mask of the lung and ignored all other tissues in the detection stage. Finally, we applied an intensity normalization to prepare data for deep networks.

The purpose of a Faster R-CNN is to generate candidate bounding boxes for suspicious nodules.

Faster R-CNN network consists of two main blocks : the RPN and the Fast R-CNN. RPN and Fast R-CNN share the same stack of CNN layers which takes as input the gray-scale image to elaborate it and extract a set of feature maps. These CNN layers ,dedicated to feature extraction, constitute the so called backbone neural network, in our work represented by the VGG16. In our implementation VGG16 was trained independently keeping fixed its weights during the training of RPN and Fast R-CNN. For these reasons VGG16 was considered like a separate structure from RPN and Fast R-CNN, communicating in the same way with both subnetworks.

We modified the architecture of VGG16 (i)by adding a deconvolutional layer after five convolution blocks in order to reach a structure more suitable for small-objects feature extraction, as suggested by Jia Ding et al. (2017). Considering that pre-trained weights have been adopted, the second main change of VGG16 consisted in (ii) adapting the classification output to the two classes of interest, slice with lesion and slice without lesion, to allow the fine tuning of last layers. Preliminary tests were done on VGG16,to establish the better architecture adaptation in order to avoid overfitting. The replacement of Max-pooling and Flatten layers with Global average pooling(GAP) layer in the training of our backbone network resulted necessary to avoid overfitting and process the input image to generate a good feature map.

As regards the RPN, we maintained the structure proposed in the original paper (Shaoqing Ren, 2017). Once the gray scale image was processed by the backbone network, from the output feature map a set of region of interests(ROI), also called region proposals, was generated through the RPN. Specifically, RPN was applied sliding a 3x3 window over the feature map where each pixel, center of each feature map region, corresponded to a region of the original image of dimension equal to the stride applied by the feature extractor network (Section 2.3.2). Different batch strategies have been tested in the RPN training: the "image-centric sampling" strategy ,which include inside the batch only samples of one image (Shaoqing Ren, 2017), obtained better performance with respect to a multi-image batch approach.

We filtered the proposals obtained from RPN in order to keep only proposals with higher probability to represent a lesion, with an appropriate dimension. Then, by means of non - maximum suppression (NMS), we eliminated redundant proposals.

The subset of proposals generated by RPN were then given as input to the Fast-RCNN. Specifically, the ROIs defined on the feature map previously generated by the feature extractor network, were warped into squares by means of ROI pooling layers and finally given as input to the dense layers, following the same procedure of Ross Girshick (2015) .Fast R-CNN for each proposal returned two outputs that consist in the following classification and regression variables respectively: the probability related to the different class and the adjusted coordinates in order to better fit the object of interest.

Different batch dimensions and strategies have been tested for the training of Fast-RCNN, balancing the negative and the positive class in order to obtain the higher sensitivity. By fixing the batch at 16 ROIs, we achieved the best performance.

From an architectural point of view, different sizes of the ROI pooling layer of the Fast R-CNN were exploited. In order to implement a correct ROI pooling layer, we decreased the dimension of the ROI fixed by Shaoqing et al. (2017)(7x7 pixels) by means of lower interpolation and the introduction of Max-pooling layer. This method did not appear the right solution to implement a real ROI pooling layer and other strategies should be tested in the future.

Our Faster R-CNN did not achieve a good performance, that were very far from the results obtained in literature. However, further investigation on network parameters can be done.

To understand if the poor performance was related to a bias of the net towards a particular lesion class, nodule-type specific models were tested, training the entire network on a specific nodule subset. Training the network only on solid class did not lead to a

substantial improvement. Only in the case of ground glass opacity (GGO) nodule and part-solid, we can observe a better result in term of sensitivity.

Future developments will be focused on (i)the implementation of alternating training(Appendix B), so that RPN and Fast R-CNN share the feature extractor network and on (ii) an additional reduction of the stride avoiding at the same time the overfitting of the feature extractor network. Future implementation could also take into account the integration of merging operation of overlapping candidates in adjacent slices and false positive reduction step, in order to set up a complete CAD system.

This project has been performed in collaboration with the Istituto Europeo di Oncologia, which provided the LDCT dataset collected during the COSMOS study.

Sommario

ll cancro ai polmoni è una delle principali cause di decessi per cancro a causa della rilevazione tardiva a stadi avanzati (Freddie Bray BSc, 2018). La diagnosi precoce del cancro del polmone può aumentare significativamente le possibilità di sopravvivenza. Generalmente, in caso di lesioni polmonari sospette, viene utilizzata la tomografia assiale computerizzata (CT, computed tomograpghy), in grado di produrre immagini ad alta risoluzione e una ricostruzione 3D dell'anatomia; tuttavia, soprattutto nelle fasi iniziali, i noduli benigni e maligni mostrano una forte somiglianza tra di loro e il rilevamento errato di lesioni benigne può portare a ulteriori analisi diagnostiche non necessarie.

La tomografia computerizzata a basse dosi (LDCT, low dose computed tomograpghy) è diventata un metodo promettente di screening di massa per la diagnosi precoce delle neoplasie polmonari. Negli ultimi 15 anni, diversi studi di controllo randomizzati hanno confermato un'elevata sensitività per LDCT nella fase iniziale e hanno dimostrato l'efficacia di questi programmi di prevenzione nel ridurre la mortalità per cancro ai polmoni (Team, 2011) (Giulia Veronesi, 2014) (Giulia Veronesi, 2014) (U. Pastorino, 2015) (Maurizio Infante, 2015).

Tuttavia, anche se questi studi hanno ottenuto risultati incoraggianti, il rischio di sovradiagnosi è ancora alto nello screening polmonare.

I noduli polmonari mostrano caratteristiche diverse per dimensione, forma e tasso di crescita. Le raccomandazioni pubblicate per la gestione clinica dei noduli polmonari si differenziano tra noduli solidi che oscurano completamente il parenchima polmonare e noduli sub-solidi che oscurano in modo incompleto il parenchima, quest'ultima classe comprende noduli a vetro smerigliato(GGO) e noduli parzialmente solidi.

L'identificazione e la caratterizzazione accurata dei noduli polmonari maligni e lo sviluppo di algoritmi chiari per la loro gestione rimane una sfida. Le linee guida pubblicate dalla Fleischner Society insieme al Lung CT Screening Reporting and Data System (Lung-RADS) (Radiology) rappresentano le due più importanti e considerate come riferimento dai radiologi per la gestione dei noduli polmonari. La maggior parte delle prove di alta qualità per la gestione dei noduli proviene da studi di screening che includono solo pazienti

ad alto rischio di cancro ai polmoni, e vi è una scarsità di prove per guidare la gestione dei noduli nei pazienti con un rischio di cancro inferiore.

In un contesto di screening su larga scala, i radiologi devono affrontare l'impegnativo compito di identificare anomalie sottili su uno sfondo molto rumoroso. Inoltre, sono tenuti a rivedere un gran numero di immagini per ogni paziente. In questo contesto, l'integrazione di sistemi CAD (computer aided detection) per il rilevamento dei noduli polmonari si è dimostrata estremamente utile, ottenendo una sensitività maggiore rispetto ad altri metodi proposti (es. tecnica di doppia lettura) (Rubin GD, 2005).

Diversi studi hanno evidenziato il contributo dei CAD nella rilevazione dei noduli polmonari, fornendo ai radiologi un secondo parere sulla diagnosi precoce ed evidenziando anche informazioni complementari nascoste nelle immagini diagnostiche. Sono stati proposti diversi algoritmi CAD per la rilevazione dei noduli polmonari, generalmente caratterizzati da cinque passaggi fondamentali: acquisizione dati, preprocessing, segmentazione, rilevamento del nodulo e la riduzione di falsi positivi.

Negli ultimi anni molti ricercatori hanno studiato lo sviluppo di sistemi CAD per il rilevamento dei noduli polmonari. Partendo da CAD molto elementari, sono stati implementati sistemi sempre più sofisticati, in grado di rilevare diverse tipologie di noduli con grandi prestazioni. Recentemente, gli approcci basati sul deep learning hanno mostrato risultati superiori rispetto ai metodi convenzionali. Questi risultati hanno aumentato la curiosità dei ricercatori che stanno implementando modelli basati sul deep learning per aumentare le prestazioni dei sistemi CAD nello screening del cancro del polmone con LDCT. Tra i vari approcci basati sul deep learning, la rete neurale convoluzionale (CNN, convolutional neural network) ha guadagnato molta importanza per la sua semplicità e prestazione nel trattare immagini diagnostiche. Diverse innovazioni sono state apportate alla struttura e ai parametri della CNN, dalla semplice CNN 2D a una CNN 3D più complicata e infine alla rete neurale convoluzionale regionale (R-CNN, regional convolutional neural network).

La R-CNN è stata introdotta da Ross Girshick et al. nel 2014 (2014). Il metodo di rilevazione di oggetti proposto consiste nella definizione di 2000 regioni che vengono inviate a una CNN. Il vettore delle features ottenuto viene quindi passato a (i)una macchina a vettori di supporto (SVM, support vector machine) per la classificazione dell'oggetto e a un (ii) bounding box regressor per la localizzazione. Il problema principale di questo approccio di rilevamento degli oggetti è che l'algoritmo deve classificare 2000 regioni per ciascuna immagine. Per questo motivo, l'addestramento della rete richiede molto tempo.

Per superare questo problema lo stesso autore propose la Fast-R-CNN (2015), dove invece di fornire 2000 regioni alla CNN, l'intera immagine viene processata una sola volta dalla CNN generando una feature map convoluzionale. Dalla feature map convoluzionale, le regioni proposte vengono definite e ridimensionate in quadrati di uguale dimensione per mezzo dello strato di ROI pooling. Dal vettore delle ROI features, lo strato softmax viene utilizzato per predire la classe della regione e anche i valori di offset per il riquadro di rilevamento.

Sia la R-CNN che la Fast R-CNN utilizzano l’algoritmo selective search (J.R.R. Uijlings, 2012) per ricavare le regioni proposte, un algoritmo lento e che richiede tempo. Pertanto, Shaoqing Ren et al. (2017) hanno proposto un algoritmo di rilevamento degli oggetti, chiamato Faster R-CNN, che sostituisce l'algoritmo di selective search con una rete neurale (RPN, region proposal network) in grado di distinguere regioni foreground da regioni background.

La Faster R-CNN rappresenta un progresso in termini di efficienza e prestazioni rispetto alla R-CNN e alla Fast R-CNN. Questa architettura, originariamente progettata per il rilevamento di oggetti in immagini naturali, negli ultimi anni è stata sfruttata per il rilevamento dei noduli polmonari attraverso diverse importanti innovazioni e modifiche(Jia Ding, 2017).

Il nostro lavoro mira a valutare l'implementazione originale della Faster R-CNN, proposta da Shaoqing Ren et al. (2017), al fine di impostare un modello di rilevamento automatico del nodulo polmonare in grado di identificare la posizione della lesione nella CT. Abbiamo rivisitato la struttura originale della Faster R-CNN e allo stesso tempo abbiamo rivalutato i parametri della rete per adattare la rete al rilevamento di noduli polmonari, oggetti molto piccoli(Figura 1).

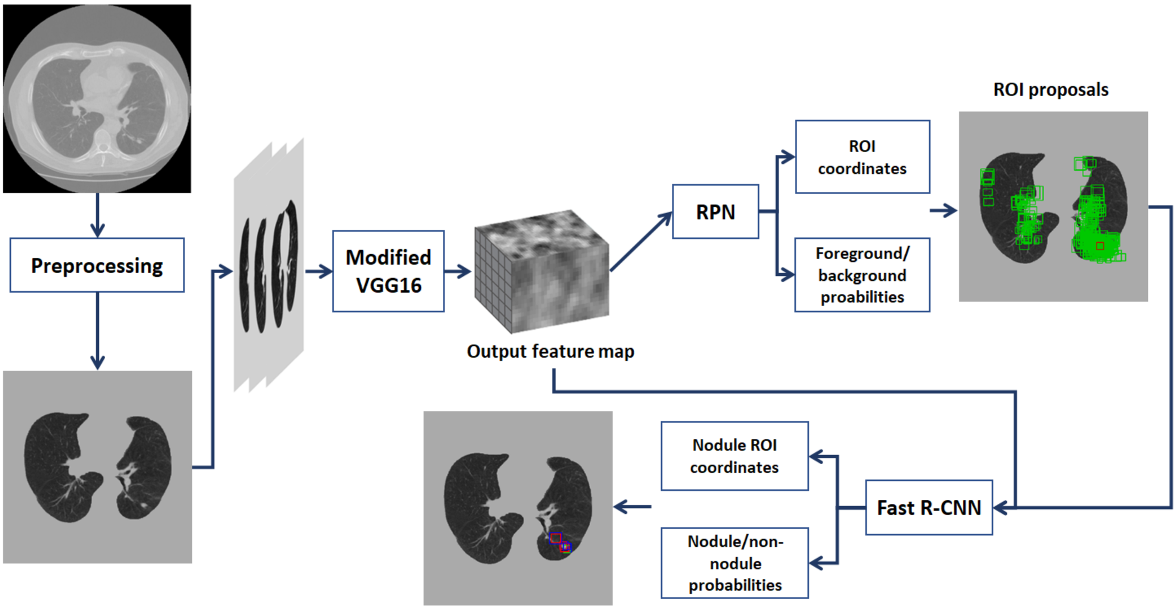


Figura 1. *Schematizzazione del sistema di rilevamento dei noduli polmonari implementato*

Per lo sviluppo del nostro progetto, è stato considerato un sottoinsieme dello studio COSMOS (Giulia Veronesi, 2014).Dalla coorte di 286 pazienti è stato raccolto un numero totale di 639 scansioni CT.

Una CT contiene non solo il polmone ma anche altri tessuti e alcuni di essi possono avere forme sferiche e assomigliare a noduli. Per ovviare a questo problema, le CT originali sono state elaborate secondo un algoritmo proposto in letteratura nel campo del rilevamento dei noduli polmonari (Fangzhou Liao, 2015). Nello specifico, per escludere quei suddetti distrattori, nella fase di preprocessing, abbiamo estratto la maschera del polmone e ignorato tutti gli altri tessuti in fase di rilevamento. Infine, abbiamo applicato una normalizzazione dell'intensità per preparare i dati per la rete neurale.

Lo scopo della Faster R-CNN è generare riquadri di delimitazione per noduli sospetti. La Faster RCNN è composta da due blocchi principali: RPN e Fast R-CNN. RPN e Fast R-CNN condividono lo stesso blocco di strati della CNN che prende come input l'immagine in scala di grigi per elaborarla ed estrarre una serie di feature maps. Questi strati della CNN, dedicati all'estrazione di features, costituiscono la cosiddetta rete neurale backbone, nel nostro lavoro rappresentata dalla VGG16. Nella nostra implementazione la VGG16 è stato addestrata in modo indipendente mantenendo i pesi fissi durante l'allenamento di RPN e Fast R-CNN. Per questi motivi la VGG16 è stata considerata come una struttura separata da RPN e Fast R-CNN, comunicando allo stesso modo con entrambe le reti.

Abbiamo modificato l'architettura della VGG16 (i) aggiungendo uno strato deconvoluzionale dopo cinque blocchi di convoluzione al fine di raggiungere una struttura più adatta all'estrazione di features di piccoli oggetti, come implementato da Jia Ding et al. (2017). Usando i pesi pre-trainati, la seconda sfida principale della VGG16 consiste nell' (ii) adattare l'output di classificazione alle due classi di interesse, fetta con lesione e fetta senza lesione, per permettere il fine tuning degli ultimi strati. Sono stati effettuati test preliminari sulla VGG16, per stabilire il miglior adattamento dell'architettura al fine di evitare l'overfitting. La sostituzione dello strato di Max-pooling e Flatten con quello di Global average pooling (GAP) nell' allenamento della VGG16 è risultata necessaria per evitare l'overfitting e per ottenere una buona feature map.

Abbiamo mantenuto la struttura della RPN del documento originale (Shaoqing Ren, 2017). Una volta che l'immagine in scala di grigi è stata elaborata dalla rete backbone, dalle feature maps in output sono state generate tramite l'RPN una serie regioni d'interesse(ROI, region of interests), anche dette regioni proposte. In particolare, la RPN è stata applicata facendo scorrere una finestra 3x3 sulla feature maps in cui ogni pixel (centro di ciascuna regione della feature maps) corrispondeva a una regione dell'immagine originale di dimensione uguale al passo applicato dalla VGG16.

Diverse strategie batch sono state testate nella costruzione della RPN: la strategia di "image-centric sampling strategy", che include all'interno del batch solo campioni di un'immagine (Shaoqing Ren, 2017), ha ottenuto prestazioni migliori rispetto a un approccio batch multi-immagine.

Abbiamo filtrato le regioni di interesse provenienti dalla RPN in modo da mantenere solo le regioni con maggiore probabilità di rappresentare una lesione, con una dimensione appropriata. Quindi, per mezzo del metodo della non-maximum suppression (NMS), abbiamo eliminato le regioni ridondanti.

Il sottoinsieme di regioni generate dalla RPN sono state date in ingresso alla Fast R-CNN. In particolare, le regioni d'interesse definite rispetto alla feature map generata dalla VGG16 sono state ridimensionate in quadrati per mezzo dello strato di ROI pooling e infine date come input agli strati densi, seguendo la stessa procedura di Ross Girshick et al. (2015). Ciascuna proposta ha restituito due output che consistono rispettivamente nelle seguenti variabili di classificazione e regressione: la probabilità relativa alla diversa classe e le coordinate corrette per meglio adattarsi all'oggetto di interesse.

Diverse dimensioni e strategie dei batch sono state testate per addestrare la Fast-RCNN, bilanciando la classe negativa e positiva per ottenere la maggiore sensitività. Fissando il batch a 16 ROI, abbiamo ottenuto le migliori prestazioni.

Da un punto di vista architettonico, sono state sfruttate diverse dimensioni dello strato di ROI pooling della Fast R-CNN. Al fine di implementare un corretto livello di ROI pooling, abbiamo diminuito la dimensione della ROI fissata da Shaoqing et al. (2017) (7x7 pixel) mediante una minore interpolazione e l'introduzione del livello di Max-pooling. Questo metodo non è risultato la soluzione corretta per implementare un livello di pooling ROI effettivo e altre strategie dovrebbero essere testate in futuro.

Il nostro metodo basato sulla Faster R-CNN non ha ottenuto una buona performance, lontana dai risultati ottenuti in letteratura. Tuttavia, possono essere fatte ulteriori investigazioni sulla rete.

Per capire se lo scarso rendimento fosse correlato a un bias della rete verso una particolare classe di lesione, sono stati testati modelli specifici del tipo di nodulo, addestrando l'intera rete su uno specifico sottoinsieme di noduli. Allenare la rete solo sui tumori solidi non ha portato a un miglioramento sostanziale. Solo nel caso di nodulo a vetro smerigliato (GGO) e parzialmente solido, possiamo osservare un risultato migliore in termini di sensitività.

Gli sviluppi futuri saranno focalizzati su (i) l'implementazione dell'allenamento alternato (Appendice B), così che RPN e Fast R-CNN condividono gli strati della CNN e su (ii) un’ulteriore riduzione del passo evitando allo stesso tempo l'overfitting della VGG16. Un'implementazione futura potrebbe anche tenere conto dell'integrazione dell'operazione di fusione di candidati sovrapposti in sezioni adiacenti e della fase di riduzione dei falsi positivi, al fine di impostare un sistema CAD completo.

Questo progetto è stato realizzato in collaborazione con l'Istituto Europeo di Oncologia, che ha fornito il dataset LDCT raccolto durante lo studio COSMOS.

Chapter 1. Introduction: lung cancer

Lung cancer is the worldwide leading cause of tumor related dead and, along with breast cancer, is the most common cancer worldwide(Figure 2) (Freddie Bray BSc, 2018). This pathology is characterized by a lack of symptoms at the early stage and at the time of diagnosis 70% of patients are already inoperable and the chances of recovery are very limited. For these reasons, early detection has a very important role for successful lung cancer treatment and for mortality related rate reduction.

In the actual clinical practice, X-ray imaging techniques are the most used to identify lung cancer in a non-invasive way. Specifically, in case of suspect of lung cancer, computerized tomography (CT) represents the gold standard granting, with respect to chest radiography (RX), a better identification of pulmonary nodules thanks to the 3D reconstruction of the anatomy but at the expenses of a higher radiation dose given to the patient. However, the introduction of low-dose CT (LDCT) brought to a reduction of the dose from 7 to around 1.4-1.6 mS paving the way of lung cancer screening based on CT.

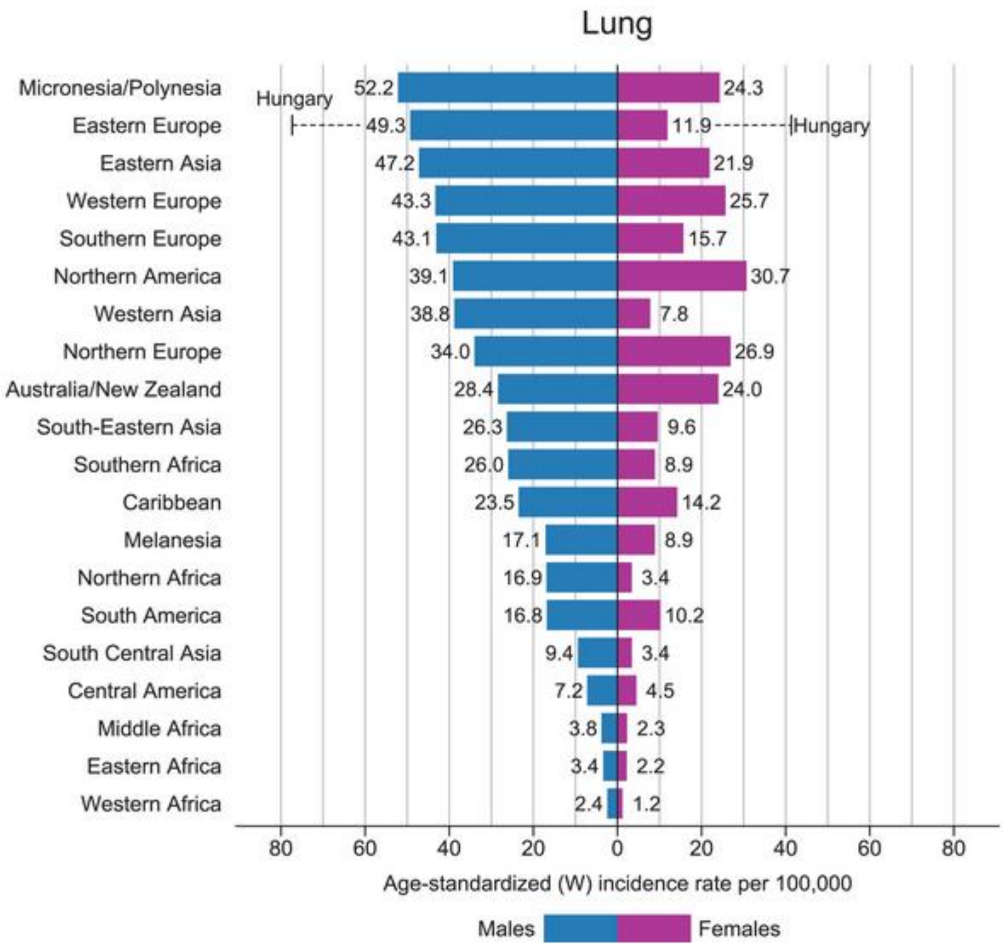


Figure 2.*Bar Chart of Region-Specific Incidence Age-Standardized Rates by Sex for lung Cancers 2018.Rates are shown in descending order of the world (W) age-standardized rate among men, and the highest national rates among men and women are superimposed.*

1.1. Lung screening

During the last 15 years, several clinical trials have shown that LDCT can reliably allow lung nodules identification of a few millimetre in diameter in asymptomatic individuals at high risk and demonstrated the effectiveness of this prevention program in reducing lung cancer mortality (Team, 2011) (Ying Ru Zhao, 2011) (U. Pastorino, 2015) (Maurizio Infante, 2015) (Giulia Veronesi, 2014).

Among the numerous studies carried on in the last decades, the National Lung Screening Trial (NLST) (Team, 2011) is one of the most important in demonstrating the role of screening in the early diagnosis of lung cancer. Started in 2002, NLST was a randomized clinical trial conducted by the American College of Radiology Imaging Network (ACRIN) and the Lung Screening Study group. The enrolled subjects were 53,454 current or former heavy smokers (30 pack-years) without signs, symptoms, or history of lung cancer with age between 55 and 74. Participants were randomly assigned to receive three annual screens with either low-dose helical CT or standard chest X-ray. It reveals that participants who received low-dose helical CT scans had a 15 to 20 % lower risk of dying from lung cancer than participants who received standard RX. This is equivalent to approximately three fewer deaths per 1,000 people screened in the LDCT group compared to the chest X-ray group over a period of about 7 years of observation.

On average over the three rounds of screening exams, 24.2 percent of LDCT screens and 6.9 percent of the chest X-rays were positive. In both arms of the trial, the majority of positive screens led to additional tests.

Despite the initial enthusiasm, the data emerging from the American NLST study was not yet sufficiently convincing to recommend spiral LDCT as a routine screening procedure but subsequent European trials opened new perspectives and hopes.

In September 2018, new data from the second largest randomized-controlled trial, the Dutch-Belgian Lung Cancer Screening study (NELSON) (Ying Ru Zhao, 2011) showed an even bigger reduction in deaths with respect to NLST.

More than 15,000 high risk patients were enrolled and followed for more than 10 years through national registries and case notes review.

In this case, the group of screened subjects was compared with those not screened which had similar baseline characteristics, including age, gender ratio, smoking history, and smoking cessation.

About 50% of the cancers diagnosed in the screening arm were early stage (65% to 70% at stages IA to II) while about 70% of cancers in the control arm were stage III/IV at diagnosis.

Overall result of NELSON trial confirmed those found in NLST trial: LDCT scanning decreased mortality by 26% in high-risk men and up to 61% in high-risk women over a 10-year period.

Another confirmation of the effectiveness of lung cancer screening, resulted from the MILD study (U. Pastorino, 2015), conducted by the National Cancer Institute of Milan. This trial involved 4,000 heavy smokers since 2005 underwent spiral LDCT, annually or biennially, for a period of 10 years. The study has shown that an early diagnosis program that continues beyond 5 years, up to 10 years of screening, can achieve a 39% reduction in lung cancer mortality with respect to the control group who had just primary prevention.

The greater efficacy of NELSON and MILD compared to the NLST study were attributed to the longer duration of follow-up but also by the choice of observational control arm. In fact, even if less sensitive, thoracic chest radiography can anticipate the presence of lung cancer. Instead the control arm of NELSON and MILD did not undergo to any type of diagnostic screening.

Different findings resulted instead from DANTE trial that reopened the debate on the efficacy of lung cancer screening.

The DANTE trial (Maurizio Infante, 2015) started in 2001 with the purpose to verify whether the large-scale application of diagnostic tests could help reduce mortality due to lung cancer. The study involved around 2,500 heavy male smokers aged between 60 and 74 years of age. The protocol was based on the use of spiral LDCT and molecular biology tests and stated that there was still insufficient evidence to recommend these tools as routine screening procedures in heavy smokers.

According to the obtained results, the author stated that,actually, it was not possible to recommend LDCT as a spontaneous screening method: it was necessary to narrow the range of patients considered at risk by means of new clinical, epidemiological and biological indicators, minimizing the phenomena of over-diagnosis and the percentage of unnecessary surgical procedures.

Another relevant trial, conducted by the European Institute of Oncology between 2005 and 2015, is the Continuous Observation of Smoking Subjects (COSMOS), observational study addressed to smokers or ex-smokers high risk patients (smoking history ≥20 pack-years).

An overall group of 5201 asymptomatic individuals aged 50 years or older were enrolled in the 10-year single-center COSMOS study and underwent multi-detector LDCT annual repeated scans (Giulia Veronesi, 2014). Specifically, in case of no lesions or in presence of nodules <5 mm, patients underwent repeated LDCT at 1 year; with nodules of 5.1–8 mm, LDCT was repeated 3 months later whereas for nodules >8 mm a combined CT-positron emission tomography (CT-PET) was applied before a normal biopsy. The minimization of invasive techniques was indeed one of the particularities of the study.

The COSMOS study provides encouraging results both for its sensitivity (90.3%) and specificity (99.4%) in the detection. Another important aspect is that 78% of diagnosed tumors occurred at the localized stage, obtaining a 5-year survival rate of 78%. With respect to the screening trials presented above, the portion of patients that undergo further tests was limited to 6.4%.

However, although the encouraging results in terms of number of false positives , the risk of over-diagnosis was still too high in lung screening. It is frequent that the detection of a lesion does not lead to malignancy and therefore leads the subject to an unnecessary exposure to radiation. This limit is partly due to the CT parameters, but it is also related to the experience of the clinician in reading the image and the position of the nodules inside the lung volume.

Even if a lot of studies and trial support the utility of LDCT screening for lung cancer, there is still concern that exposure to the ionizing radiation of LDCT might increase the risk of developing solid cancers and leukemia. (Council, 2016)

To address this problem, in a recently study, Rampinelli et al (2017) evaluated the cumulative radiation exposure and lifetime attributable risk of cancer incidence associated with LDCT in the COSMOS study.

The median cumulative effective dose after 10 years was found equal to roughly 9 mSv for men and 13 mSv for women. Compared with standard dose CT exam, this means that a patient taking part to a 10 year LDCT screening program would receive a dose similar to that delivered with a standard chest (7-8 mSv) or abdomen-pelvis (13-14 mSv) CT scan.

On the overall set of lung cancer diagnosis of the COSMOS study, only 1.5 lung cancers and 2.4 major cancers were found to be hypothetically caused by radiation which corresponds to an additional overall risk of major cancer of 0.05% (Cristiano Rampinelli, 2017)(Figure 3).

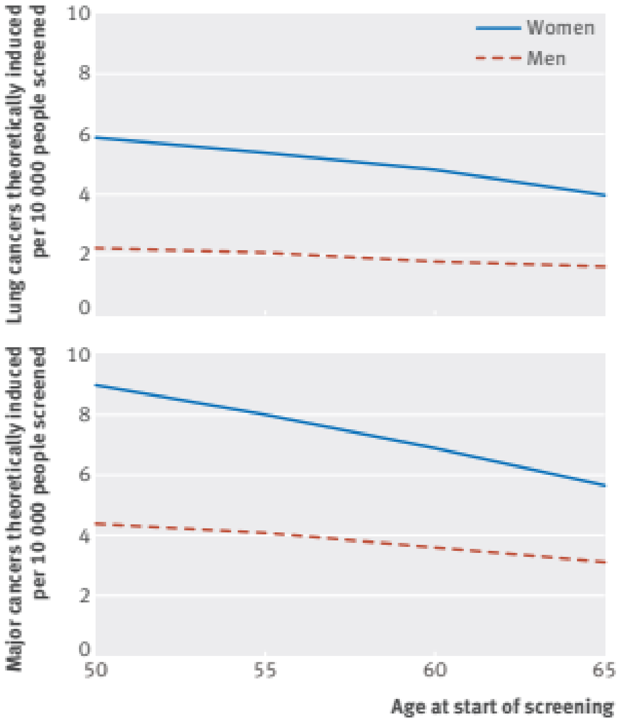


Figure 3. *Lung cancers and major cancers theoretically induced per 10 000 people screened, according to sex and age at start of CT screening for lung cancer*

Also, additional measures can be applied for further dose reduction: firstly, with accurate patient selection and optimization of acquisitions protocols; secondly, taking advantage of new reconstruction algorithms, such as iterative reconstruction, that allow to obtain the same quality of the image with a 80% dose reduction respect to standard filtered back-projection.

Despite the importance of lung cancer screening for mortality reduction underlined by the majority of the trials (NLST, NELSON, MILD and COSMOS) and the results obtained by Rampinelli study for the dosimetric concerns, the problems of high cost and risk of false positives remain and constitute a potential obstacle to the large-scale clinical implementation of this prevention program.

These problems stress the need for automatic or semi-automatic tools to support the detection of pulmonary nodules and the clinical decision-making process, where an increase of exams, linked to the introduction of screening programs in clinical practice, needs more advanced and fast algorithms (Cristiano Rampinelli, 2012).

1.2. Management of pulmonary nodules

The incidental finding of lung nodules in asymptomatic individuals is an increasingly common clinical issue encountered by radiologists in daily clinical practice. Accurate identification and characterization of malignant lung nodules and implementation of clear algorithms for their management, still remain a challenge. (Konstantinos Loverdos, 2019)

According to the appearance in the CT scan, non-calcified pulmonary nodules are distinguished by radiologists in solid and sub-solid nodules. The latter are further classified as non-solid and part-solid nodules(Figure 4). In addition to their aspect, different nodule classes are characterized also by different growth rates and probability of malignancy, therefore a distinct management is required.

Specifically, non-solid nodules, called also ground grass opacity (GGO), are more likely to be malignant, but their growth rate is usually slower with respect to solid lesions. Solid and part-solid nodules are the lesion categories most frequently identified and also those related to the majority of false positive diagnosis. Indeed, when malignant, these lesions are more likely to be invasive and faster growing cancers.

Size and growth are the most important parameters in the management of pulmonary nodules. However, the decision of the radiologist, when an indeterminate pulmonary nodule is detected, often depends on his experience and other external factors that cause variability among operators.

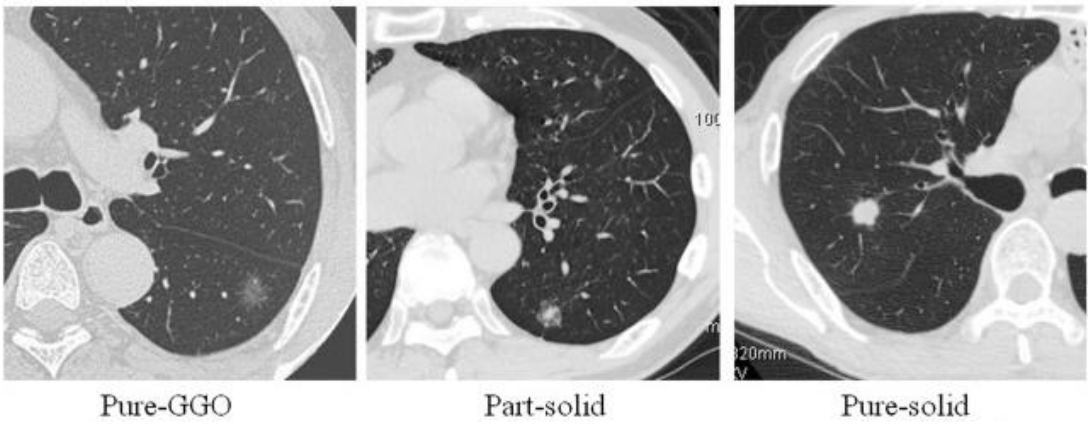


Figure 4. *Example of lung nodules. Pure ground glass opacity (GGO) tumor was defined as a lung tumor without a solid component and part solid tumor was defined as a lung tumor characterized by both a GGO and solid component, whereas pure solid tumor was defined as a lung tumor showing only consolidation without GGO component*

1.2.1 Clinical practice in managing and detecting pulmonary nodules through LDCT

Many scientific societies published guidelines recommending standard procedures for the management of lung nodules(Figure 5).

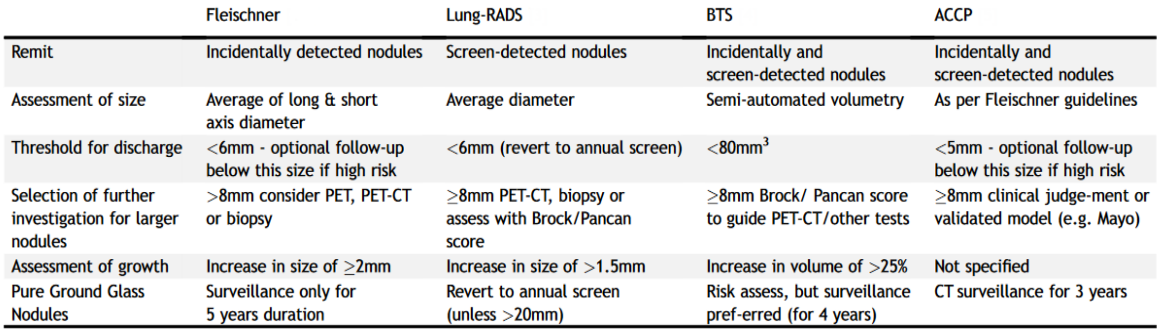
The guidelines published by the Fleischer Society, recently revised in 2017 (Heber MacMahon, 2017), along with the Lung CT Screening Reporting and Data System (Lung-RADS) (Radiology) created by the American College of Radiology in 2014, are two of the most important and considered as reference by radiologists to manage indeterminate pulmonary nodules.

All the mentioned guidelines agree with the need to minimize radiation dose for CT surveillance, according to the low likelihood of malignancy of small nodules and to the choice to avoid nodule management in patients with a lower background risk of cancer. All guidelines recognize that sub-solid nodules need a different management approach, characterized by a less interventional and aggressive approach (M.Callister).

The threshold size, under which nodules can be ignored, is similar between the guidelines: Lung-RADS recommends to avoid intervention for nodules<6mm (or<4mm for new nodules), while the Fleischner Society guidelines states to consider nodule under 6 mm only with high risk subjects.

On the contrary of Lung-RADS that defines a risk-prediction scores, the Fleischner guidelines highlight the presence of several risk factors to be considered, but do not suggest the use of a risk prediction score.

Another determinant parameter in considering the lesion as malignant, is the growth rate which is considered as significant for both guidelines when an increase in diameter of 1.5/2mm is observed. Other guidelines (M. Callister, 2015) fix this threshold on the volume, classifying as malignant nodules with volume changes higher than 25%.



	Fleischner	Lung-RADS	BTS	ACCP
Remit	Incidentally detected nodules	Screen-detected nodules	Incidentally and screen-detected nodules	Incidentally and screen-detected nodules
Assessment of size	Average of long & short axis diameter	Average diameter	Semi-automated volumetry	As per Fleischner guidelines
Threshold for discharge	<6mm - optional follow-up below this size if high risk	<6mm (revert to annual screen)	<80mm^3	<5mm - optional follow-up below this size if high risk
Selection of further investigation for larger nodules	>8mm consider PET, PET-CT or biopsy	≥8mm PET-CT, biopsy or assess with Brock/Pancan score	≥8mm Brock/ Pancan score to guide PET-CT/other tests	≥8mm clinical judge-ment or validated model (e.g. Mayo)
Assessment of growth	Increase in size of ≥2mm	Increase in size of >1.5mm	Increase in volume of >25%	Not specified
Pure Ground Glass Nodules	Surveillance only for 5 years duration	Revert to annual screen (unless >20mm)	Risk assess, but surveillance pref-erred (for 4 years)	CT surveillance for 3 years

Figure 5.*Summary of significant differences between nodule management strategies recommended by various guidelines/assessment categories.*

Even though a lot of studies exist regarding the management of lung nodule, these guidelines are followed only by minority of clinicians and usually the management is based on the judgment of them.

For what concerns the detection of pulmonary nodules, also in this case there is the necessity of standards in the reading procedure. Different studies revealed a low inter-observer agreement amongst radiologists due to multiple factors such as CT parameters, reader experience and nodule location, and some of them investigated how the radiologist's sensitivity can be increased.

As first, the detection of the nodule is influenced by its location, as asserted in the study of Naidich et al. (Naidich DP, 1993) where they showed that perihilar lung nodules were detected with a sensitivity of 36.7% versus 73.9% of peripherally located nodules. They also noticed that nodules attached to vessels were detected with a low sensitivity (32.5%).
At the same time CT parameters play an important role: the application of thin-slice CT increases the readers sensitivity for lung nodule detection (Fischbach F, 2003)
Many approaches have been introduced in order to improve the radiologist performance.
Specifically, a double reading technique has been proposed for LDCT. Wormanns et al. (2005) showed that the average sensitivity in identifying lung nodules for single readers raised from 64% to 79% with double reading. Also, the integration of a computer aided diagnosis/detection (CAD) systems in the clinical practice have been demonstrated to be extremely useful for the detection of small pulmonary nodules and can achieve an even higher sensitivity with respect to double reading, as showed by Rubin (Rubin GD, 2005). Therefore, CAD systems can act as a “second opinion” for the radiologists, by making final decision quickly with higher accuracy and greater confidence.
Moreover, the advance of technology lead to an increasing number of diagnostic images to be reviewed and as consequence a larger amount of data radiologists have to deal with. This limit opens the research interest in CAD system that are able to automatically detect and characterize pulmonary lesions.

1.3 Cad systems

Different studies demonstrated the importance of CAD systems as tools that can help the detection of pulmonary nodules, provide radiologists a second opinion on early decisions diagnosis and also can highlight complementary information hidden in biomedical images. Particularly attention was always given to the automation of nodules detection, being the most tedious part and prone to human errors.
CAD systems can be subdivided in two branches: CADe (computer-aided detection system) and CADx (computer-aided diagnosis system), which aim respectively to detect

lesions and to a propose a characterization of the lesion, for example, determining the malignancy and staging of the cancer.

CAD systems for detecting pulmonary nodules are usually composed of five subsystems (Macedo Firmino, 2014)(Figure 6):

- *Data acquisition* subsystem is responsible for obtaining medical images.
- *Preprocessing* is the treatment that attempt to improve the quality of the image and to increase the precision and accuracy of algorithms that are introduced after this step. Preprocessing step aims to remove imperfections caused by the image acquisition process, noise and lack of contrast.
- *Segmentation* aims to separate the region of interest(lung) from other organs and tissues in order to reduce the computational cost and to simplify the detection of lung nodule.
- N*odule detection* stage attempts to check the presence of lung nodule and then to detect it. The main challenge is to recognize the true nodule from other pulmonary parenchymatous injuries or different organs and tissues(false positive).
- *False positive reduction* step aims to solve the main problem of lung nodule detection: the number of false positive, that compromise and reduce precision. For this reason, after the detection stage, it is used a classifier that aim to learn the characteristic of nodule and then try to separate the nodule from other tissue or injury (false positive). The main classifiers of false positive reduction steps are: linear discriminant analysis , clustering , Markov random field , artificial neural networks , support vector machines (SVM) , massive-training neural network (MTANNs) , and double-threshold cut.

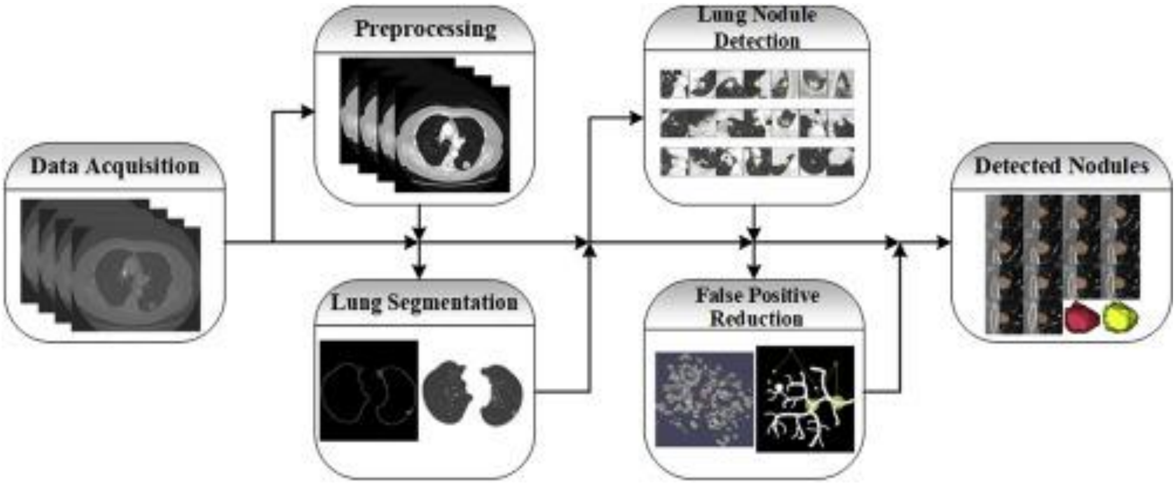


Figure 6. *Typical workflow of CAD system for lung nodule detection composed by five steps: data acquisition, preprocessing, lung segmentation, lung nodule detection and false positive reduction step.*

In the last years, several studies have been conducted on the development of CAD systems for lung nodule detection. Starting from very simple CAD algorithm, more and more sophisticated ones have been implemented, able to reach an higher detection speed and sensitivity and also able to detect different type and shape of nodules. Recently, deep learning approaches have shown impressive results outperforming classical methods. This increases the curiosity of the researchers that are implementing different deep learning techniques to improve CAD systems performance in lung cancer screening with computed tomography.

In the following overview, several CADe systems proposed in literature during the last decades will be discussed.

1.3.1. Conventional CAD systems

In 1963 Lodwick et al. (1963) proposed for the first time the use of digital computers for lung nodule detection. However, only in the late 80s, the first CAD systems for detecting lung nodules was proposed. Although encouraging results have been obtained, these first attempts were not successful, due to lack of computational resources and sophisticated image processing algorithm.

The first proposed CAD systems were usually based on simple thresholding operations and on the circularity or sphericity calculation of the observed regions. The nodule isolation strategy of Giger et al. consists in the investigation of circularity and size and their variation with threshold level (Giger M. D., 1988) .

In a second work (Giger M. A., 1990), with the addition of a feature-extraction technique, Giger reduced the true-positive rate by 13% and the false-positive rate by 50% . With respect to Giger et al. (1990), the framework proposed later by Armato et al. (1999) includes both 2D and 3D analyses. A rolling ball algorithm and multiple gray-level thresholds were applied to the lung regions to identify nodule candidates reaching an improvement on juxtapleural nodules detection (Armato SG G. M., 1999).

In 2001, Lee et al. found a solution to speed up the template matching technique. Specifically, with the introduction of a genetic algorithm, Lee was able to determine the target position in the image and to select the correct template from several patterns for a faster matching (Lee Y, 2001).

One of the biggest barriers was the lack of labelled data and especially the absence of database with a significant number of medical images. In 2004, a public database of chest CT images of healthy patients and patients with lung cancer in different stages was created by the Lung Image Database Consortium (LIDC). This database had a fundamental role in the improvements done with CAD systems for lung nodule detection in the last two decades in particular for what concerns the detection of smallest lesions.

Few years later, Hara et al. proposed a small-object detection system that use second order autocorrelation and multi-regression analysis to detect small nodules (diameter ≤7 mm) on CT scans. By combining a previously developed technique, the algorithm improved the sensitivity(94%) and decreased the value a false positive per scan(2.05 FP/scan) (T. Hara, 2005).

In 2007, Murphy et al. presented a CAD system, ISI-CAD, characterized by the introduction of the region growing technique and morphological smoothing. Geometric filters and the k-nearest neighbor classifier were then introduced to determine the candidate nodules and to reduce false positives (Murphy K, 2007).

In 2009 Ye et al. presented a new system that optimizes the detection of non-solid nodules, one of the main criticality of CAD system . The algorithm consisted in calculating , for each voxel of the lung, the volumetric shape index map and the ldquodotrdquo map in order to highlight objects with spherical shape. The combination of volumetric shape index map and idquodotrdquo map offered a descriptor for the initial nodule candidate generation (Ye, 2009).

During the evolution of CAD systems, detection algorithm started to be integrated with classification techniques, that aim not only to distinguish malignant from benign nodules, but also to stratify the degree of malignancy. Namin et al. proposed a CAD system for lung nodule detection and classification on CT scans using volumetric shape index (SI) and fuzzy k- NN. Features such as sphericity, mean and variance of the gray level, elongation and border variation of potential nodules were extracted to classify nodules as benign or malignant. Finally, fuzzy K-Nearest Neighbor(fuzzy K-NN) was employed to classify potential nodules as non-nodule or nodule with different degree of malignancy (S. Matsumoto, 2008).

Similarly in 2011, Kumar et al. (2011) presented a CAD system that not only attempts to detect lung nodules through fuzzy inference system but at the same time classifies nodules into benign nodule (granuloma, hamartoma, for example), malignant neoplasia or malignant neoplasia in advanced stage.

The use of SVM, often associated with other machine learning algorithms, was also central both for nodule detection and false positive reduction. Riccardi et al. presented a new system where 3D fast radial filter was applied in order to detect candidate nodules and estimate their geometrical features. Finally, a false positive reduction step, comprising a heuristic FPR, applied threshold based on geometrical features and a supervised false positive reduction, based on SVM classification, was enhanced by a feature extraction method based on maximum intensity projection and Zernike moments. (A. Riccardi, 2011). In 2012 Hong, Li and Yang proposed a CAD system where adaptive thresholding was used for detection of candidate nodules and then SVM was used to eliminate false positives. (Shao H, 2012). In the same year Orozco et al. presented a CAD system that computed the characteristics of texture by means of Discrete Cosine Transform and the Fast Fourier Transform and used SVM for detecting lung nodules (Orozco HM, 2012).

1.3.2. Deep learning based CAD systems

Deep learning algorithms have been considered valuable tools in the field of medical imaging, for lesion detection, characterization, and analysis.

In the context of lung nodule detection, Artificial neural networks (ANN) were initially adopted by many authors to reduce false positives, replacing SVM algorithm, and then, with the evolution of deep learning, significant improvements in speed and sensitivity have been done.

The first applications of ANN in CAD took place in the late 90s when a work of particular relevance was proposed by Xu et al. (1997).They introduced a CAD system where nodule candidates were selected initially by multiple gray-level thresholding of the difference image and then classified into six groups. A large number of false positives were eliminated by adaptive rule-based tests and an ANN (Xu X-W, 1997).

In the last twenty years, several contributions and progresses were done with deep learning based CAD systems especially with the advent of convolutional neural networks (CNN) and massive training artificial neural network (MTANN) that are the two most recent approaches used in deep learning. Both structures use pixel values in images directly as input information, instead of handcrafted features calculated from segmented regions of interest (ROIs) (Tajbakhsh N, 2016).

Among the two, CNN is the most used architecture when deep learning is adopted to solve imaging related problems.

A convolutional layer is characterized by a convolution operation, applied by sliding a kernel on the image. The kernel acts as a filter and, on the basis of its values, different features are extracted from the input image (Appendix A). Stacking multiple convolutional layers, deep learning models based on CNN are obtained, also known as "ConvNets". Usually, in the first layers of a ConvNet, low-level features such as edges, color and gradient orientation are extracted while last layers are associated to more abstract features. According to the kernel dimensions, ConvNets can elaborate both 2D or 3D images.

For its efficiency and lower computational cost with respect to a 3D approach, 2D CNN have been more frequently used for lung nodule detection. Setio et al. (2016) proposed a CAD system based on multiple streams of 2-D Convolution. Initially, they combined three candidate detector algorithms specially designed for solid, sub-solid, and large nodules for nodule candidate detection. Then for each candidate, a set of nine 2-D patches of size 64×64 pixels were extracted from differently oriented planes and feature extraction from each one was done through multiple 2-D ConvNets .The outputs were finally merged applying three fusion techniques: committee fusion, late fusion and mixed fusion.

Dou et al. (2016) proposed a 3-D CNN-based architecture for lung nodule detection. A hierarchical architecture consisting in 3-D CNNs was used to encode spatial information and representative features. For the three architectures, different receptive field sizes were adopted and finally, features extracted from the three CNNs were merged for nodule detection. The main advantage of 3-D CNN is that it is able to take into account more information related to the context around the lesions producing multi-view features.

The availability of a large amount of data and innovations in the hardware technology has intensified the research in CNNs. Several inspiring ideas to bring advancements in CNNs based deep learning models have been explored, such as the use of different activation and loss functions, parameter optimization and regularization techniques, and especially architectural innovations. Regional convolutional neural networks (R-CNN) represents certainly one of the main deep learning architecture discovery which brought to considerable progresses in solving object detection problems.

1.4. Overview on R-CNN based network

R-CNN is a pioneering approach originally implemented to detect objects in a natural image where the object detection aim is to estimate the position of the bounding box around the object of interest. In the last decade, different improvement has been done with RCNN: it has been used for pulmonary nodules detection too, harder task with respect to the identification of objects in a natural image .

In this paragraph, an overview of the main improvements done with R-CNN is presented, to understand how it was born the Faster-R-CNN, one of the most recent implementations of R-CNN as well as the deep learning architecture chosen for our project.

1.4.1. R-CNN

One of the first versions of R-CNN is that proposed by Girshick et al. (2014). As schematized in Figure 7, the workflow designed by Girshick et al. starts with the extraction of a series of region proposals from the input image. After the normalization of regions dimensions to a fixed size, a feature map is extracted from each warped region through a deep CNN. The generated feature map is then used by SVM classifier to assign a specific category to the region. Despite the use of the selective search algorithm (J.R.R. Uijlings, 2012) that allowed to limit the set of initial proposed regions to around 2000, the computational cost of this R-CNN is too high, being the proposed regions singularly processed. Moreover, selective search is a fixed algorithm, so there is no learning happening at that stage. This could lead to the generation of bad candidate region proposals.

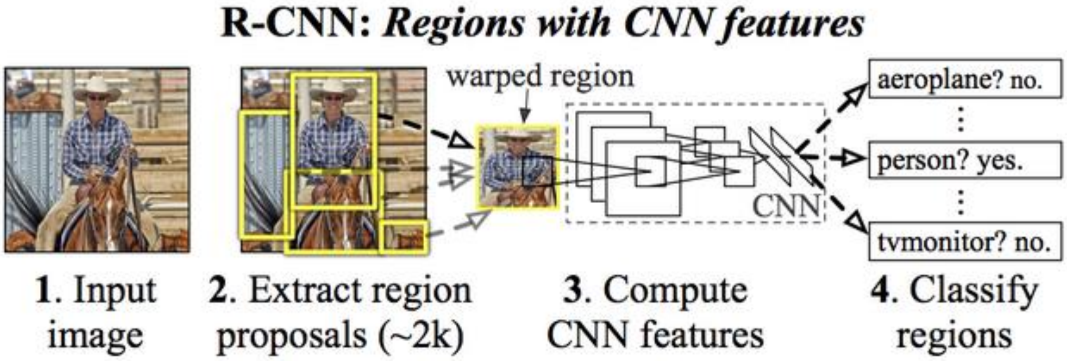


Figure 7. R-CNN workflow

1.4.2. Fast R-CNN

The limitations of the first R-CNN version of Girshick et al. (2014) were partially overcome in a subsequent work (Girshick, 2015) where the author presented an object detection algorithm known as Fast R-CNN. The main difference introduced in Fast R-CNN is that the entire input image (not the region proposals as before) is fed into the deep CNN to generate a convolutional feature map (Figure 8). The region proposals are then selected from the feature map given as output from the deep CNN. Feature map regions are then normalized to a fixed size by means of a ROI Pooling layer. Following, a series of fully connected layers elaborate the normalized feature map region whose belonging object category is estimated along with its bounding box position in the original image.

Fast R-CNN is definitely faster respect to R-CNN because one image instead of 2000 regions is fed into CNN to obtain feature map and so the convolution operation is done only one time.

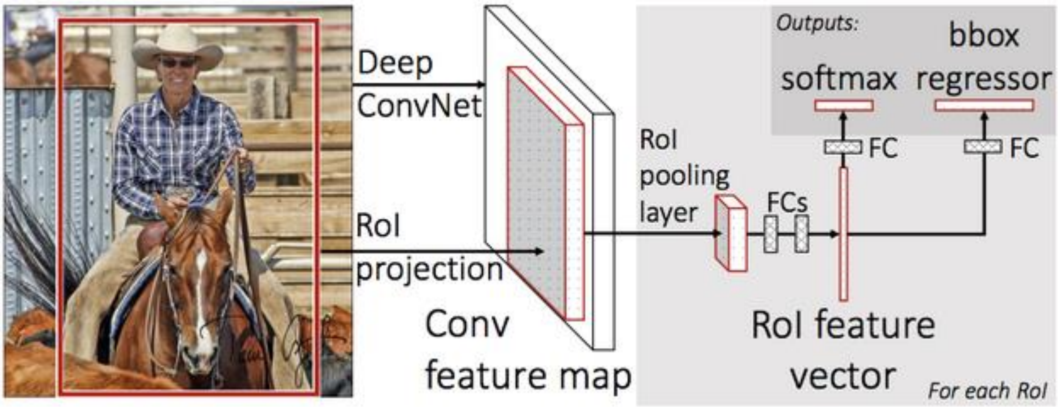


Figure 8. *Fast R-CNN workflow*

1.4.3. Faster-RCNN workflow and relative studies for lung nodule detection

Both R-CNN and Fast R-CNN use selective search to provide region proposals. Selective search is slow, time consuming for the algorithm performance and especially ,not being a trainable algorithm, it is not able to learn from the data.

In the solution proposed by Shaoqing Ren et al. (2017) selective search was replaced with a neural network dedicated to the selection of region proposals, known just as Region proposal Network (RPN). The inclusion of the RPN into the Fast R-CNN gave rise to the so-called Faster R-CNN.

Similar to Fast R-CNN, in the Faster R-CNN the image is provided as input to a deep convolutional network which extracts a feature map.

Instead of the application of selective search, the feature map is elaborated by the RPN that generate a set of region proposals. The aim of the RPN is to distinguish background regions from foreground ones. This purpose is done by sliding a window on the feature map and by processing singularly each feature map region inside the window. For each feature map region, the RPN returns two outputs related to multiple regions of the image: the first output is the probability of being foreground while the second consists in its coordinates in the image reference frame.

The particularity of the RPN training procedure is that for each region defined on the feature map, multiple regions on the input image are associated and labelled to a specific class (background or foreground). These predefined regions are known as “anchors”. To each feature map region, a fixed set of anchors is assigned, which are caracterized by different shape and size but all centered in the image reference frame.

The region on which is build the set of anchors is called “base anchor box” and corresponds directly to the feature map region analyzed by the RPN. The dimensions of base anchor box are equal to the stride that has two different values, one given from the ratio of the width and the other one coming from the ratio of the heights of image and feature map(Figure 9).

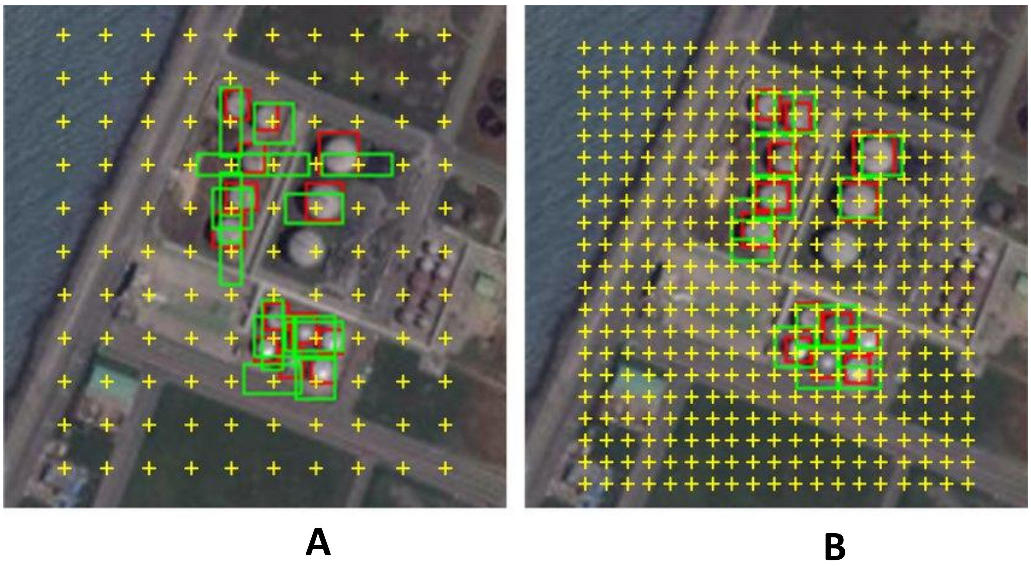


Figure 9. *An illustration of the arrangement of the anchors in the case of stride 16[a.u.](A) and stride 8[a.u.](B)*

The operations underlying the calculation of anchors can be summarized in few steps: each center of base anchor box constitutes the center point of k anchors with different scale and ratio respect to the original box. The original paper (Shaoqing Ren, 2017) set by default 3 scales and 3 aspect ratios, yielding so k= 9 anchors at each sliding window position(Figure 10).

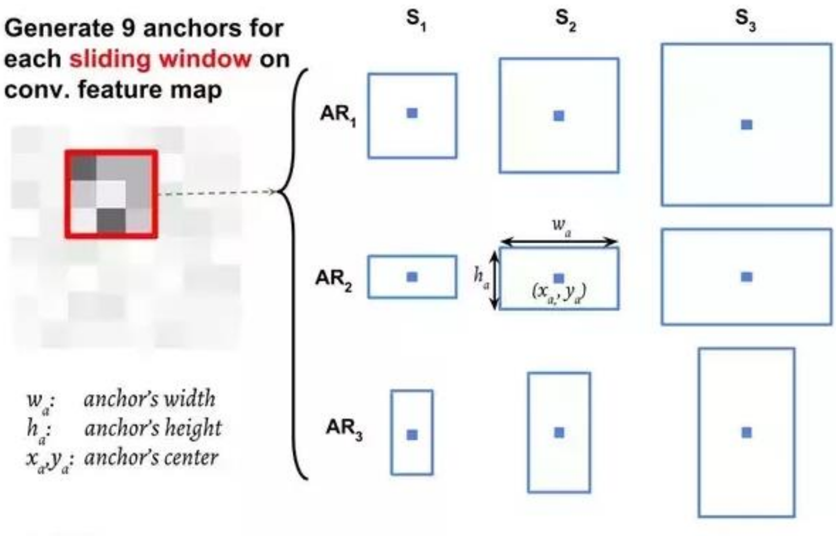

Generate 9 anchors for each sliding window on conv. feature map
S1
S2
S3
AR1
AR2
AR3
w_a
h_a
(x_a, y_a)
w_a: anchor's width
h_a: anchor's height
x_a, y_a: anchor's center

Figure 10. *Each sliding window position on the convolution feature map correspond to a group of k anchors built on the corresponding base anchor box of the image*

As already mentioned above, the RPN returns a series of region proposals associated with a probability of including or not an object of interest along with the etimate of the position of these regions. RPN does not share any information about which class the foreground refers to but it only aims to establish if the region could contain the object of one of the multiple classes.

Proposals coming from RPN, are then used by the region-based object detection CNN represented by the Fast R-CNN like in the previous work(Figure 11).

Specifically, in Fast R-CNN the proposals generated by RPN are extracted from the feature map generated by feature extractor, warped into squares by means of ROI pooling layers and finally input to the dense layers, following the same procedure of Ross Girshick et al. work (2015) (section 1.4.2). Fast R-CNN for each proposal return two outputs which consist, as in RPN, in the following classification and regression variables respectively: the probability related to the different class and the adjusted coordinates in order to better fit the object of interest. In contrast to RPN, Fast R-CNN give information about which object the proposals actually contain ,including the background class.

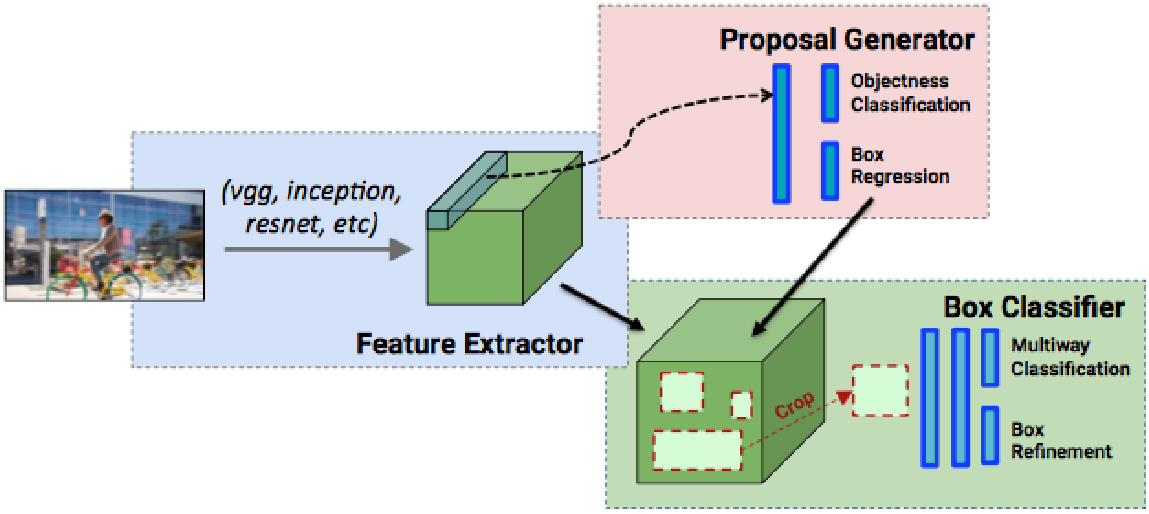


Figure 11. *Faster R-CNN workflow*

The training of Faster R-CNN is certainly the trickiest part. The easiest way to train RPN and Fast R-CNN is to train independently the two networks. However, more sophisticated implementations allow the sharing of convolutional layers between the two networks, approach that should improve the performance of the overall Faster-RCNN architecture. Different methodologies have been already proposed in literature such as alternative training (Appendix B), approximate joint training and non-approximate joint training.

In the last years, many researchers work on Faster R-CNN for lung nodule detection, introducing some important innovations and changes.

In the implementation of CAD system for lung nodule detection, Faster R-CNN start to be juxtaposed to a preprocessing part and followed by a false positive reduction part, usually implemented with a 2D-CNN.

Following this approach ,Xia Huang et al. proposed a Faster R-CNN CAD system that incorporates several steps: Faster RCNN was trained by means of four step alternative training (Appendix B) according to the original implementation (Shaoqing Ren, 2017), followed by a merging operation which fuses overlapping candidates (obtained from Faster R-CNN) combining 2D patches with close Euclidean distances. A traditional three-layer 2D CNN based FP reduction further eliminated FPs and finally a modified FCNs computed nodule segmentation (Xia Huanga, 2019).

A very important innovation in the structure of the network was implemented by Jia Ding et al. that proposed a Faster R-CNN based on deep convolutional neural network(DCNN), introducing a deconvolutional structure to the classical structure of Faster RCNN for lung nodule detection on 2D slices. Then a three-dimensional DCNN was added for false positive reduction step. The distinctive feature of this project is the addiction of one deconvolutional layer at the end of backbone neural network to solve the issue related to the small size of lung nodule. (Jia Ding A. L., 2017)

Yanfeng Li et al. (2019)proposed a Faster R-CNN method based on deep learning for thoracic MR image. They aimed to replicate the results for CT-scans for a different diagnostic image(MRI). The advantage is that MRI is a non-radiation examination and can provide not only morphological information but also functional information. The proposed method is pretty the same and consists in a Faster RCNN followed by false positive reduction step. This study shows that is possible to obtain good result in term of sensitivity and false positive even with thoracic MRI images.

1.5. Open issues and aim of the project

CAD systems are fundamental to improve the accuracy of the diagnosis and assist the radiologist in early detection, aspect that becomes fundamental in case of lung cancer prevention.

Different studies obtained great results from the employment of CAD systems, but many problems and obstacles need to be overcome to allow their introduction in the clinical practice and give a real support to radiologists (Macedo Firmino, 2014).

An open issue in CAD system for lung nodule detection is related to the identification of GGO nodules: characterized by not-defined boundaries and low contrast with respect to the background, these lesions are often missed by detection model and representing a risk in automatic detection being the likelihood of malignancy of this particular nodule higher than that of solid nodules.

Beside GGO nodules, false positives are surely the main challenge for CAD system and are also the reason why CAD system are still not used as standalone tools in the clinical practice .

This work aims at creating an automatic detection model for pulmonary nodules able to identify the position of suspected lesions in low-dose CT (LDCT) screening. To reach this task a Faster-RCNN has been implemented and preliminary optimized. For the feature extraction part, a pre-trained network was used and its architecture adapted according to the dataset available. For the detection part, only the ROI pooling layer, dedicated to size normalization, was exploited from an architectural point of view, whereas many experiments were performed to identify the more efficient detection model. Additionally, the behavior of the implemented network on different nodule types was evaluated.

This work was conducted within a collaboration with the Istituto Europeo di Oncologia, which provided the LDCT dataset collected during the COSMOS study.

Chapter 2. Materials and methods

In this chapter, materials and methods adopted for the Faster R-CNN implementation are explained. In particular we report the modification applied to the original implementation of the network (Shaoqing Ren, 2017). The proposed CAD following the typical CAD workflow presented in Section 1.3.

2.1. Dataset

For the development of the presented work, a subset of the COSMOS study was considered. From the cohort of 286 patients a total number of 639 CT-scans was collected. Different CT-scans could be therefore associated to the same subject according to the fact that we are dealing with a longitudinal data. Only CT scans with similar acquisition parameters were included in our subset. Specifically, 639 cases were acquired with a kVp and mA respectively equal to 120 and 30 while a standard kernel was applied to reconstruction all the series. Images resolution on the axial plane ranged between 0.5234 and 0.9160 mm while slice thickness is always equal to 2.5 mm.

On each of the collected LDCT scan, at least a pulmonary nodule was previously identified by an expert radiologist and the same could be present in multiple scans. Overall, a set of 639 pulmonary nodules was available and, for each one, we disposed of a binary mask defined through manual segmentation by an expert radiologist.

In a first analysis a subset of 639 CT-scans was used for the development of Faster R-CNN network. In particular, a subset of 500 CT-scans (80%) was used for training and 125(20%) CT-scans for validation.

Among our set of data, pulmonary nodules belonging to the three class of non-calcified nodules were included: solid, part-solid and non-solid nodules. The majority of the cases were solid nodules (286) while the number of part-solid and non-solid was respectively equal to 149 and 197. Considering that the implemented model works in 2D, cases of solid, part-solid and non-solid nodules in terms of number of CT slices were equal to 1380, 642 and 902 respectively. Figure 12 summarizes how the dataset is subdivided in terms of nodule type and as can be noted, for few isolated samples (31), the texture was not known. Additionally, 8 samples of calcified nodule was also present.

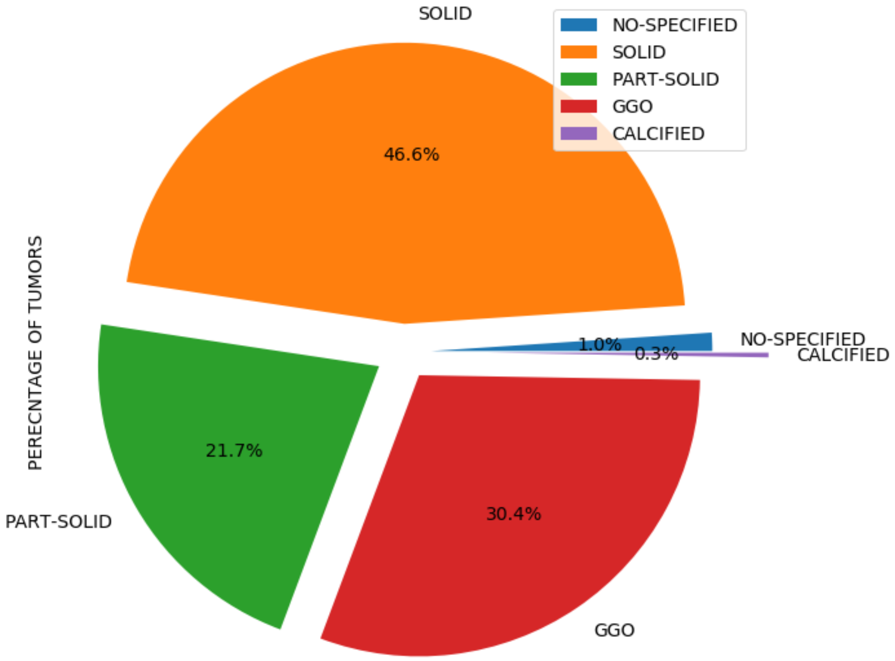


Figure 12. *percentage of different type of tumors*

In Figure 13 are shown three different nodules. Specifically, a solid nodule with its typical homogeneous soft-tissue attenuation is reported in Figure 13A; figure 13C shows an example of non-solid nodule characterized by an hazy increase in local attenuation of lung parenchyma not obscuring the underlying bronchial and vascular structures; in the central panel (Figure 13B) a lesion with mixed solid and non-solid components is finally reported, i.e. a part-solid nodule.

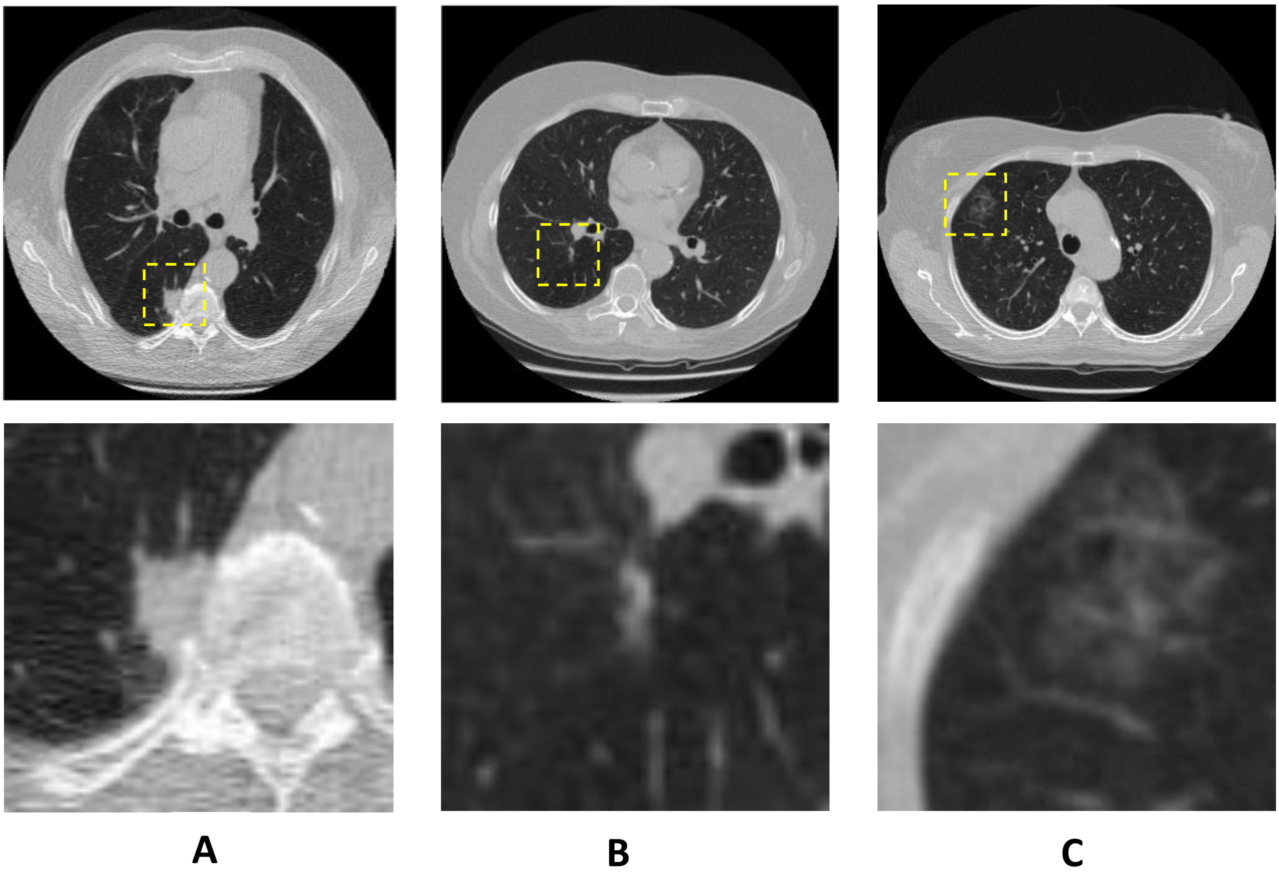


Figure 13. *An illustration of solid (A), part-solid(B) and GGO(C) tumors*

The slices considered are int16(pixel's values range from [-32768, 32767]) grayscale images of 512x512 without any type of preprocessing previously applied.

2.2. Preprocessing

In the field of automatic object detection, to improve the training process of the network the input image can be processed to limit possible confounding factors. In case of pulmonary nodule detection, the region we are interested to inspect is the lung parenchima. However, other organs and anatomical structures are always present in a thorax CT scan.

Some of them may have spherical shapes and intensity value very similar to a lung nodule representing therefore a possible confounding factor that can negatively influence the behavior of the model. In order to limit this problem, the original CT scans were processed according to an algorithm proposed in the literature in the field of pulmonary nodule detection based on deep neural networks (Fangzhou Liao, 2015). After a preliminary conversion of the image from Housfield Units (HU) to UINT8 values, a clipping operation was applied to limit the gray scale image values within the range [-1200, 600]. A linear transformation was then applied to compress the range of values within 0 and 255.

To apply the last step of the processing algorithm, lung binary masks were needed to define the region of the lungs in which we can find pulmonary nodules. For each LDCT scan, the lung segmentation approach proposed by Yashin Dicente et al. (2015) et al was applied, by taking advantage of the online available application(http://publications.hevs.ch/index.php/publications/show/1871).

The pre-processed image was then multiplied by the lung mask and a value equal to 170 was assigned to all the voxels outside the lung's parenchima. In addition, all values greater than 210 (high-luminance) were replaced with 170, to avoid the inclusion of some areas surrounding the lung mask previously defined. These areas contain generally bones (the highest luminance tissues), that are easily misclassified as calcified nodules (also high-luminance tissues).

LDCT were processed also to a dimensional point of view. To homogenize the resolution of different CT scans, a resampling was performed by fixing the value of pixel in the axial plane at 0.5 mm on both directions. This step allowed also to increase the dimension of the input image to 683x683 pixels. Finally lungs areas(obtaining by the multiplication of the original image and lung mask) were centered in the image . In Figure 14, a comparison between an original CT slice and a slice after the preprocessing procedure is reported.

Unlike other works (Qiang Li) on lung nodule detection, dot filtering techniques were not applied to the image, keeping the difficulty in the distinction between nodule and tissue like vessel or bronchi, that are often eroded by filtering.

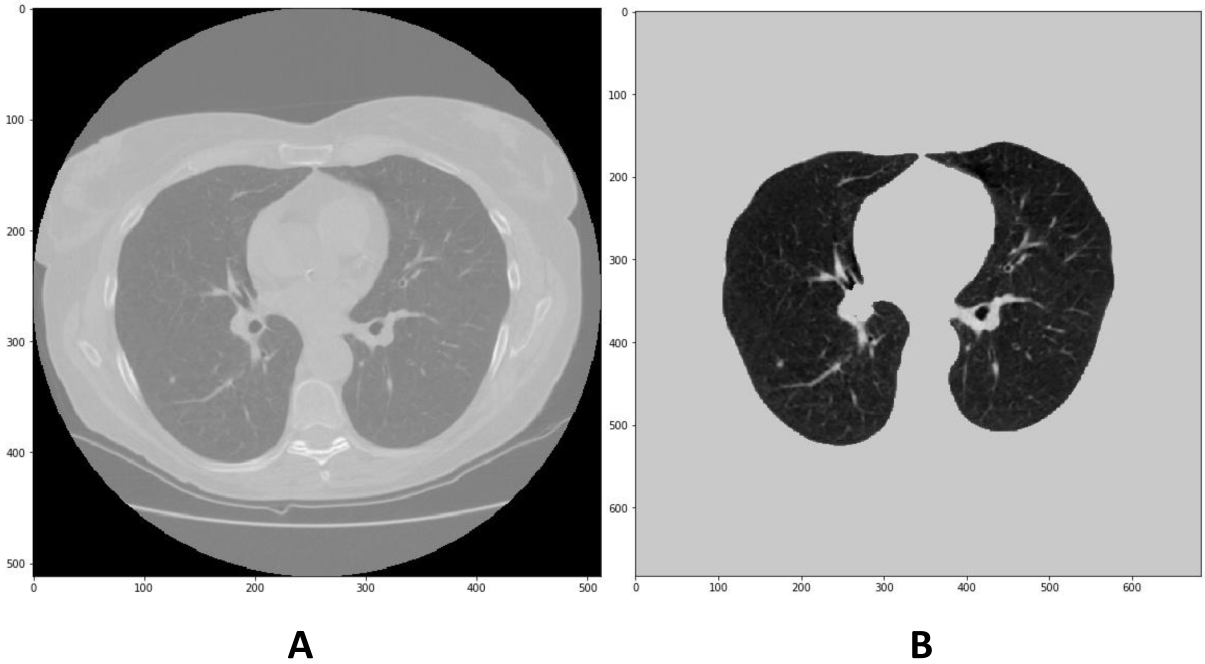


Figure 14.*CT image before(A) and after(B) preprocessing procedure*

2.3. Detection network implementation

The purpose of a nodule detection model is to generate candidate bounding boxes for suspicious nodules.

Faster R-CNN is proposed in this work for pulmonary nodule detection, being the network architecture with highest performance in object detection tasks (Shaoqing Ren, 2017).

Faster R-CNN network consists of two blocks (Figure 15): the Region Proposal Network (RPN) and the Fast R-CNN. RPN and Fast R-CNN share the same stack of CNN layers which takes as input the gray-scale image to elaborate it and extract a set of feature maps. These CNN layers dedicated to feature extraction constitute the so-called backbone neural network, in our work represented by the VGG16. In our implementation VGG16 was trained independently and therefore it kept weights fixed during the training of RPN and Fast R-CNN. For these reasons VGG16 was considered like a separate structure from RPN and Fast R-CNN, communicating in the same way with both networks (Figure 15).

The RPN was fed with the output feature maps of the backbone network with the aim to propose candidate regions in which the pulmonary nodule could be present. Finally, Fast R-CNN, starting from the proposals of RPN and the feature map coming from VGG16, provided the class whose proposals belong to and also the coordinates adjusted to better fit the object of interest.

In the following sections a detailed description of each part of the architecture is reported.

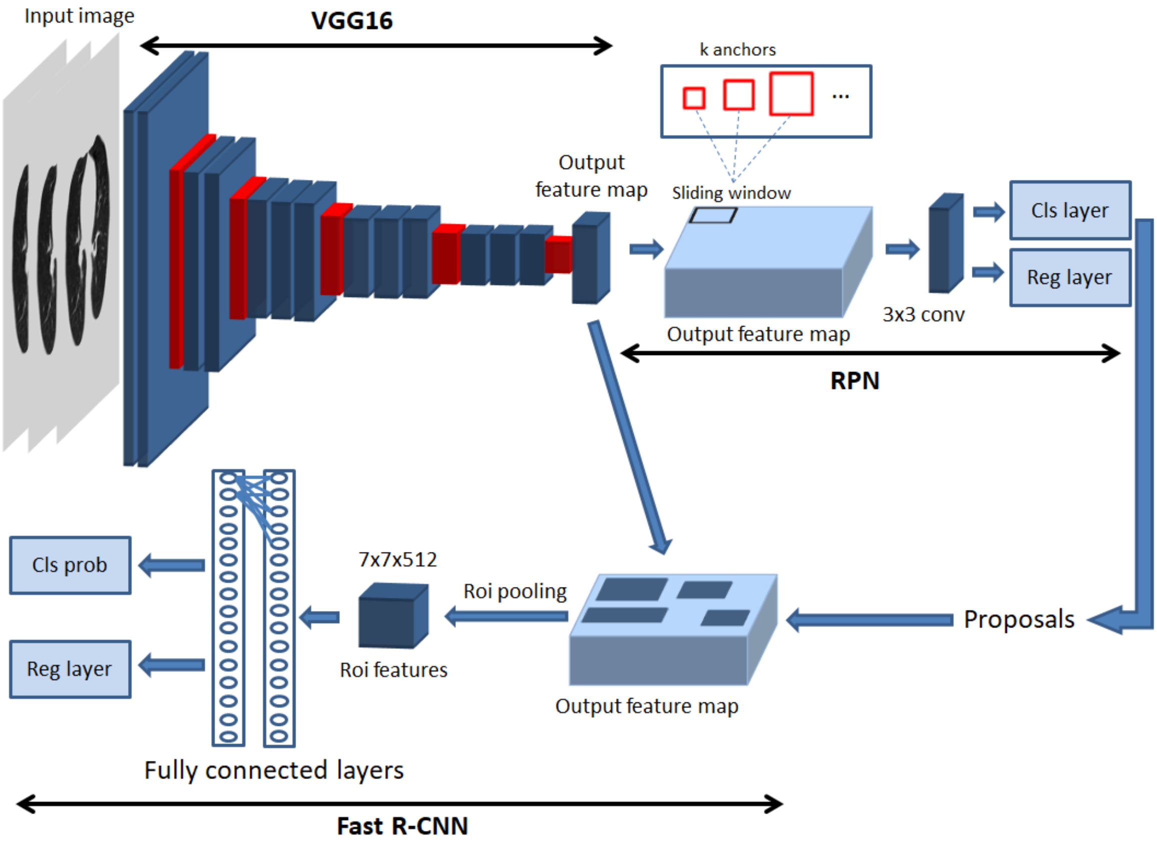


Figure 15. *Architecture of Faster R-CNN implemented for lung nodule detection. The network can be divided in turn in three networks: VGG16, RPN and Fast R-CNN.*

2.3.1. VGG16

Different backbone networks have been tested in literature to support the Faster-RCNN. Among these, VGG16 is one of the architectures more frequently used.

VGG16 was in origin designed for the classification of 1000 different classes of the ImageNet dataset. This network can be considered composed by two parts: feature extractor, represented by five convolution blocks, and the fully connected layers needed for the classification purposes. In the presented work, we took advantage of the original feature extraction architecture of VGG16 where each one of the five convolution blocks is composed by both convolutional and max pooling layers creating an overall set of eighteen layers (Figure 16)(Appendix A) .

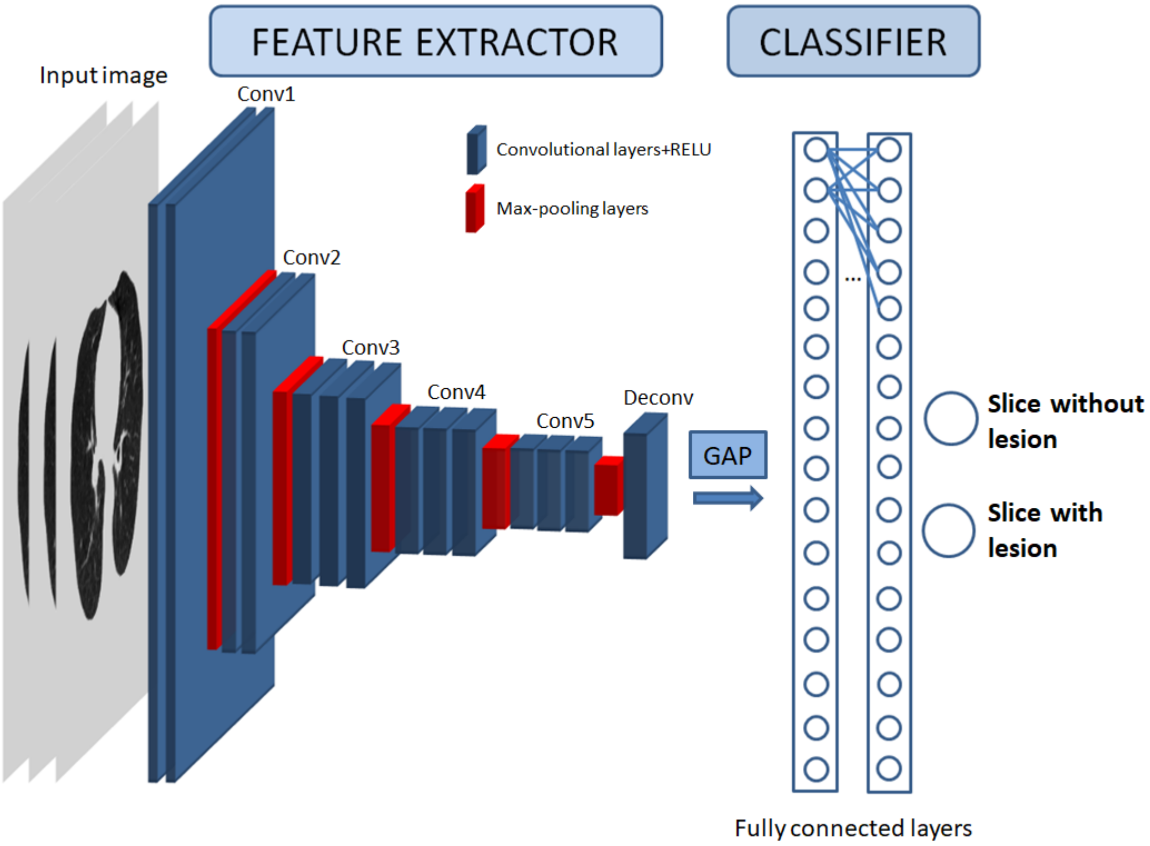
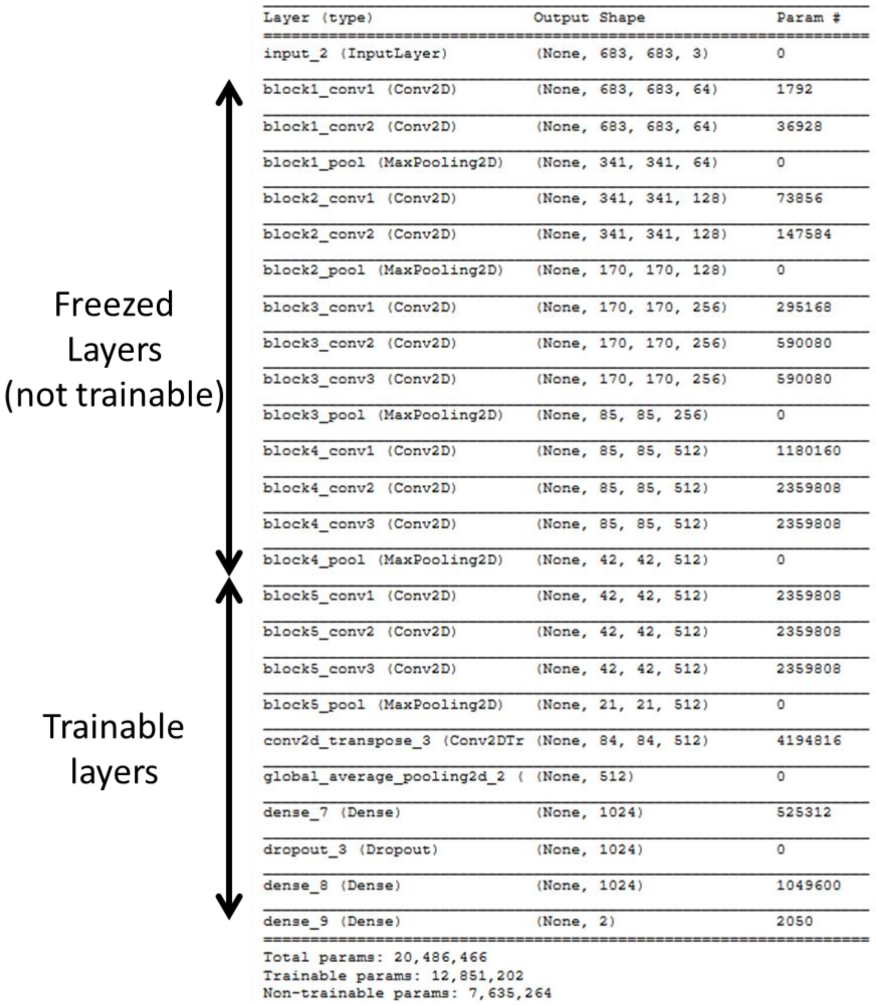


Figure 16. *Input image is concatenated three times in order to be compatible with the input of VGG16, that is composed by a feature extractor part and a classifier part.*

The second part of VGG16 was instead modified to (i) reach an architecture more suitable to small-objects feature extraction, such as in the case of lung nodule detection, and (ii) adapt the classification output to the two classes of interest.

For what concerns point (i), a deconvolutional layer was added at the end of the convolution blocks setting the kernel, stride and padding sizes equal to 4, 4 and 2, respectively. The number of output feature maps was instead fixed to 512 (Figure 16). The introduction of the deconvolutional layer lead to the recovering of more fine-grained features (Jia Ding A. L., 2017) which are fundamental to detect small objects. Therefore, we considered a feature map of dimensions (84,84,512) instead of (42,42,512), i.e. dimension of the output feature map in the original architecture. Indeed, being the original VGG16 architecture established to extract features from objects of higher size with respect to pulmonary nodules, the original-architecture feature map is not able to explicitly depict the features of nodules and consequently poor detection performance would be reached. Section 2.3.2 in which RPN is introduced, clarifies the reasons behind the inclusion of a deconvolutional layer.

The other main modification applied, is related to the classification part of the net (point (ii)) which is needed to allow the network training on our dataset where the two classes of interest are only two: *slice with lesion* and *slice without lesion*. Specifically, after the deconvolutional layer, a global averaging pooling layer was inserted, followed by 2 dense layers (1024 units) and finally a 2 unit softmax layer according to the number of classes considered (Figure 17).



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_2 (InputLayer)         (None, 683, 683, 3)       0
_________________________________________________________________
block1_conv1 (Conv2D)        (None, 683, 683, 64)      1792
_________________________________________________________________
block1_conv2 (Conv2D)        (None, 683, 683, 64)      36928
_________________________________________________________________
block1_pool (MaxPooling2D)   (None, 341, 341, 64)      0
_________________________________________________________________
block2_conv1 (Conv2D)        (None, 341, 341, 128)     73856
_________________________________________________________________
block2_conv2 (Conv2D)        (None, 341, 341, 128)     147584
_________________________________________________________________
block2_pool (MaxPooling2D)   (None, 170, 170, 128)     0
_________________________________________________________________
block3_conv1 (Conv2D)        (None, 170, 170, 256)     295168
_________________________________________________________________
block3_conv2 (Conv2D)        (None, 170, 170, 256)     590080
_________________________________________________________________
block3_conv3 (Conv2D)        (None, 170, 170, 256)     590080
_________________________________________________________________
block3_pool (MaxPooling2D)   (None, 85, 85, 256)       0
_________________________________________________________________
block4_conv1 (Conv2D)        (None, 85, 85, 512)       1180160
_________________________________________________________________
block4_conv2 (Conv2D)        (None, 85, 85, 512)       2359808
_________________________________________________________________
block4_conv3 (Conv2D)        (None, 85, 85, 512)       2359808
_________________________________________________________________
block4_pool (MaxPooling2D)   (None, 42, 42, 512)       0
_________________________________________________________________
block5_conv1 (Conv2D)        (None, 42, 42, 512)       2359808
_________________________________________________________________
block5_conv2 (Conv2D)        (None, 42, 42, 512)       2359808
_________________________________________________________________
block5_conv3 (Conv2D)        (None, 42, 42, 512)       2359808
_________________________________________________________________
block5_pool (MaxPooling2D)   (None, 21, 21, 512)       0
_________________________________________________________________
conv2d_transpose_3 (Conv2DTr (None, 84, 84, 512)       4194816
_________________________________________________________________
global_average_pooling2d_2 ( (None, 512)               0
_________________________________________________________________
dense_7 (Dense)              (None, 1024)              525312
_________________________________________________________________
dropout_3 (Dropout)          (None, 1024)              0
_________________________________________________________________
dense_8 (Dense)              (None, 1024)              1049600
_________________________________________________________________
dense_9 (Dense)              (None, 2)                 2050
=================================================================
Total params: 20,486,466
Trainable params: 12,851,202
Non-trainable params: 7,635,264
```

Figure 17. VGG16 architecture for training

Global average pooling (GAP) layers was used in order to minimize overfitting by reducing the total number of parameters in the model. Similar to max pooling layers, GAP layers are used to reduce the spatial dimensions of a three-dimensional tensor. However, GAP layers perform a more extreme type of dimensionality reduction, where a tensor with dimensions 84×84×512 is reduced in size to have dimensions 1×1×512. GAP layers reduced each 84×84 feature map to a single number by simply taking the average of all 84 values (Figure 18). The importance of GAP was stressed by MIT researcher (olei Zhou) that demonstrated that CNNs with GAP layers (a.k.a. GAP-CNNs), originally trained for a classification task, can also be used for object localization.

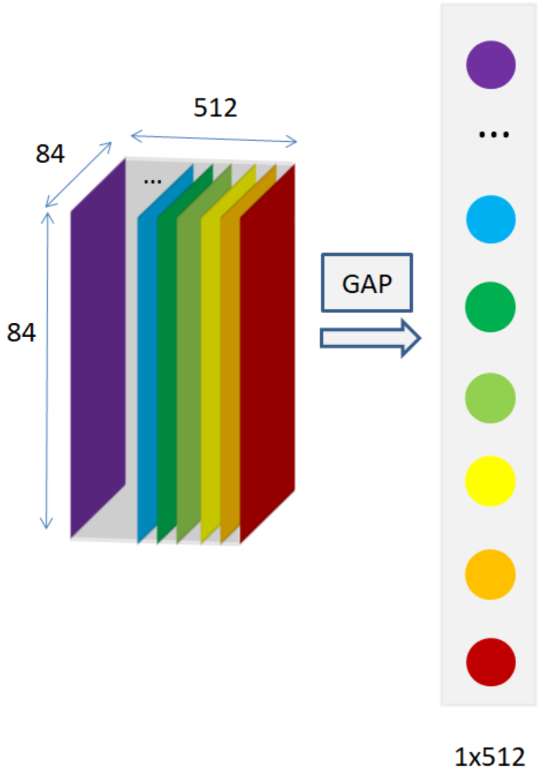


Figure 18. *GAP layers dimensionality reduction: each channel of the feature map is reduced to a single neuron*

Deep learning models generally require large datasets to train properly and to avoid overfitting. VGG16, to reach good performances, was trained using 1.2 million images from ImageNet dataset. Unfortunately, in the medical field, a similar amount of labelled images does not exist, and therefore is very difficult to reach a good representation of the data by training the network from scratch. For these reasons, the use of pre-trained weights along with the application of transfer learning is a commonly adopted approach to train deep learning models for clinical support.

According to these limitations, we applied transfer learning by initializing the VGG16 weights with those of a VGG16 pre-trained on the ImageNet (Chao Tong, 2019).

Weights of the first 15 layers were kept fixed, indeed is well known that in the first layers, image features with a lower level of abstraction, such as edges, are usually extracted, i.e. features that are common to any type of image. Fine-tuning was instead applied to the last convolution block, to adapt the original weights so to reach a good representation of the new dataset (Thakur).

The training process was carried on using a Stochastic Gradient Descent optimizer (SGD) with momentum equal to 0.9 while the learning rate was fixed to 0.001. Categorical cross-entropy was used as loss since the output of the model is categorical. The model was trained for 100 epochs using a batch size of 16 images As final model, the one at the epoch characterized by lowest validation loss was chosen.

With respect to RGB images that are represented by three channels, CT scans, being gray-scale images, are associated to a single channel. Although VGG16 architecture was designed to deal with RGB images, a replicate of the 2D gray scale image was used to reach consistent dimensions with the number of channels and therefore to allow the weights transfer.

The image taken from the deconvolutional layer represent the output of the feature extractor network and it was fed into both region proposal network and object detection network (Fast R-CNN). As such, the choice of the addition of the deconvolutional layer is crucial for the performance of the network considering our limited dataset.

2.3.2. Region proposal network

The implementation of RPN and Fast R-CNN have been done following the work of Shaoqing Ren (2017) and two reference codes from github(https://github.com/dongjk/faster_rcnn_keras, https://github.com/you359/Keras-FasterRCNN).

Once the gray scale image was processed by the backbone network, from the output feature map a set of region proposals was generated through the RPN. Specifically, RPN was applied by sliding a window over the feature map where each pixel was connected to a region of the original image by a correspondence which was defined through a set of “anchors”.

The smallest regions on the input image which have a correspondence with a single pixel of the feature map, are called “base anchor box”. The dimension and shape of the base anchor boxes limit the minimum size of the detectable object and are determined by the overall stride applied in the backbone network. This parameter needs therefore to be adjusted to reach anchors dimensions that better fit pulmonary nodules sizes.

The overall stride of the backbone network can be calculated as the ratio between the dimension of the original image and the dimension of the deconvolutional layer feature map taken as output.

$$stride_x = \frac{image_{width}}{feature\ map_{width}} = base\ anchor\ box_{width}$$

$$stride_y = \frac{image_{height}}{feature\ map_{height}} = base\ anchor\ box_{height} \quad (1)$$

Where

- $image_{height}$ and $image_{width}$ represent respectively image height and width, both equals to 683.
- $featuremap_{height}\ and\ featuremap_{width}$ represent respectively deconvolutional layer feature map height and width, both equals to 84.

Knowing the stride, a grid was defined on the input image where each cell corresponded to a base anchor box with height and width equal to the stride along x and the stride along y, respectively.

In our implementation, squared input images of 683x683 pixels were considered and same stride ratio along the two directions was applied, therefore base anchor boxes were squared too(Figure 19).

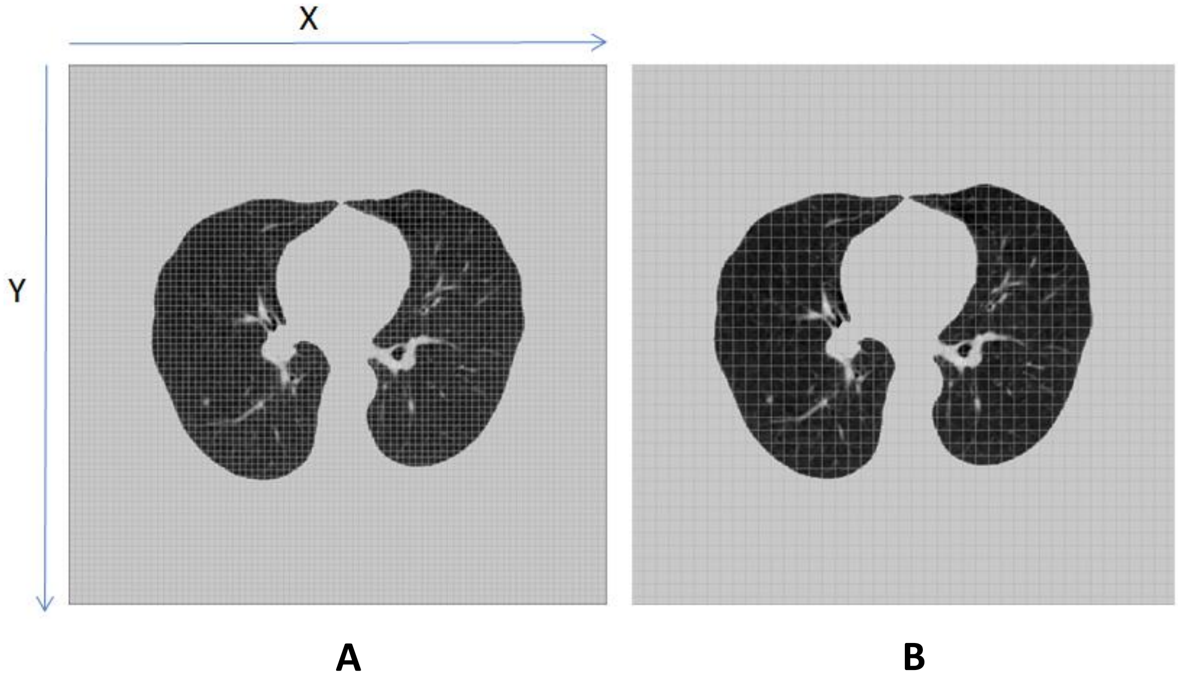


Figure 19 An illustration of the change of base anchor box according to a stride value of 8[a.u.](A) and 16[a.u.](B). It is also represented the image reference frame with axis X and Y.

Figure 19 shows a comparison between grids of base anchor boxes derived from a feature map associated with a stride equal to 8[a.u.] (Figure 19A) and equal to 16[a.u.] (Figure 19B), respectively.

The dimension of the base anchors is therefore directly proportional to the stride and both are inversely proportional to the size of the feature map as described in Equation (2).

$$stride \downarrow\downarrow \quad baseanchor \downarrow\downarrow \quad featuremapdimension \ \uparrow\uparrow$$

$$stride \uparrow\uparrow \quad baseanchor \uparrow\uparrow \quad featuremapdimension \ \downarrow\downarrow \tag{2}$$

A lower size of the base anchors allows to detect smaller pulmonary nodules; therefore, the more suitable base anchor size could be reached considering a larger feature map. However, if we do not modify the original architecture of the VGG16, the only way to obtain a larger feature map is taking it from a less deep convolution block, losing the information elaborated in the subsequent layers.

To overcome this problem, as said in section 1.4.3, a deconvolutional layer was introduced to decrease the stride value from the usual 16[a.u.] (VGG16) to 8[a.u.] without losing information. An ulterior addition of deconvolutional layer was avoided, in fact it will increase drastically the number of parameters inside the neural network, increasing the risk of overfitting even with GAP layers.

Once we defined the base anchors, for each one of these, a set of 9 additional anchors with common center was defined (Figure 20). Considering that lung nodules have approximately a circular shape, we chose to exploit anchors dimensions which differ each other only in terms of scale while the ratio parameter was kept constant and equal to 1:1. Nine different scales were therefore applied to the base anchor box. The set of scales was established in order to maximize the number of anchors overlapping with the nodule binary masks, whose maximum bounding box dimension was equal to 74 pixels. The optimum set of 9 scales was found in the range between 1:2 and 1:10.

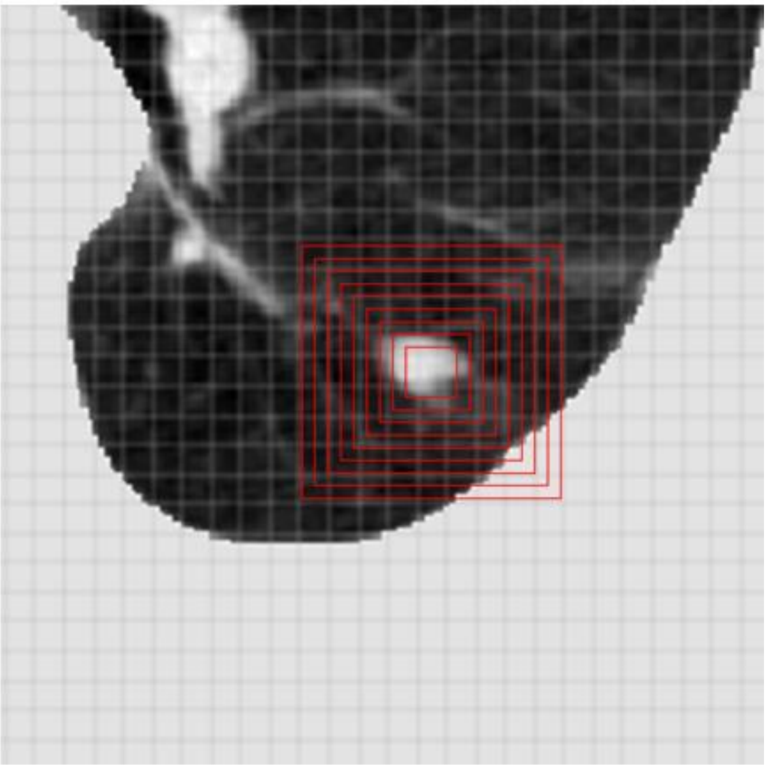


Figure 20. *In red are represented k=9 different scale anchors(from1:2 to 1:10) built on the same base anchor box. Note that all anchors share the same center.*

Having computed 9 anchors for each base anchor, the overall set of anchors was equal to 84x84x9 (63504) where 84x84 pixels is the dimension of our feature map. It is evident that a lower stride lead to an increase of the number of anchors and consequently to a higher computational load.

As such, small object detection requires a low stride and so smaller base anchor boxes. Indeed, keeping a lower anchor scale in the case of high value of stride do not solve the problem: all different size anchors were centered on the same square base anchor box and if an object is positioned on the border of the square, its detection results impossible(Figure 21). No methods present in the literature use anchor scale ratio lower than 1:1.

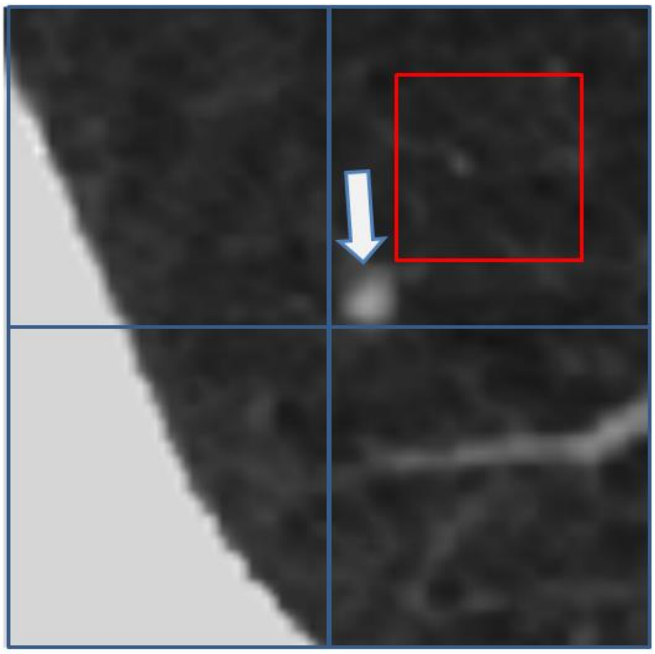


Figure 21. *Also the use of scale ratio lower than 1:1 does not allow the detection of small nodules localized at the boarder of base anchor box*

Anchors that overflow from the boarder of CT scan were excluded according to the original paper (Shaoqing Ren, 2017), since it is improbable that a boarder anchor includes a pulmonary nodule having centered the lungs in the preprocessing step.

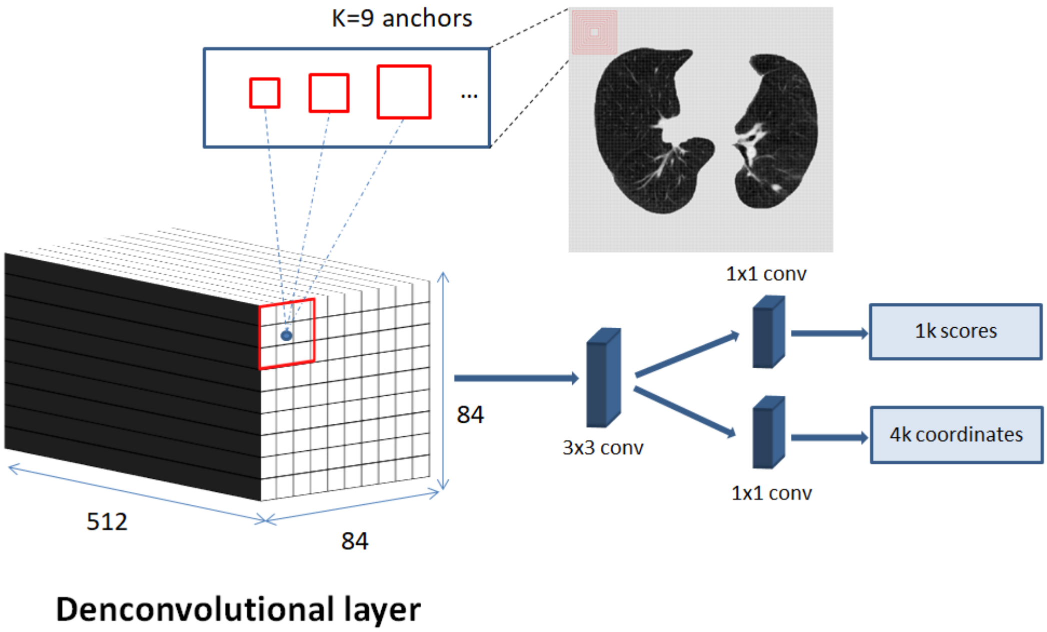


Figure 22. Illustration of the architecture of RPN and of the relationship between anchors and output feature map: each group of k=9 anchors correspond to a pixel of the feature map, that is the center of 3x3x512 input block of RPN.

Figure 22 shows the architecture of the RPN network which was not modified with respect to the original implementation.

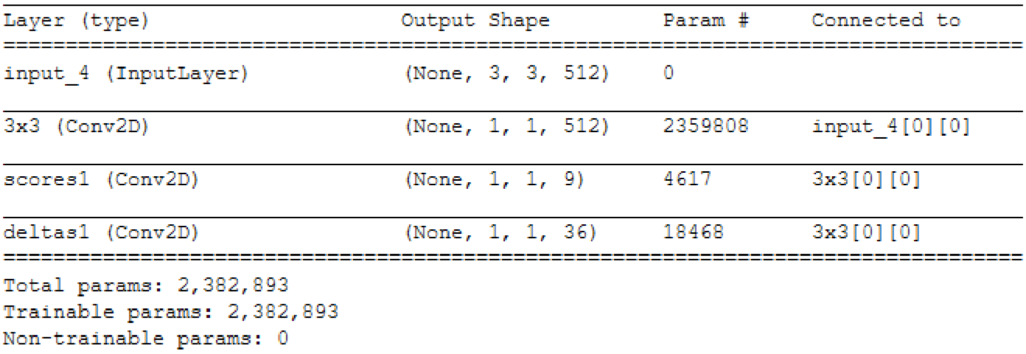
As already mentioned, a window of 3x3 and with depth of 512 channels, slid on the feature map to define a new input for the RPN. The input of the RPN consists therefore in a (3,3,512) block of feature map. In order to consider also the pixel on the boarder of feature map a padding operation was applied.

The first layer is a 3x3 convolutional layer characterized by 512 channels (with RELU activation function), since VGG16 was used as feature extractor network (Shaoqing Ren, 2017).

The first convolution is followed by two parallel 1x1 convolutional layers(Figure 23):

- the first one returns an output vector of 1*9 units, which represent the predicted probabilities of the presence of the object of interest in the anchor set. Specifically, each one of the 9 probabilities is associated to a specific anchor of the same set, i.e anchors with common base anchor. Anchors with highest probabilities will be classified as foreground while as background in the other cases.

- the second output returns an output vector of 4*9 units, which represent the predicted transformed coordinates of the box which should contain the object of interest.



```
Layer (type)                     Output Shape          Param #     Connected to
==================================================================================
input_4 (InputLayer)             (None, 3, 3, 512)     0
__________________________________________________________________________________
3x3 (Conv2D)                     (None, 1, 1, 512)     2359808     input_4[0][0]
__________________________________________________________________________________
scores1 (Conv2D)                 (None, 1, 1, 9)       4617        3x3[0][0]
__________________________________________________________________________________
deltas1 (Conv2D)                 (None, 1, 1, 36)      18468       3x3[0][0]
==================================================================================
Total params: 2,382,893
Trainable params: 2,382,893
Non-trainable params: 0
```

Figure 23. RPN network architecture

Considering the nature of the two different outputs, the training of the RPN consists in the optimization of a classification and a regression problem. Two different loss functions needed therefore to be minimized.

The classification loss L_{cls} is a function of the predicted probability p and the target class p^* which is equal to 1 for anchors that belong to the foreground and equal to 0 for anchors of the background. Binary cross entropy was used as classification loss, therefore, for an anchor i, L_{cls} can be calculated as follows:

$$L_{cls}(p_i, p_i^*) = \quad -\left(p_i^* log(p) + (1 - p_i^*) log(1 - p) \right) \qquad (3)$$

Because the anchors target class p^* is not known a priori, it needs to be defined. Knowing the ground truth box (i.e. minimum squared box that contains the lesion), Intersection over Union (IoU),that is the ratio between the area of overlap and area of union between two boxes, between each anchor and the ground truth box was calculated. Two IoU thresholds were then applied: anchors with IoU above the upper threshold (th_{up}) were considered as foreground ($p* = 1$) while anchors with IoU below the lower threshold (th_{low}) will be labelled as background ($p* = 0$).Values of th_{up} and th_{low} were fixed to 0.5 and 0.02 respectively, according to an RPN implementation presented in literature for pulmonary nodules detection (Broyelle, 2018).

Anchors resulted with a IoU between th_{up} and th_{low} were discarted to avoid the insertion of confounding anchors. These anchors are called "hard negative" and they will become central in the implementation of Fast R-CNN.

For some images no foreground anchors were found, and it happens when the condition $IoU < th_{up}$ is true for the overall set of anchors. In order to include these images, Shaoqing Ren et al. introduced a second condition assigning positive label also to the anchor with the highest intersection-over union (IOU) overlap with a ground-truth box. However, in a small object detection problem, it happens frequently that no anchors have IOU>0.5 and, for this reason, this condition was not implemented, avoiding the identification of anchors with very low IoU as positive label.

In order to avoid having an RPN biased for a specific class, we need to deal with the class imbalance problem; indeed, for each image sample, the number of negative anchors was always >> of the number of positive anchors. A random subsampling of negative anchors was therefore applied: a number of negative anchors equal to $2 * N_{anc\_pos}$ was considered for each image sample according to the reference code, where N_{anc_pos} is the number of positive anchors. Following the reference code, the batch size was fixed to 512, including more than one image in the batch, always maintaining the same balance. Only in a second time we referred the batch to a single image following the so called "*image-centric sampling strategy*" of the original paper (Shaoqing Ren, 2017).

Similarly, to L_{cls}, the regression loss L_{rgs} is a function of a prediction t and a target term t^*. Since we want to predict the position of the squared box that better fits the object of interest, 4 are the variables to be estimated:

- x and y, that represent the center coordinates of the box.
- h and w, that represent the height and the width of the box.

Likewise, the ground truth box is represented by 4 target variables x^*, y^*, h^* and w^*. However, to properly maximize the similarity between predicted and target box, we have also to consider their dependence to the anchor box. For this reason, the predicted box and the target box coordinates are parameterized as a function of the anchor box coordinates as reported in the following transformations:

$$
\begin{gathered}
t_x = \frac{x - x_a}{w_a}, t_y = \frac{y - y_a}{h_a}, \\
t_w = \log\left(\frac{w}{w_a}\right), t_h = \log\left(\frac{h}{h_a}\right), \\
t_x^* = \frac{x^* - x_a}{w_a}, t_y^* = \frac{y^* - y_a}{h_a}, \\
t_w^* = \log\left(\frac{w^*}{w_a}\right), t_h^* = \log\left(\frac{h^*}{h_a}\right),
\end{gathered}
\tag{4}
$$

Where:

- t_x/t_x^* and t_y/t_y^* denote the transformed center coordinate of the predicted/ground truth box;
- t_w/t_w^* and t_h/t_h^* are the transformed width and height of the predicted/ground truth box;
- x_a, y_a, w_a and h_a are finally the 4 variables that describe the anchor position in the image;

Both x^*, y^*, h^*, w^* and x_a, y_a, w_a, h_a were derived from upper-left corner(x_{min}, y_{min}) and lower-right corner(x_{max}, y_{max}) coordinates of the corresponding box.

L_{rgs} is therefore a function of the trasnsformed coordinates of the predicted box (t) and the transformed coordinates of the ground truth box (t^*). The adopted regression loss was the Huber loss, which is a piecewise function that for an anchor i can be calculated as follows:

$$L_{rgs}(t_i, t_i^*) = \begin{cases} \sum_{c=1}^{4} \frac{1}{2}(t_{i,c} - t_{i,c}^*)^2 = \sum_{c=1}^{4} \frac{1}{2}(dist_{i,c})^2 & , \quad |dist_{i,c}| < \delta \\ \sum_{c=1}^{4} \delta|t_{i,c} - t_{i,c}^*| - \frac{1}{2}\delta^2 = \sum_{c=1}^{4} \delta|dist_{i,c}| - \frac{1}{2}\delta^2 \ , & \quad otherwise \end{cases} \tag{5}$$

where

$$c = 1, . . , 4 = x, y, w, h$$

While δ is a distant threshold which in our case was left to its default value ($\delta = 1$).
To minimize in parallel L_{cls} and L_{rgs}, the two losses in each batch were merged as follows:

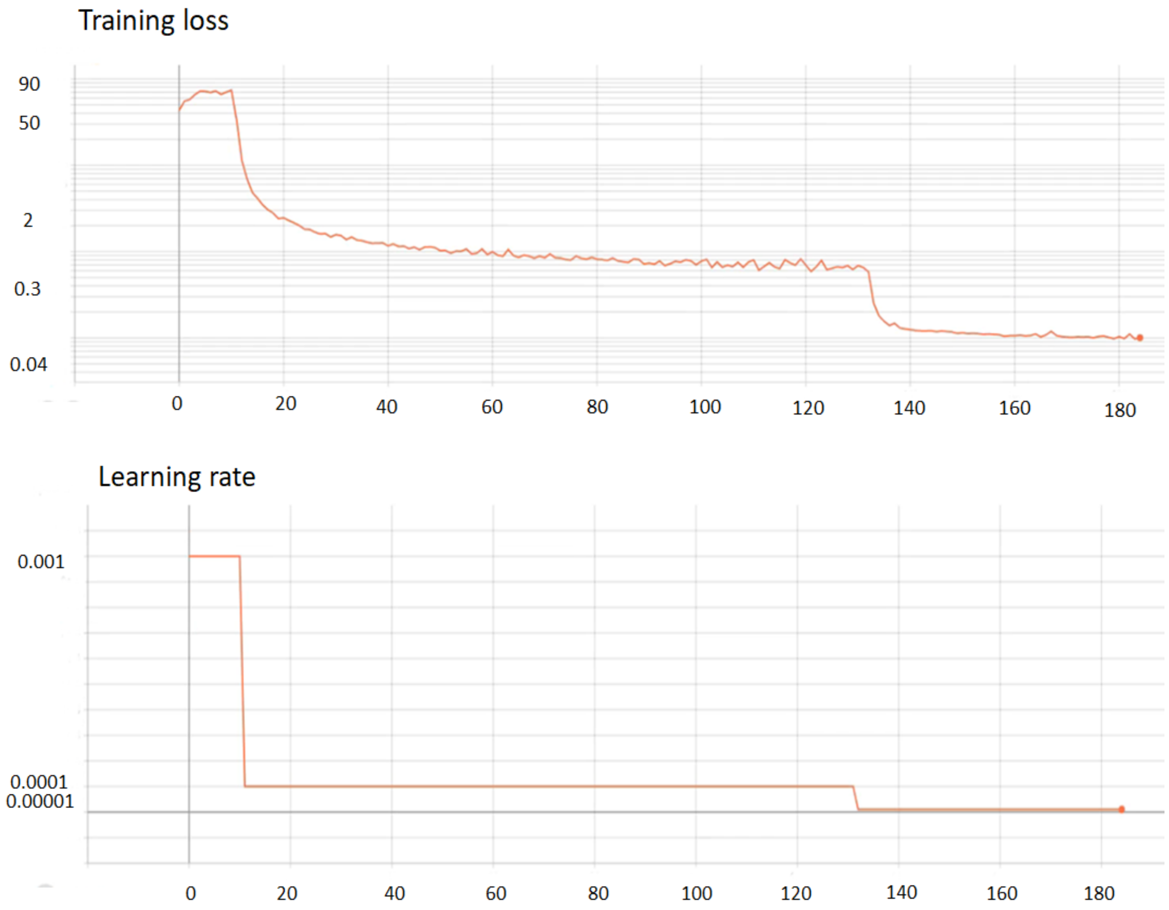
$$L(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls} (p_i, p_i^*) + \boldsymbol{\lambda} \frac{1}{N_{reg}} \sum_i p_i^* \, L_{reg}(t_i, t_i^*). \tag{6}$$

Where i is the i-th anchor in the batch and λ is a regularization factor used to balance the influence of the two losses. By default λ is set to 10 , and thus both classification and regression terms are roughly equally weighted . Shaoqing Ren et al. (2017) show by experiments that the results are insensitive to the values of λ in a wide range(Table 1).

λ	0.1	1	10	100
mAP(%)	67.2	68.9	69.9	69.1

Table 1. *Mean average precision obtained in function of different values*

To train our implementation of RPN, we chose Adam optimizer. The learning rate was initialized to 0.001 and adapted for two times when a plateau was reached. Specifically, when the training loss did not decrease for 10 subsequent epochs (patience=10), the learning rate was decreased by a factor of ten(Figure 24).


Training loss

Learning rate

Figure 24. *Representation of trend of training loss and learning rate in RPN over epochs. It is evident two drop of learning rate linked to the insertion of ReduceLrOnPlateau and the effects on training loss.*

To go through the Fast R-CNN, we need to know the position of the regions proposals given by the RPN in the image reference frame. This was done by applying the following inverse transformation of previous equation (4) :

$$
\begin{gathered}
x = t_x w_a + x_a, \qquad y = t_y h_a + y_a, \\
w = \mathrm{e}^{t_w} w_a, \qquad h = e^{t_h} h_a, \\
x^* = t_x^* w_a + x_a, \qquad y^* = t_y^* h_a + y_a, \\
w^* = e^{t_x^*} w_a, \qquad h^* = e^{t_h^*} h_a,
\end{gathered}
\tag{7}
$$

2.3.3. Fast R-CNN

For each anchor given as input to the RPN, a new region of interest (ROI) was defined through the coordinates given as output from the RPN. Along with the set of coordinates, the probability of the ROI of being background or foreground was calculated. However, the RPN makes a high-level estimation of regions that can contain the objects of interest. Fast R-CNN was therefore needed to apply a lower-level classification and to refine the predicted position of the nodules bounding boxes.

To properly apply this refinement, the number of ROI proposals was limited in four ways, following the reference code implementation:

i. Eliminating ROI overflowing image boundary
ii. Including ROI of size between 8 and 50 pixels with respect to image reference frame.
iii. Taking those with highest probability of being foreground;
iv. Excluding redundant ROIs through Non Maximum Suppression (NMS).

The first filtering(i) consist in the removal of ROI that cross the image boundary as already done for the RPN.

The second operation attempt to circumscribe the size of the ROI to better fit the nodule’s sizes. ROIs with size inferior to 8 pixels with respect to image reference frame represent an area inferior to one pixel in feature map, making senseless their inclusion. For what concerns high dimension ROIs, these are usually characterized by higher score and, except for few cases, by a low value of IoU with nodules. Considering that the filtering step (iii) keeps only the ROIs with higher score, the inclusion of ROIs with dimensions above 50 pixels, causes the exclusion of smaller ROIs in the next step. For these motivations we chose 50 pixels as maximum dimensional threshold to let the model to better fits the validation set characterized by nodules with size between 3 and 55 pixels.

For what concerns the solution (iii), since there is not a clear distinction between probabilities distributions of ROIs associated to positive anchors and probabilities distributions of ROIs related to negative anchors, it is not convenient to limit the proposals on the basis of a probability threshold. For this reason, a maximum number of 6000 ROIs was fixed in this first filtering step which correspond to the 6000 ROIs classified with highest probability of being foreground.

In the fourth filtering step (iv), NMS was applied to avoid redundant information. Specifically, IoU was applied among ROIs and, when there is an overlap higher than 0.7, only the ROI with highest p was considered. Figure 25 shows a comparison between the region proposals before (Figure 25A) and after (Figure 25B) the filtering procedure.

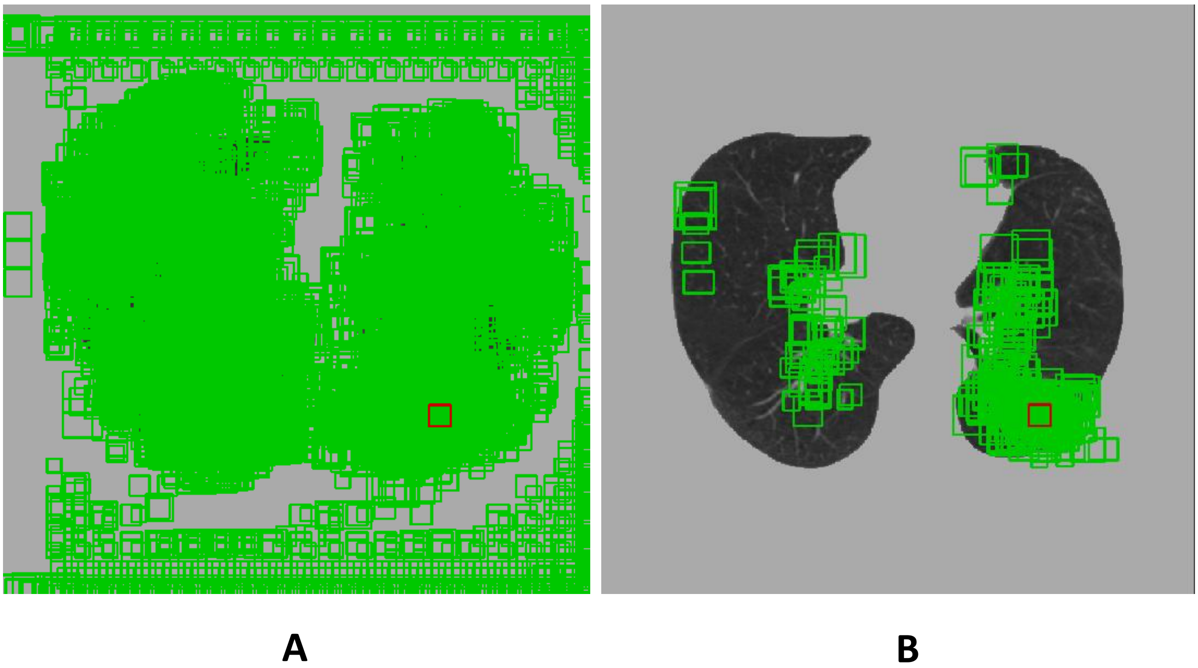


Figure 25. *Output of RPN without any filtering(A) and after the filtering operations(B).In the figure A ROIs appears to be concentrated on the lung region,while in figure B ROIs give more information about the lung nodule position even with lots of false positive. In red is represented ground truth box.*

The remaining proposals are 300 regions which were differently managed in training and validation phase. Specifically, for the training phase, a subsampling was applied to have balanced positive and negative ROIs in the same batch.
In the validation phase, no subsampling was applied and all the 300 ROIs are treated.

As regards the Fast R-CNN architecture, the implementation of Shaoqing Ren et al. (2017) was reproduced as done for RPN.
As already mentioned in section 1.4.3 and as summarized in Figure 27, the Fast R-CNN with respect to RPN takes two inputs:

- the feature map given as output by the backbone network (“Input_1”, Figure 26);
- the position of the ROIs with respect to the feature map reference frame which is described as usual by the center (x_{roi}, y_{roi}) ,the height and the width of the ROI (w_{roi}, h_{roi})(“Input_2”,Figure 26) .

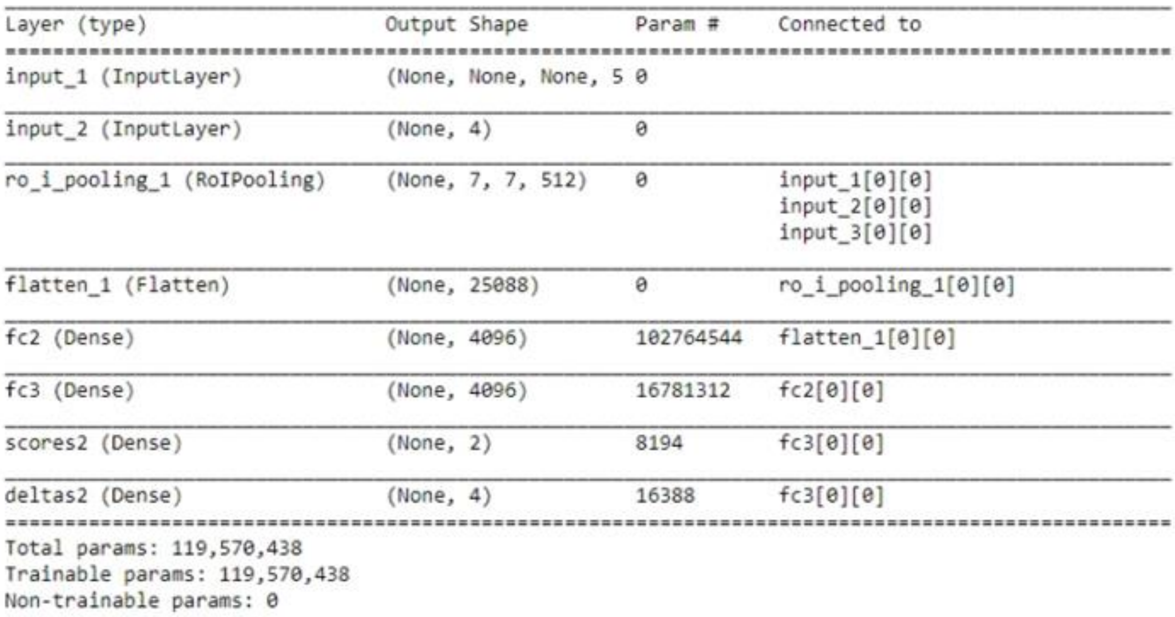
  Knowing the ROIs coordinates with respect to the image reference frame, coordinates of the ROI with respect to feature map were obtained through the following proportion:

$$x_{image}: image_{width} = x_{feature\ map}: feature\ map_{width}$$

$$y_{image}: image_{heigth} = y_{feature\ map}: feature\ map_{heigth}$$

  Where

  - x_{image} and y_{image} represent ROI coordinates with respect to the image reference frame.
  - $x_{featuremap}$ and $y_{featuremap}$ represent ROI coordinates with respect to feature map reference frame.



```
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            (None, None, None, 5 0
__________________________________________________________________________________________________
input_2 (InputLayer)            (None, 4)            0
__________________________________________________________________________________________________
ro_i_pooling_1 (RoIPooling)     (None, 7, 7, 512)    0           input_1[0][0]
                                                                 input_2[0][0]
                                                                 input_3[0][0]
__________________________________________________________________________________________________
flatten_1 (Flatten)             (None, 25088)        0           ro_i_pooling_1[0][0]
__________________________________________________________________________________________________
fc2 (Dense)                     (None, 4096)         102764544   flatten_1[0][0]
__________________________________________________________________________________________________
fc3 (Dense)                     (None, 4096)         16781312    fc2[0][0]
__________________________________________________________________________________________________
scores2 (Dense)                 (None, 2)            8194        fc3[0][0]
__________________________________________________________________________________________________
deltas2 (Dense)                 (None, 4)            16388       fc3[0][0]
==================================================================================================
Total params: 119,570,438
Trainable params: 119,570,438
Non-trainable params: 0
```

Figure 26. Fast *R-CNN network architecture*

The feature map ROIs were then processed through a ROI Pooling layer (“ro_i_pooling_1”,Figure 26) that bring all different sizes input ROIs to a fixed size, that is usually set equal to 7x7 pixels as reported in the original paper (Shaoqing Ren, 2017). Generally in ROI Pooling layer, a Max pooling operation is applied in order to bring all different sizes ROIs to 7x7 pixels with respect to the feature map reference frame. However, in our dataset, the majority of the nodules were represented by ROIs of size equal to 1x1 or 2x2, condition in which the size is too limited to apply a subsampling through Max-Pooling and reach the desired dimension. For this reason, the ROIs dimension was increased through the application of a bilinear interpolation. In our experiments, three different ROI Polling input size were investigated: 7x7 (section 2.4.1.4),3x3 (section 2.4.1.4) and 2x2 (section 2.4.1.4) pixels.

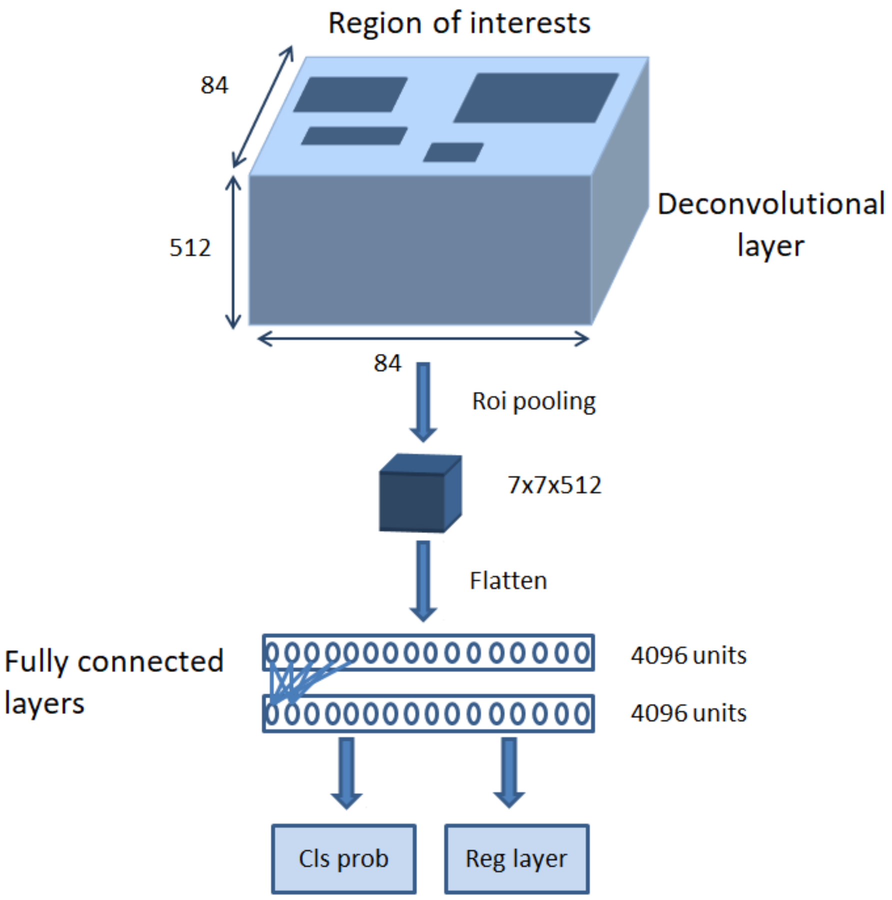


Figure 27. *Fast R-CNN workflow. You may notice that Roi pooling receive in input the coordinates of the ROIs with respect to the feature map reference frame and extract from the deconvolutional layer(feature map considered) region of 7x7x512 by means of a bilinear interpolation.*

Once ROIs size have been normalized by the ROI Pooling layer, a flatten operation was applied to proceed with two subsequent fully connected layers (4096 units) where a Hyperbolic tangent activation function (TanH) was used instead of a ReLU. This solution was adopted to avoid the death of neurons after few steps; indeed, the Fast R-CNN is subjected to very high gradient values that could cause the weights to update in such a way that the neuron will never activate again on any data point.

The Fast R-CNN, as well as the RPN, terminates with two output layers:

- a classification output, given by a softmax layer which returns for each ROI a 1x2 vector with the probabilities of being and not being a pulmonary nodule;

- a regression output, given by a linear layer which returns a 1x4 vector which describes the refined predicted position of the ROI with respect to image reference frame.

Also the training of the Fast R-CNN thus consists in the parallel minimization of a classification loss $L_{cls-Fast\ R-CNN}$ and a regression loss $L_{rgs-Fast\ R-CNN}$.

Similarly, to the RPN, the L_{cls} is a function of the predicted probability of being a pulmonary nodule and the target class.

Since ROIs correspond to different patches of the original image with respect to the previously defined anchors, the target class associated to each ROI needs to be established again. Specifically, IoU between the Fast R-CNN input ROIs and the ground truth box was calculated and new IoU upper was fixed$(th_{Fast\ R-CNN} = 0.4)$. The threshold value was decreased from RPN according to reference code. Thus, ROIs with IoU < $th_{Fast\ R-CNN}$ were associated with the target class [1,0] (i.e. non-pulmonary nodule), otherwise the target class [0,1] was assigned. The original implementation (Shaoqing Ren, 2017) excluded background ROI from the training: in a such small object dimension problem the high number of background ROI makes necessary to compare the results excluding and including background ROIs. For the accomplishment of this comparison, we fixed the lower threshold $(th_{low} = 0.016)$ in order to distinguish background from hard-negative ROIs. Also th_{low} was reduced with respect to the RPN value, with the same reduction applied for $th_{Fast\ R-CNN}$.

According to the chosen softmax classification layer, Categorical cross-entropy was used as Lcls which is defined as following:

$$L_{rgs-Fast\ R-CNN} = 1/M \sum_{p}^{M} -\log(e^{s_p} / \sum_{j}^{C} e^{s_j})$$

(8)

Where M are the positive class of a sample and where each s_p in M is the CNN score for each positive class.

For what concerns $L_{rgs-Fast\ R-CNN}$, same implementation of Huber loss was used. $L_{rgs-Fast\ R-CNN}$ is consequently a function of the predicted and target transformed coordinates which in this case depend on the ROIs position instead of the anchors position. Therefore, the previously defined equation (4) becomes:

$$
\begin{gathered}
t_x = \frac{x - x_{roi}}{w_{roi}}, t_y = \frac{y - y_{roi}}{h_{roi}}, \\
t_w = \log\left(\frac{w}{w_{roi}}\right), t_h = \log\left(\frac{h}{h_{roi}}\right), \\
t_x^* = \frac{x^* - x_{roi}}{w_{roi}}, t_y^* = \frac{y^* - y_{roi}}{h_{roi}}, \\
t_w^* = \log\left(\frac{w^*}{w_{roi}}\right), t_h^* = \log\left(\frac{h^*}{h_{roi}}\right),
\end{gathered}
\tag{9}
$$

Where

- t_x/t_x^* and t_y/t_y^* denote the transformed center coordinates of the predicted/ground truth box;
- t_w/t_w^* and t_h/t_h^* are the transformed width and height of the predicted/ground truth box;
- $x_{roi}, y_{roi}, w_{roi}$ and h_{roi} are finally the 4 variables that describe the ROI position in the image.

The overall loss of Fast R-CNN was obtained summing up $L_{cls-Fast\ R-CNN}$ and $L_{rgs-Fast\ R-CNN}$ multiplied by a regularization factor (λ= 10) similarly to equation (6).

As for RPN, to train Fast R-CNN we used Adam as optimizers. The learning rate was initialized to 0.0001 and reduced when a plateau was reached. Specifically, if the loss does not change for two subsequent weight updating, the learning rate was reduced(Figure 28). This procedure was repeated for two times.

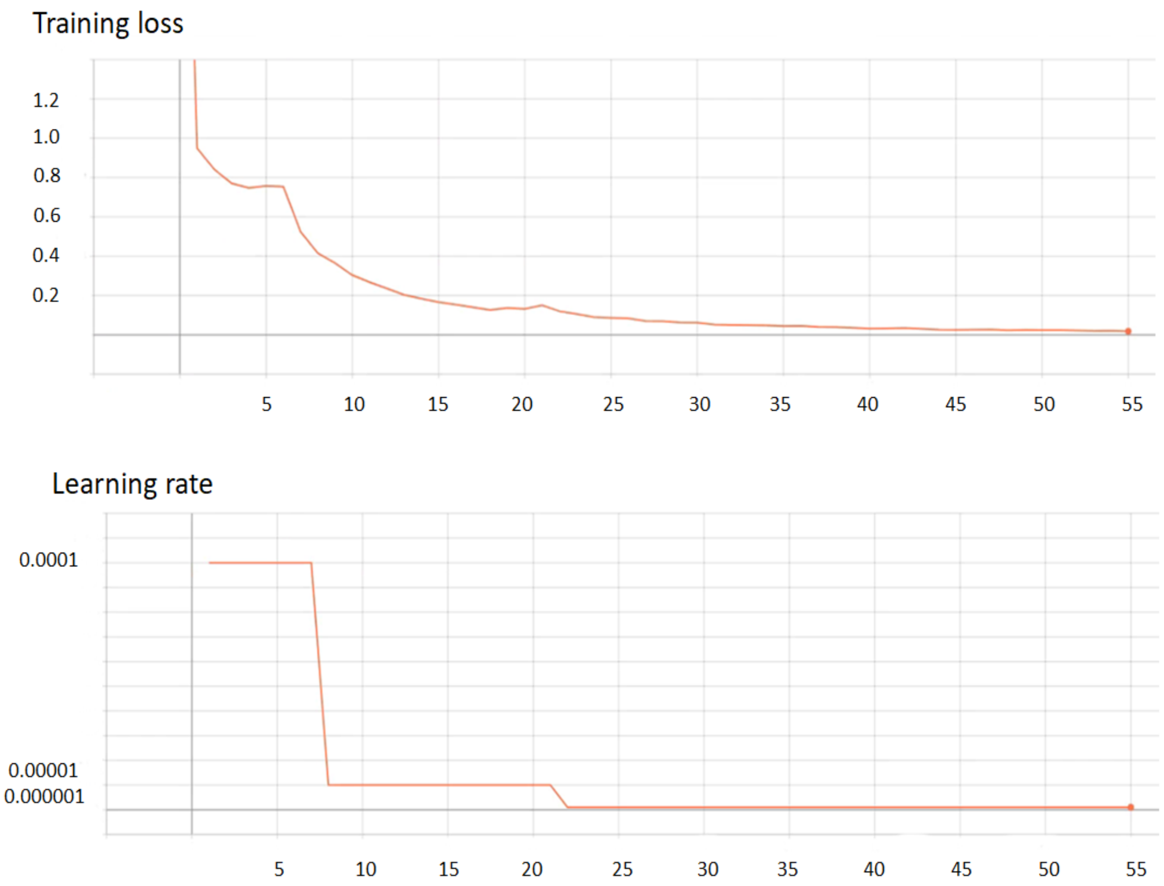


Figure 28. *Representation of trend of training loss and learning rate in Fast R-CNN over epochs. It is evident two drop of learning rate linked to the insertion of ReduceLrOnPlateau and the effects on training loss.*

In order to obtain the final proposal adjusted coordinates, we need to know the coordinates of the ROI with respect to image reference frame. This was done applying the following inverse transformation of the equation (9):

$$
\begin{aligned}
x = t_x w_{roi} + x_{roi}, &\qquad y = t_y h_{roi} + y_{roi}, \\
w = \mathrm{e}^{t_w} w_{roi}, &\qquad h = e^{t_h} h_{roi}, \\
x^* = t_x^* w_{roi} + x_{roi}, &\qquad y^* = t_y^* h_{roi} + y_{roi}, \\
w^* = e^{t_x^*} w_{roi}, &\qquad h^* = e^{t_h^*} h_{roi},
\end{aligned}
\tag{10}
$$

2.4. Experiments

To optimize the performance of the Faster-RCNN in detecting pulmonary nodules, the net was exploited from multiple points of view. We first investigated how modifications of parameters and architecture of the networks can influence the detection of lung nodules (Section 2.4.1). In a second time we carried out experiments on different classes of nodules, by examining the behavior of the network on different type of nodules(solid, part solid and GGO) and trying to figure out if a training on a specific class of nodule can improve the performance on that specific class (Section 2.4.2).

For the following reported experiments, the dataset and the subsets considered were always partitioned in training, the 80 % of the samples, and validation the remaining 20%. The validation set was considered to evaluate models on samples not involved in the training procedure and so to avoid conditions of overfitting in new data. For each test, the best model was considered as that the one associated with the lowest validation loss.

To train the VGG16, an equal set of images without lesions was considered to represent the negative class (*slice without nodule* class), along with 2D images where at least an identified pulmonary nodule was present (*slice with nodule* class). To train and validate RPN and Fast-RCNN only the set of 2901 2D images with presence of a lesion was instead considered.

In this implementation we did not use the alternative training (Appendix B), but VGG16, RPN and Fast R-CNN were considered as separated network and they were trained separately.

As evaluation metric, sensitivity and false positive per scan (FP/scan) were computed to derive the Free-Response ROC Curve (FROC curve). The FROC curve was chosen to evaluate the performance of different models and methods (Yanfeng, 2019).The FROC curve shows the relationship between the true positive rate (TPR, sensitivity) and the false positives per scan (FPs/scan) at different probabilities thresholds.

We also used the competitive performance metric (CPM) score, used in the LUNA16 challenge (Arnaud Arindra, 2016), to quantify the improvements between different experiments and to compare our results with those reported in the literature.

The CPM score was defined as the average sensitivity at the following seven predefined false positive points: 0.125, 0.25, 0.5, 1, 2, 4, and 6.

2.4.1. Technical experiments

Preliminary tests were done both on the backbone network (VGG16), to establish the best architecture adaptation in order to avoid overfitting in the application of transfer learning, and on the dimensional filtering applied to RPN outputs, by setting different thresholds (section 2.4.1.1). Defined the feature extraction procedure and relying on the public available implementation of RPN and Fast-RCNN, different batch dimension and strategies have been tested to train the Fast-RCNN (section 2.4.1.2). Different batch strategies have been tested also in RPN training (section 2.4.1.3). From an architectural point of view of the detection subnet, only different sizes of the ROI pooling layer of the Fast-RCNN were exploited (section 2.4.1.4).

2.4.1.1. Preliminary experiments on VGG16 architecture and RPN filtering procedure

The principal issue of backbone neural network is the high risk of overfitting, related to its depth and so to the high numbers of parameters to train(Section 2.3.1).

As preliminary tests, we therefore investigated multiple solutions to adapt the last portion of the backbone neural network structure in order to avoid overfitting. Specifically, before establishing the use of a GAP layer, max pooling layers were also tested. Keeping the same number of neurons in dense layers (1024), we replaced GAP with (i) two max-pooling layers and a Flatten layer and also with (ii) three max pooling layers and a Flatten layer.

Lung nodules represent a difficult target not only for the small dimension but also for their dimension variability. For this reason, another parameter established with a preliminary experiment, was the dimensional threshold applied to filter the ROI proposed by the RPN (section 2.3.3). Considering that in the validation set nodule's size ranges from 3 to 55 pixels, in the second preliminary experiment we evaluated three different maximum ROI size thresholds equal to 50,60 and 70 pixels, respectively.

2.4.1.2. Experiments on different Fast R-CNN batch composition

In the following experiments, different batch size and composition strategies were exploited to train the Fast-RCNN network, keeping the same feature extractor network and RPN. Specifically, VGG16 adapted with a GAP layer and the insertion of a deconvolutional layer was used as feature extractor after a training of 69 epochs. The RPN model applied in this phase was derived by training the net for 34 epochs and using batches of 512 samples; therefore, multiple 2D slices were involved in the same batch that could belong to different lesions or subjects. The set of 2D images considered to train and validate the RPN as well as the Fast R-CNN was the same and consisted in the entire set of nodules available. Solid, part-solid and non-solid nodules were therefore considered a unique set for this set of experiments.

A preliminary test was applied to evaluate the effect of including in the training phase ROIs with IoU < 0.016 instead of considering only hard negatives ($0.016<IOU<0.4$). Indeed, in the original implementation of Faster R-CNN proposed by Shaoqing Ren et al. (2017) background proposals ($IOU<0.016$) were not included; the negative class used to train the Fast R-CNN consisted only in hard negatives since they have a major overlap with the object of interest. This experiment was motivated by the difference between a natural images detection problem, such as the one faced by Shaoqing Ren et al. (2017), and lung nodule detection problem, where the difficulty related to the detection of small object could lead to an high number of background proposals output from RPN. Indeed, as already mentioned, a clear distinction between background and foreground classification probabilities output by RPN was not observed, meaning a poor ability of the subnet in catching better proposals. Considering background ROIs in Fast-RCNN training, can allow it to learn additional features previously not seen by the RPN.

The best strategy to define the negative training samples was defined evaluating the model behavior on the validation set and then different batch composition were tested in order to balance negative and positive class.

The batch dimension was chosen according to the medium number of positive ROIs (IOU>0.4) found among the 300 proposals sampled from the RPN output in the training dataset. The medium number of positive ROIs, with the generated RPN model, was equal to 8. Considering that the number of positive ROIs ranged between 20 and 1, the following 4 different batch size composition were evaluated:

1. Fixed batch size equal to 25 ROIs with maximum of positive ROIs fixed to 12
2. Fixed batch size equal to 16 ROIs with maximum of positive ROIs fixed to 8
3. Fixed batch size equal to 8 ROIs with maximum of positive ROIs fixed to 4
4. Variable batch size in order to keep 1:1 ratio between positive and negative ROIs

For tests 1-3 in case of N_{pos_rois} higher than the fixed quantity, a random subsampling was applied to both negative and positive ROIs.

The Fast-RCNN batch strategy resulted the most performant was then adopted in the subsequent experiments.

2.4.1.3. Experiments on different RPN batch strategy

In our experiments, we focused not only on the Fast R-CNN training(Section 2.4.1.2), but we attempted to figure out also the best way to train RPN.

As said in section 2.4.1.2, for the experiments reported in the same section, the RPN was trained considering a fixed batch size equal to 512 according to reference code(https://github.com/dongjk/faster_rcnn_keras), including feature map coming from different lesions or patients inside the same batch. In this experiment section we wanted to compare the previously RPN adopted training approach with the "image-centric sampling strategy " used in the original implementation (Shaoqing Ren, 2017)in order to define the best strategy. This second approach consists in the inclusion of samples coming from a single image in each batch.

Keeping the same RPN batch composition explained in section 2.3.2(number of negative anchors equal to $2 * N_{anc\_pos}$), we compared the performance of a Faster R-CNN trained with a RPN characterized by mono-image batch and a Faster R-CNN trained with RPN characterized by multi-image batch approach.

2.4.1.4. Experiments on different implementation of ROI pooling layer

The first implementation of the network did not operate a real ROI pooling , but performed a simple interpolation of the ROI to 7x7 pixels regions. Indeed, as explained in section 2.3.3, the ROIs with respect to the feature map reference frame were always inferior than 7x7 pixels. For this reason we can not operate a features selection by means of Max-pooling operation and also we have to introduce an interpolation procedure in order to bring all the ROIs to the desired dimension (7x7 pixels).In order to overcome this problem, in these experiments, we followed a different strategy changing the dimension of the ROI pooling fixed by the paper which was equal to 7x7 pixels (Shaoqing Ren, 2017). Specifically, the interpolated ROI size was decreased and then a Max-pooling operation was applied as explained following where the two experiments are reported:

1. ROIs interpolation to 6x6 pixels and max-pooling application of pool-size(2,2), obtaining 3x3 pixels regions in input to Flatten layer(Figure 26).
2. ROIs interpolation to 4x4 pixels and max-pooling application of pool-size(2,2), obtaining 2x2 pixels regions in input to Flatten layer(Figure 26).

2.4.2. Experiments on different classes of lung nodules

This section of experiments had the aim to evaluate the behavior of the net on the three different types of nodules (solid, part-solid and non-solid) and to establish if there is an advantage in using a specific model for each class.

Three different subsets were defined for solid, part-solid and GGO nodules respectively. With respect to the overall set of nodules, only lesions associated with box dimensions above 16 pixels were included to simplify the problem of small object detection and to better evaluate the model's limits with respect to nodule type.

For each subset, a partition in training and validation was applied. A nodule type-specific Faster-RCNN was therefore derived retraining VGG16, RPN and Fast-RCNN. For what concerns solid nodules, a number of 724 images were considered as training samples. The generated model was then evaluated on the 158 samples of the validation set.

The same approach was used to evaluate part-solid and GGO specific models. For the part-solid model, a subset of 337 and 74 2D images samples were considered for training and validation respectively; in the two sets, for the non-solid model, a number of 461 and 103 images was instead included.

On these three sets of data(solid, part-solid and GGO) also the best model obtained using the entire set of nodules (section 2.4.1.2) was evaluated and its performance were compared with those of the solid type-specific model.

Chapter 3. Results and discussions

In this chapter we present and discuss the results in terms of performance of the different Faster R-CNN implementation.

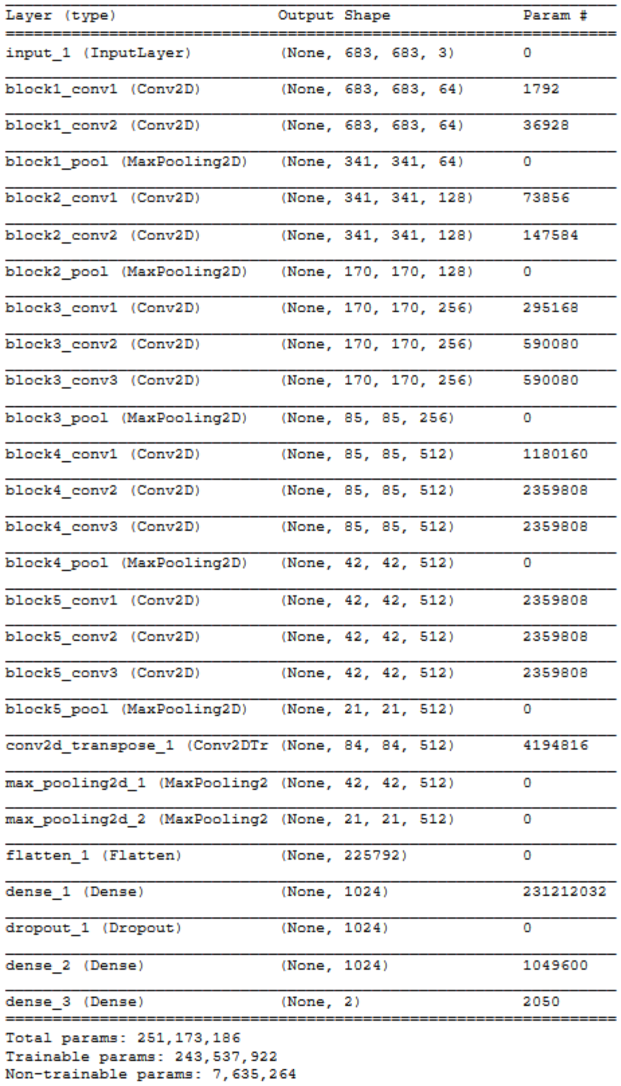
3.1. Results on technical experiments

In the first part we present and discuss the results relative to the technical experiments underlining the detection improvement or worsening obtained from parameters and structure changes(Section 2.4.1).

3.1.1. Preliminary results on VGG16 architecture and RPN filtering procedure

For what concern the architecture of VGG16, we evaluated the model by replacing GAP with max pooling layers.

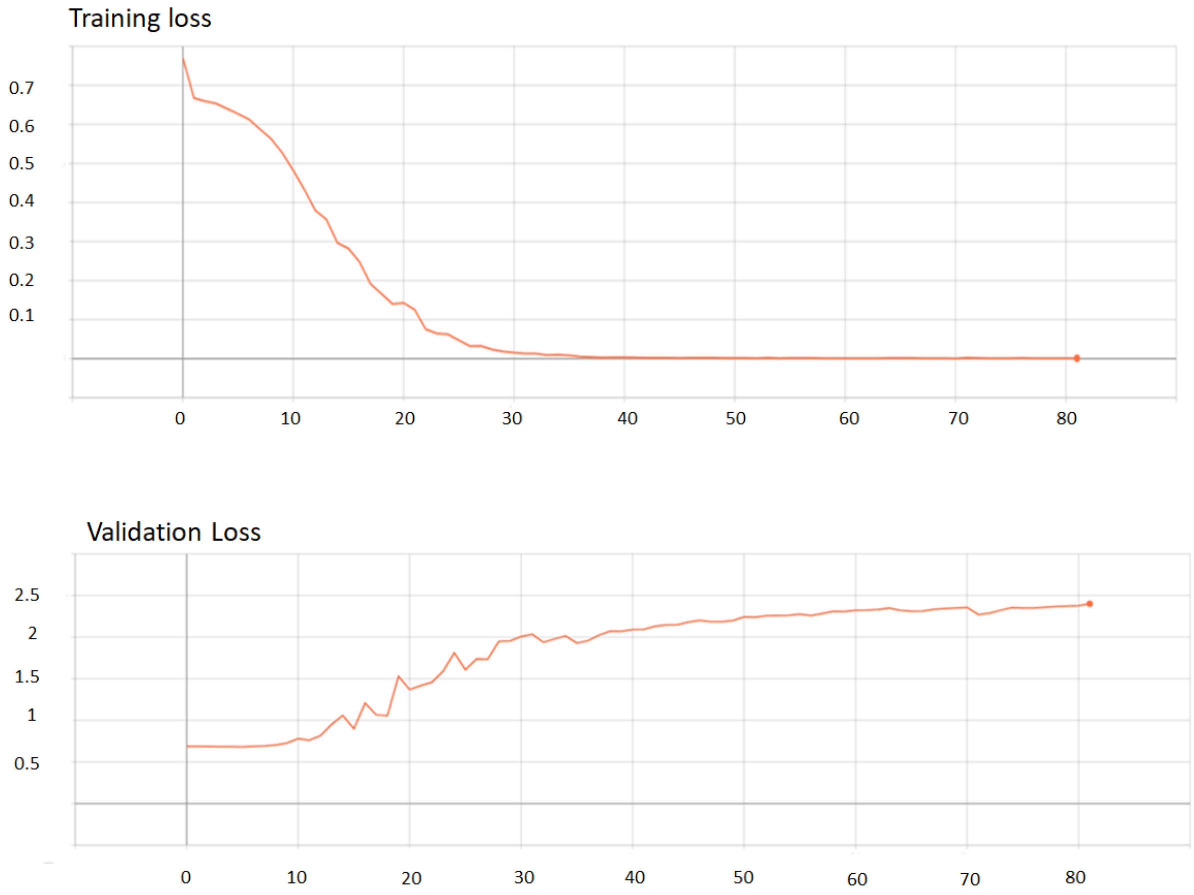
In the first preliminary experiment, keeping the same number of neurons in dense layers (1024), we substituted GAP with two max-pooling layers and a Flatten layer(i) (section 2.4.1.1).



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
input_1 (InputLayer)         (None, 683, 683, 3)       0         
_________________________________________________________________
block1_conv1 (Conv2D)        (None, 683, 683, 64)      1792      
_________________________________________________________________
block1_conv2 (Conv2D)        (None, 683, 683, 64)      36928     
_________________________________________________________________
block1_pool (MaxPooling2D)   (None, 341, 341, 64)      0         
_________________________________________________________________
block2_conv1 (Conv2D)        (None, 341, 341, 128)     73856     
_________________________________________________________________
block2_conv2 (Conv2D)        (None, 341, 341, 128)     147584    
_________________________________________________________________
block2_pool (MaxPooling2D)   (None, 170, 170, 128)     0         
_________________________________________________________________
block3_conv1 (Conv2D)        (None, 170, 170, 256)     295168    
_________________________________________________________________
block3_conv2 (Conv2D)        (None, 170, 170, 256)     590080    
_________________________________________________________________
block3_conv3 (Conv2D)        (None, 170, 170, 256)     590080    
_________________________________________________________________
block3_pool (MaxPooling2D)   (None, 85, 85, 256)       0         
_________________________________________________________________
block4_conv1 (Conv2D)        (None, 85, 85, 512)       1180160   
_________________________________________________________________
block4_conv2 (Conv2D)        (None, 85, 85, 512)       2359808   
_________________________________________________________________
block4_conv3 (Conv2D)        (None, 85, 85, 512)       2359808   
_________________________________________________________________
block4_pool (MaxPooling2D)   (None, 42, 42, 512)       0         
_________________________________________________________________
block5_conv1 (Conv2D)        (None, 42, 42, 512)       2359808   
_________________________________________________________________
block5_conv2 (Conv2D)        (None, 42, 42, 512)       2359808   
_________________________________________________________________
block5_conv3 (Conv2D)        (None, 42, 42, 512)       2359808   
_________________________________________________________________
block5_pool (MaxPooling2D)   (None, 21, 21, 512)       0         
_________________________________________________________________
conv2d_transpose_1 (Conv2DTr (None, 84, 84, 512)       4194816   
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 42, 42, 512)       0         
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 21, 21, 512)       0         
_________________________________________________________________
flatten_1 (Flatten)          (None, 225792)            0         
_________________________________________________________________
dense_1 (Dense)              (None, 1024)              231212032 
_________________________________________________________________
dropout_1 (Dropout)          (None, 1024)              0         
_________________________________________________________________
dense_2 (Dense)              (None, 1024)              1049600   
_________________________________________________________________
dense_3 (Dense)              (None, 2)                 2050      
=================================================================
Total params: 251,173,186
Trainable params: 243,537,922
Non-trainable params: 7,635,264
```

Figure 29.*Replacemenet of GAP with two max-pooling and Flatten layer in VGG16 architecture. Number of features drastically increase to 251 millions.*

As reported in Figure 29, the number of features relative to VGG16 training increased from 21 millions (in the GAP implementation) to approximately 251 millions. Even if we use the pre-trained Imagenet weights, this solution led to overfitting due to a huge increase of features as can be noted comparing the trend of training and validation losses in Figure 30.



Training loss

Validation Loss

Figure 30. *The replacement of GAP with two max-pooling layers cause overfitting*

In the second experiment(ii) (section 2.4.1.1), we also evaluated the behavior of the VGG16 after the addition of an additional max pooling layer, reducing number of features from 251 to 72 millions.

Even the addition of this ulterior max pooling layers, did not solve the problem of overfitting resulting in a behavior of losses during training similar to that observed in Figure 30.

These results show that the reduction of features operated by GAP layers, which reduce a tensor of dimensions 84×84×512 to 1×1×512, proved its usefulness to avoid overfitting.

Regarding the maximum size ROI threshold applied in dimensional filtering, we chose the threshold value according to the minimum validation loss obtained which corresponded to a threshold equal to 50 pixels (Table 2).

Maximum ROI size threshold[pixels]	50	60	70
Validation loss	0.1868	0.2387	0.3156

Table 2. *Validation loss corresponding to different maximum ROI size thresholds*

Including ROIs of size higher than 50 pixels reduce the performance of the network. Indeed high size ROIs are generally characterized by high score but a low IoU with the ground truth box, thus causing the exclusion of small size ROIs in the score filtering step(iii), that more likely fit better lung nodules(Section 2.3.3). For these reasons, we chose 50 pixels as maximum size threshold.

3.1.2. Results on different batch composition

Here we report results related on how the dimension and composition of batch size of Fast R-CNN influenced the performance of the Faster R-CNN.

The Fast R-CNN training approach of Shaoqing Ren (2017) that excludes background ROI(IOU<0.016) and considers only the hard negative ROIs (0.016<IOU<0.4) as negative class, did not lead to a good performance in our experiments. Conversely, the inclusion of background ROIs in the training phase improved the performance of the network: Figure 31 reports the FROC curve related to the Fast RCNN trained excluding the background samples in comparison to the case characterized by the inclusion of background samples, with both trainings using a fixed batch equal to 16 (Section 2.4.1.2).

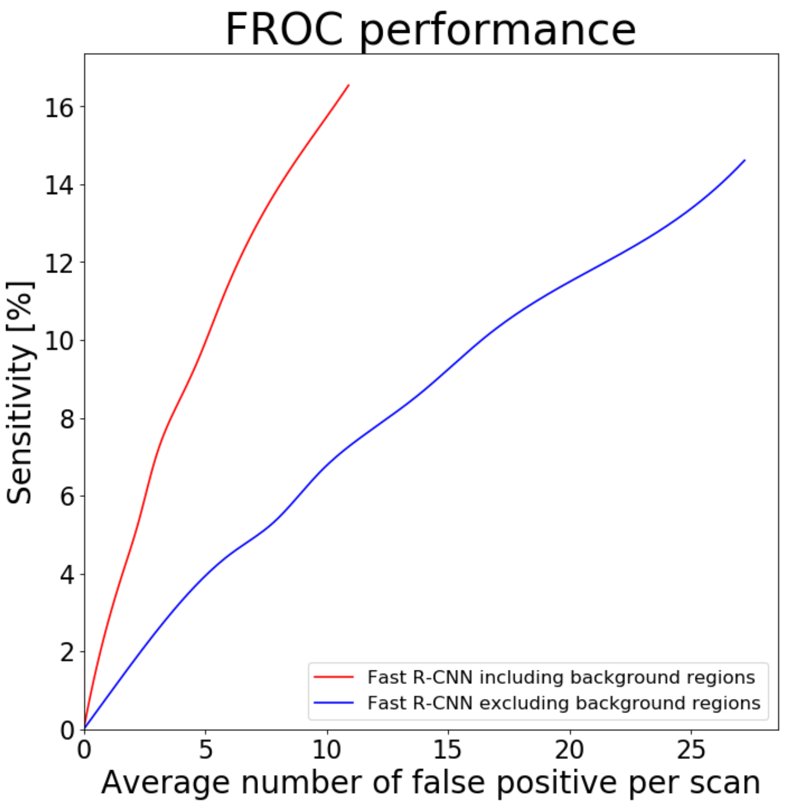


Figure 31. *Comparison between two Faster R-CNN: Fast R-CNN including(red) and excluding(green) background region(IOU<0.016) inside negative label of batch.*

With the first method, characterized by the inclusion of background ROIs inside the training of Fast R-CNN, a CPM score equal to 0.0431 was obtained confirming the improvement with respect to the second experiment where a CPM score of only 0.0158 was reached.

This result underlines the importance including background ROIs inside the Fast R-CNN training and can be associated in part to a bad performance of the RPN, which was not sufficiently able to discriminate between background and foreground proposals. In addition, some aspects of the four step filtering can be call into question too; the choice to reduce the number of ROIs to 300 in NMS step(section 2.3.3) is one of them. Being a number established a priori, different values needs to be exploited in order to reduce the bakground proposals output from RPN and at the same time to not lose proposals that well represent the nodule. However, further investigations should be done on a larger dataset and after improving the performances of the backbone network.

In Figure 32, a comparison of models generated according to experiments 1-4 of section 2.4.1.2 is reported. As can be noted, better performance was reached training the Fast-RCNN with a fixed batch of 16 ROIs samples: with equal false positive per scan, a higher sensitivity can be appreciated with respect to the others training approaches. Indeed the CPM score relative to Batch16 method was 0.0431, higher than the CPM score of Variable batch, Batch25 and batch8 methods equal to 0.0424,0.0381 and 0.0356, respectively.

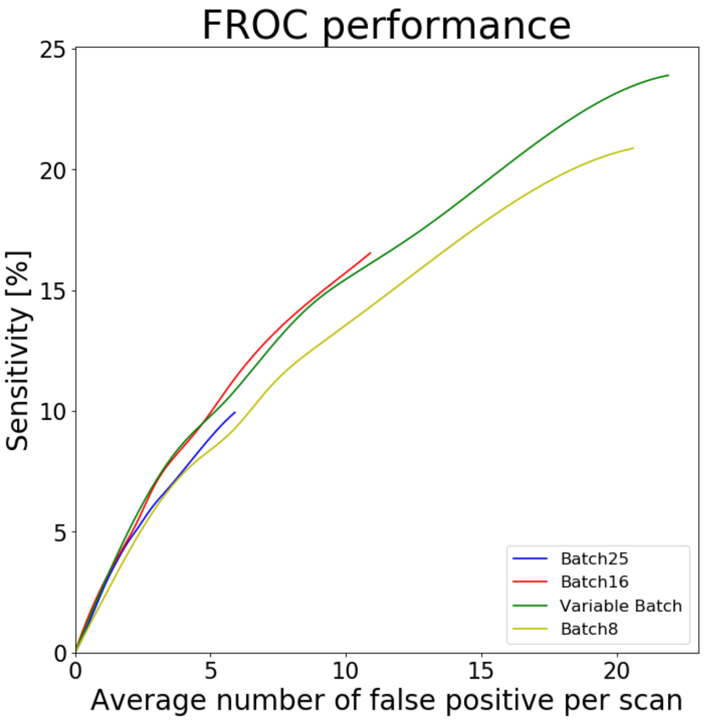


Figure 32. *FROC curve for different Fast R-CNN batch composition and dimension.*

Considering that, after the application of the filtering processes i-iv (section 2.3.3), the mean number of positive ROIs was equal to 8, it should be noticed that fixing a batch at 8 ROIs and so a more precise balance of positive/negative samples in the batch, higher values of sensitivity are obtained (yellow line, Figure 32), but at the same time there was an increase of FP/scan. This behavior characterizes also the batch “variable batch” (green line, Figure 32) implementation, where we have always a precise balance of positive/negative samples. The better performance of variable batch implementation with respect to batch 8, can be due to the fact that ,by fixing a batch at 8 ROIs, we applied a subsampling of positive ROIs causing a general reduction of the informative ROIs samples included in the training.

The opposite result can be observed in the implementation with batch fixed at 25 ROIs (blue line, Figure 32) , where positive sample represent a smaller portion of the batch with respect to the previous cited case. For this reason, we obtained better performance in term of FP/scan, but at the same time a reduction of sensitivity.

According to the considerations reported above, the choice of a batch size equal to the double of the mean number of positive ROIs, represents a good trade-off between the tested strategies and this has been considered in the next experiments(Section 2.4.1.3/2.4.1.4).

Figure 33 show a qualitative example of the outputs of Faster R-CNN with Batch 25 (A) ,Batch 16 (B),Batch 8 (C) and Variable batch (D) implementations.

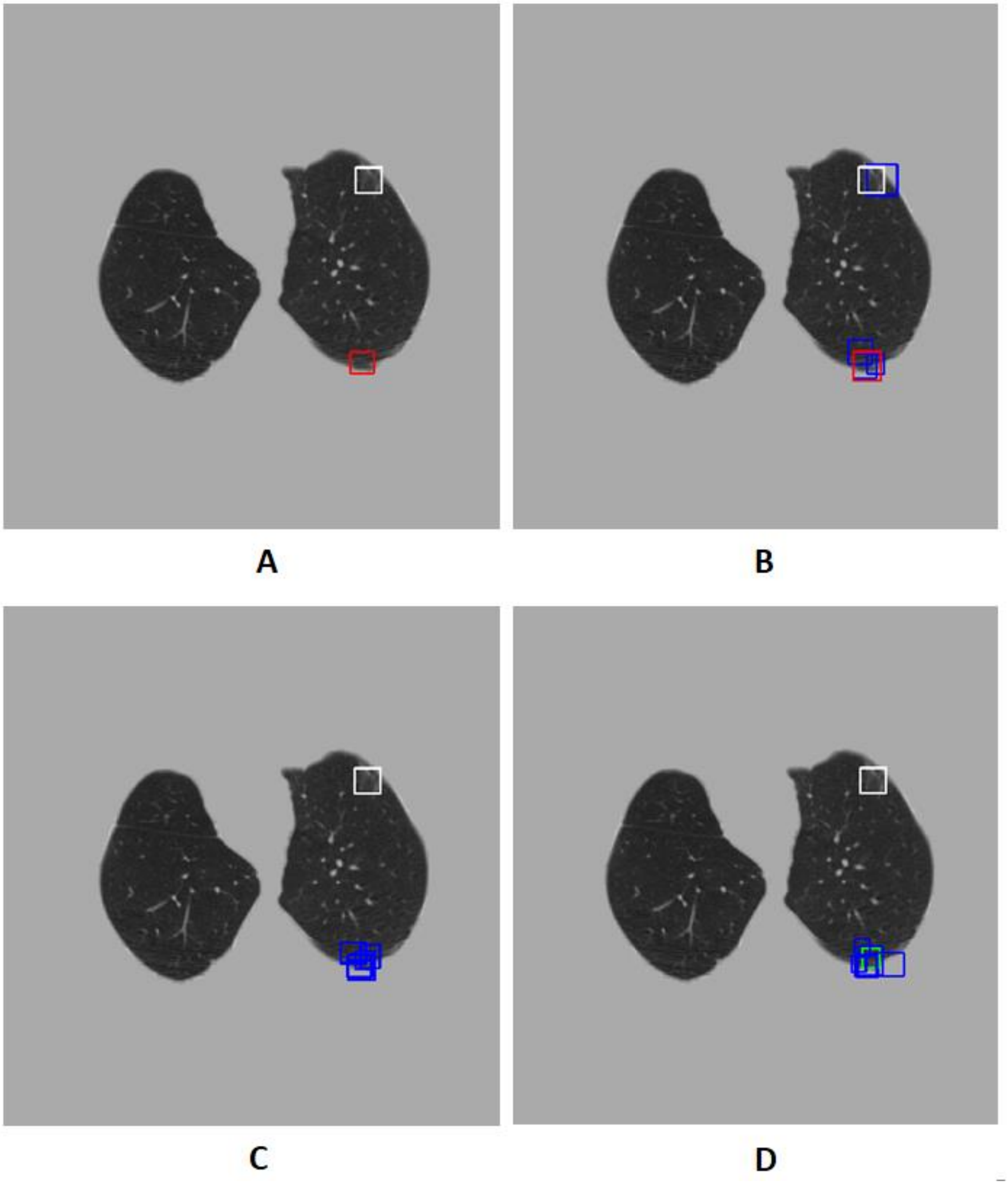


Figure 33. *Output of Faster R-CNN with Batch 25 (A) ,Batch 16 (B),Batch 8 (C) and Batch-variable (D) implementations. In white is represented the ground truth box, while in green, red and blue the ROIs with probability(p) of being nodules respectively: p>0.9(green), 0.7<p<0.9(red) and 0.5<p<0.7(blue).*

As can be noted from the qualitative example reported in Figure 33, using a batch size of 16 ROIs (Figure 33 B), a better detection of the nodule is achieved, avoiding lots of FP that are conversely present using a batch size of 8 ROIs (Figure 33 C) and "variable batch"(Figure 33 D). Batch of 25 ROIs (Figure 33 A) , even if characterized by a reduction of false positive with respect to the other cases, does not detect the nodule. Moreover, although Batch 16 implementation recognized the nodule, the ROIs are focused with higher probability on another part of the parenchyma (Figure 33B), behavior that characterize also the other implementations.

The general poor performance of the network may be related to the difficulty encountered in identifying small nodules. For this reason, we evaluated the dimension of the tumors that the network identifies correctly (Figure 34). Overall, the network mainly fell in identifying small nodules (nodules with dimension < 13), although errors are also present in case of medium and large size nodules. It is expected that this result could potentially improve by performing a further reduction of the stride value. A stride of 8[a.u.] make impossible the extraction and classification of ROIs of size inferior to 8 pixels with respect to the image, because they become smaller than one pixel with respect to the feature map reference frame. As such, a reduced stride allows recovering more fine-grained features which are fundamental to detect small objects (Jia Ding A. L., 2017).

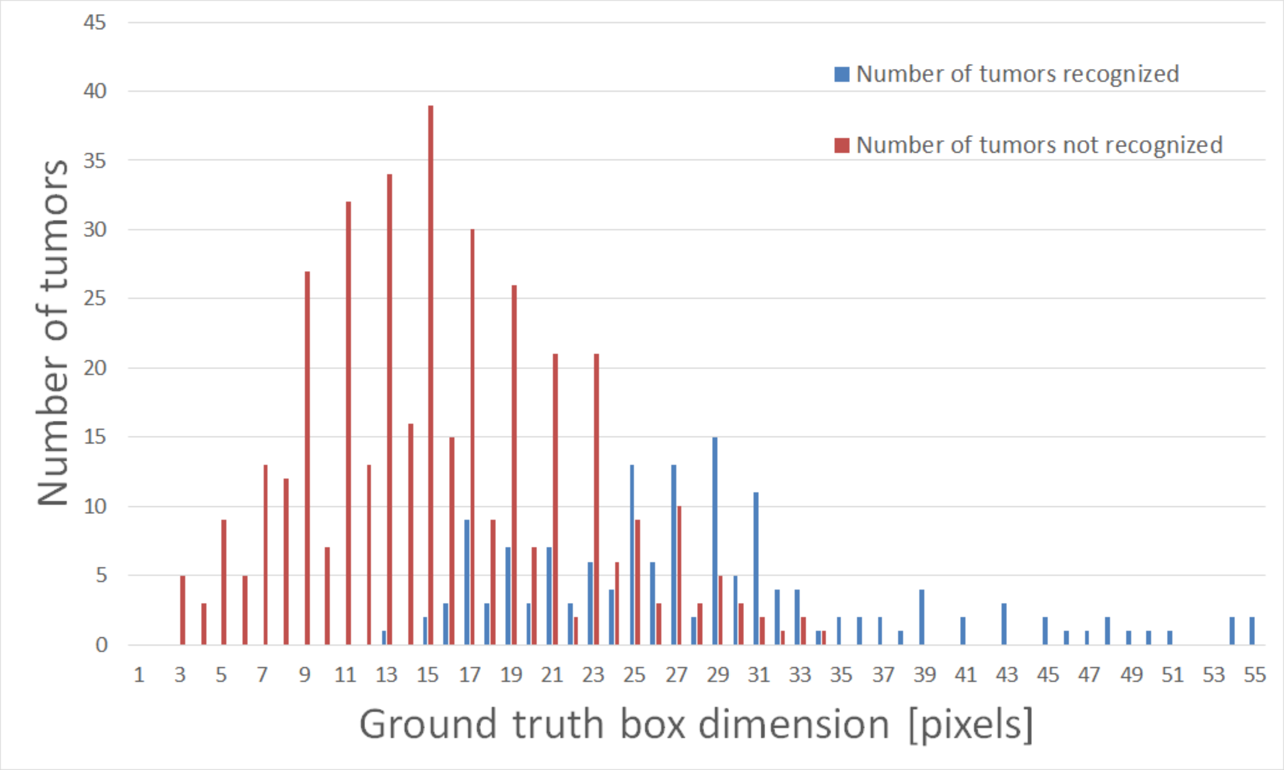


Figure 34. *Number of tumors recognized(blue) and not recognized(orange) in function of their dimension in pixel.*

3.1.3. Results on different RPN batch strategy

An additional attempt done to improve the network behavior acts on the RPN training procedure and consists in the application of the image-centric sampling strategy(section 2.3.2). In Figure 35 the FROC curves relative to Faster R-CNN with multi-image RPN batch and Image-centric sampling strategy are reported.

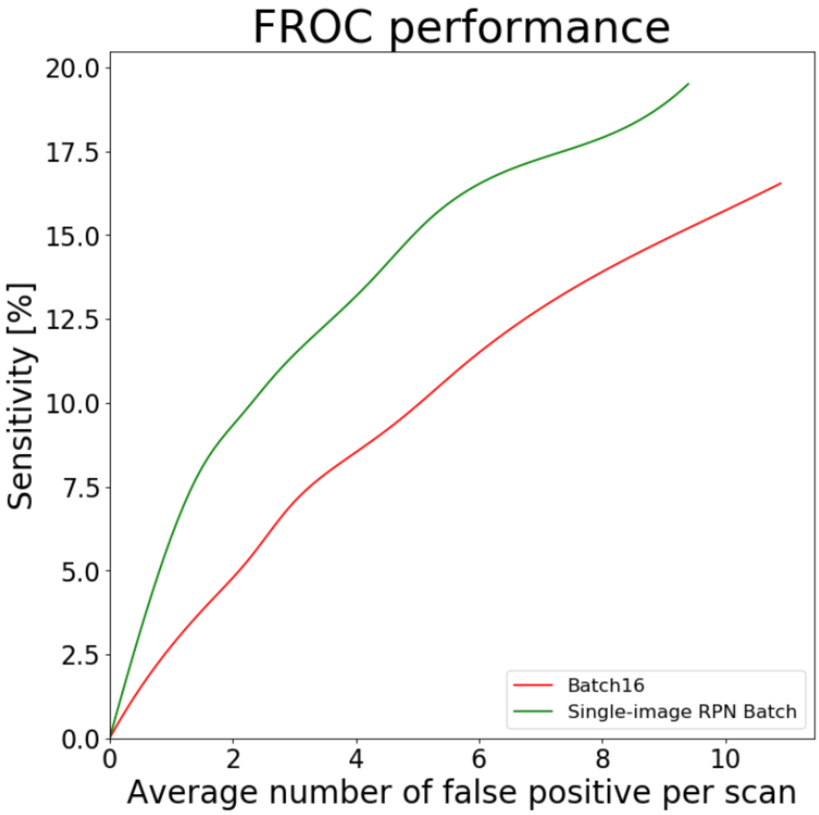


Figure 35. *FROC curve of Faster R-CNN with multi-image RPN batch(red) and image-centric sampling strategy (green)*

As we can see from Figure 35, the application of the image-centric sampling strategy(green curve), with respect of having a larger batch of 512 samples with multiple image features, brought to an improvement in terms of sensitivity : The CPM score increased from 0.0431 to 0.0723. This result not only highlights the convenience of treating image singularly but also that RPN achieved better performance adopting a smaller batch (Shaoqing Ren, 2017).

For this reason, the RPN model trained through a single-image batch was considered for the latter section of technical experiments(Section 2.4.1.4). On Figure 36 a qualitative example is reported, where the RPN trained trough image-centric sampling is compared with the RPN trained with a fixed batch of 512.

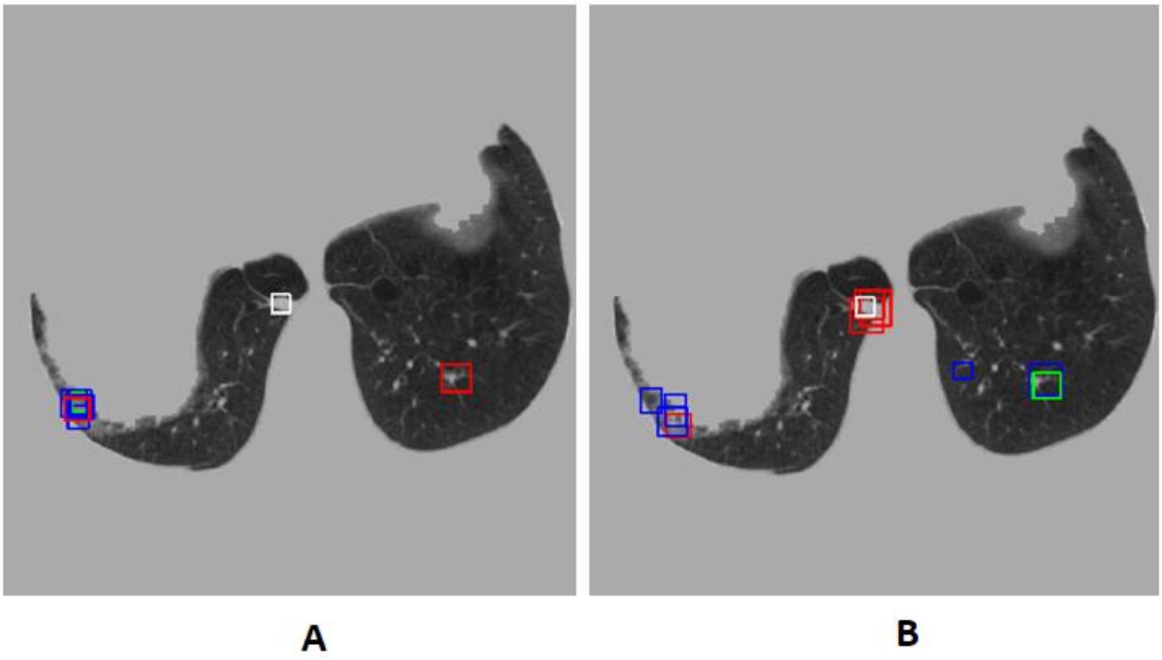


Figure 36. *Output of Batch 16 (A)(Section 3.1.2) and Faster R-CNN trained with single-image RPN Batch(B). In white is represented the ground truth box, while in green, red and blue the ROIs with probability(p) of being nodules respectively: p>0.9(green), 0.7<p<0.9(red) and 0.5<p<0.7(blue).*

Although the improvement obtained with RPN image-centric strategy, the performance of the network was still poor, underlining the need for evaluating other parameters and structure's characteristics to improve results.

3.1.4. Results on different implementation of ROI pooling layer

The dimension of the input ROI which was fixed in the original paper (Shaoqing Ren, 2017) and equal to 7x7 pixels, does not allow the correct implementation of ROI pooling layer in such small object detection problem, as lung nodules detection is. A possible solution that we investigated, was the reduction of original 7x7 pixels dimension by means of a lower interpolation and the insertion of a max-pooling operation(Figure 37).

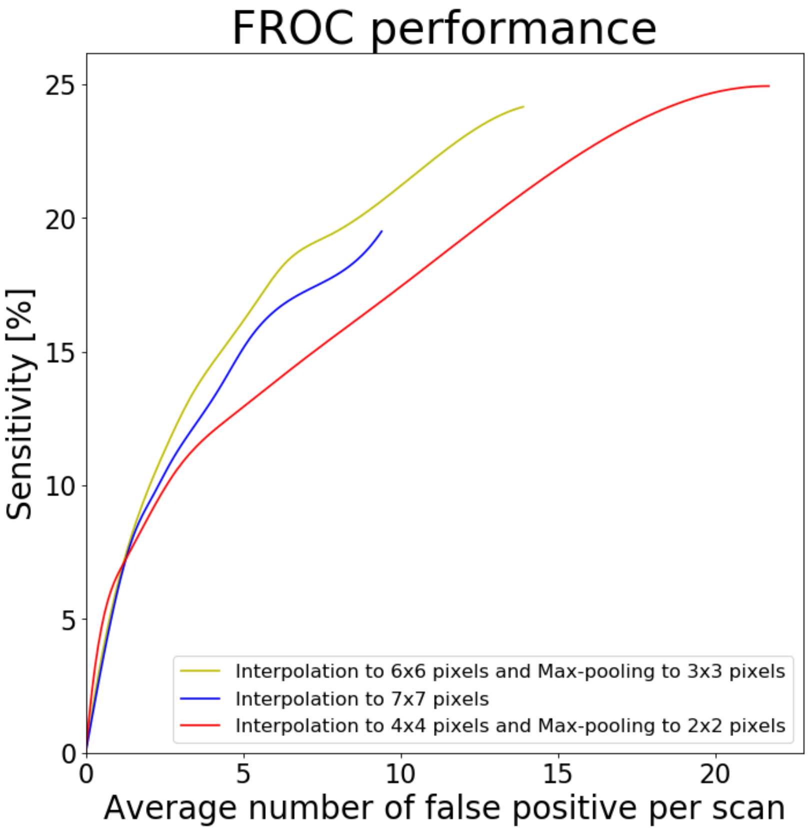


Figure 37.*FROC curve of different ROI-pooling implementations. In blue is represented the performance of the image-centric RPN training(section 2.4.1.3) with the previous implementations of ROI pooling layers. In red and yellow are represented the two experiments concerning the modification of ROI pooling layers(section 2.4.1.4).*

Figure 37 reports the result related to the three different approaches .

The use of a 3x3 warped ROI (yellow line, Figure 37)by means of a lower interpolation and the introduction of Max-pooling layer led to a slight improvement of the performance(CPM score of 0.0789) with respect to the previous implementation based on the interpolation to 7x7 pixels ROI that obtained a CPM score of 0.0723(Section 3.1.3). The additional reduction of the dimension of the ROIs to 2x2 pixels did not bring to an improvement obtaining a CPM score of 0.0719 (red line, Figure 37). These results show that a lower interpolation and the selection of features operated by Max pooling layers ,as implemented in the paper (Shaoqing Ren, 2017) , aids the network in the detection task, but at the same time a too high reduction of the ROI features input to the network(2x2 pixels ROIs) worsen the performance of the Faster R-CNN.

The approximate implementation of ROI pooling layers was certainly an issue of our detection system but the correct implementation by means of dimensionality reduction of input ROIs from 7x7 pixels (Shaoqing Ren, 2017) does not seem to be a convenient solution. The results of this section express furtherly the need for reducing the stride value in order to increase the ROIs dimension with respect to the feature map reference frame and implement a real ROI pooling layer without a reduction of the ROI’s size fixed by Shaoqing (2017).

Further investigation of this parameter is therefore necessary and need to be done in parallel to the changes on the feature extraction part of the detection model in order to improve the performance of the network, that still remains far from a good result.

3.2. Results on different classes of lung nodules

According to results reported in section 3.1.2, a clear lower performance only on nodule with lower size was not observed. To understand if the poor performance is related to a bias of the net towards a particular lesion class, nodule-type specific models were tested and, in this section, we present the obtained results. We removed nodules with box dimension below 16 pixels from the three subclasses(solid, part-solid, GGO)(Section 2.4.2) in order to simplify the small object detection issue and investigate more deeply on model's behavior with respect to the lesion class.

Firstly, the performance of batch16 model(Section 3.1.2) trained on the overall set of nodules, was evaluated on the three different subclasses(Figure 38).

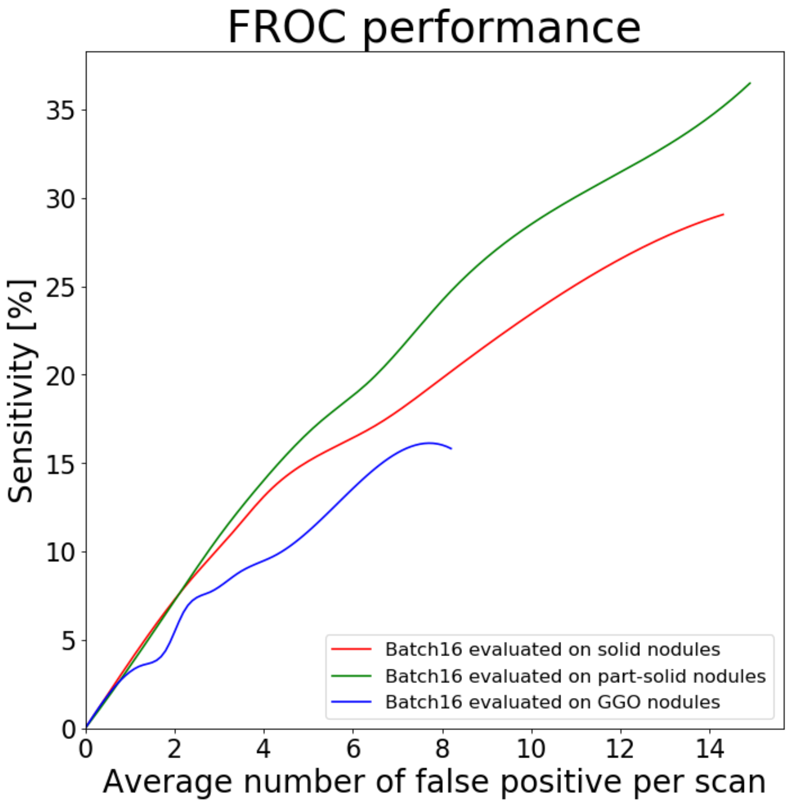


Figure 38. *Performance of Batch-16 implementation(section 3.1.2) on solid(red) part-solid(green) and GGO(blue) nodules of size >16.*

As it can be seen from Figure 38, the performances of the network on solid and part-solid nodules are better with respect to the performance on GGO nodules.

The bad results could be related to the inability of the network in learning part of the features related to a particular type of nodules(groundglass features), therefore a comparison with a nodule type-specific network was necessary to understand if a large number of properties are caught.

Following, the curves reported in figure 38 are compared with those trained on a specific class of nodules(Figure 39).

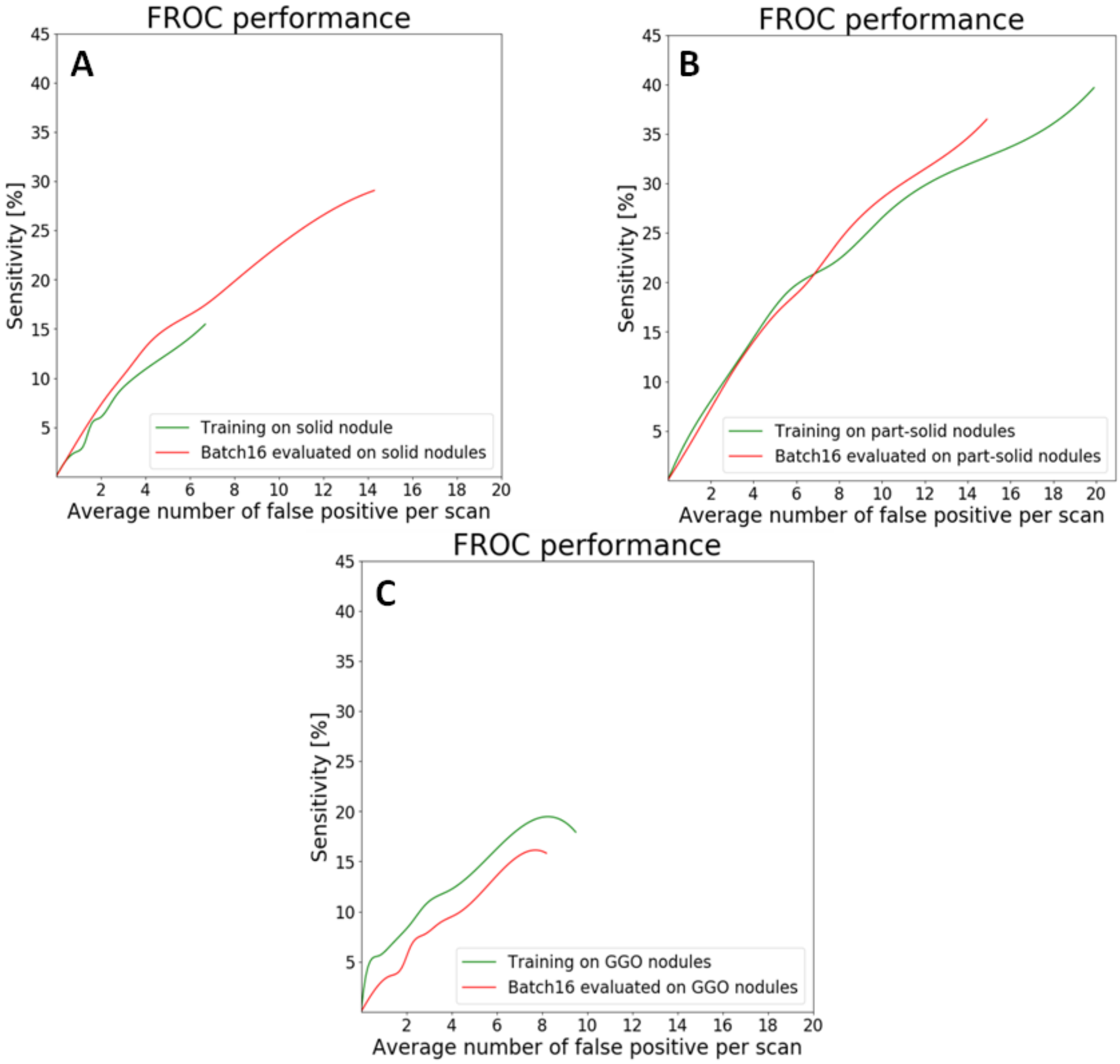


Figure 39. *In red is represented the result of batch-16 model(Section 3.1.2) validated on only solid(A), part-solid(B) and GGO(C) nodules of size >16, while in green the network trained and validated on only solid(A),part-solid(B) and GGO(C) nodules of size >16.*

Moreover, in order to obtain a better understanding on the results, we compared the CPM scores of the two strategies for the three subclasses(Table 3).

CPM score	Solid nodules	Part-solid nodules	GGO nodules
Batch16 evaluated on the specific class	0.0627	0.0664	0.0499
Training on the specific class	0.0526	0.0728	0.0777

Table 3. *CPM scores relative to the experiments on different classes of nodules. In the first row are represented the CPM scores relative to Batch16 model(Section 3.1.2) evaluated on solid, part-solid and GGO nodules of size>16. In the second row are represented the CPM scores relative to the model trained on the specific classes of size>16.*

As shown in Table 2 no improvements were observed training the model with solid nodules as unique type of lesion(Figure 39A). This means that the network has already learnt the solid features and a mono-type training do not add additional information. On the contrary the performance is slightly lower, probably due to the reduction of images fed to the network.

In contrast to the observed results on solid specific model, training a model on GGO(Figure 39B) or part-solid(Figure 39C) nodules only, brought to an improvement of the performance on the specific class(Table 2). These results could be linked to the inability of the network, trained on all nodules, to learn the groundglass specific features, also present in part-solid nodules which, as stressed in (Section 1.2), contains both solid and groundglass features. As such, a mono-type training allowed the network to concentrate and learn groundglass features. The improvement in term of CPM score was more evident on GGO with respect to part-solid nodules, where solid features were already learnt by the algorithm and a mono-type training did not add additional information.

On Figure 40 a qualitative example is reported where the result of Batch 16 implementation is compared with Faster R-CNN trained only on GGO nodules.

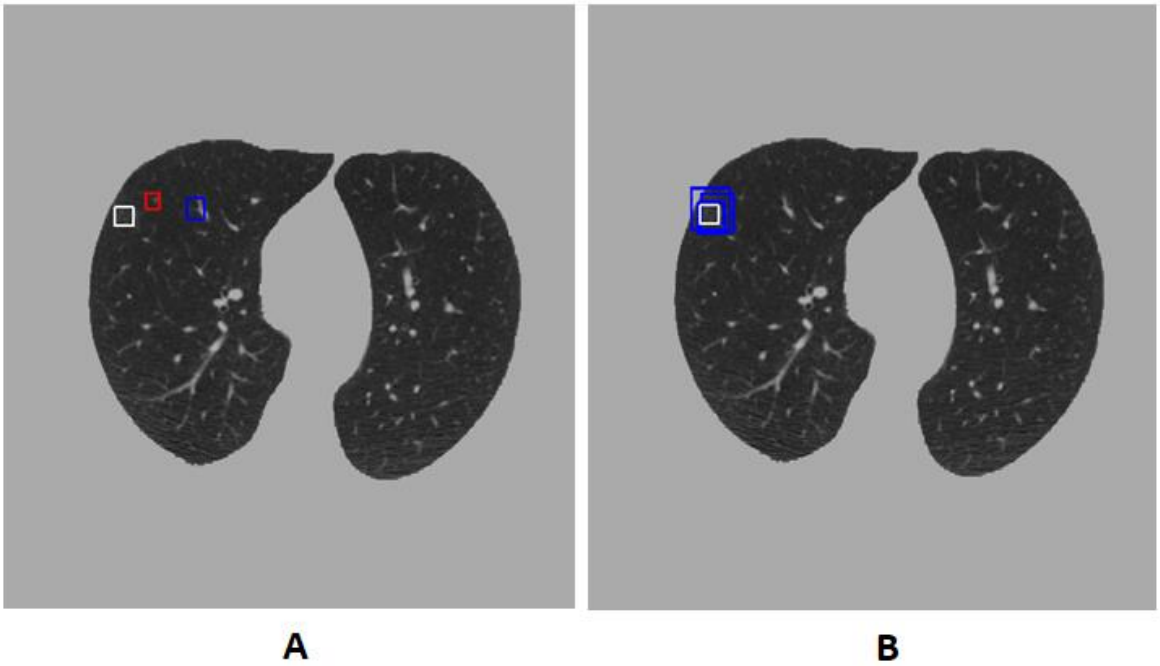


Figure 40. *Output of Batch 16 (Section 3.1.2) (A) and Faster R-CNN trained with only GGO nodules (B) on a GGO lesion of size>16. The ROIs output from a training on all classes are focused on different parenchyma's structures(A) and the GGO nodule is not recognized. In white is represented the ground truth box, while in green, red and blue the ROIs with probability(p) of being nodules respectively: p>0.9(green), 0.7<p<0.9(red) and 0.5<p<0.7(blue).*

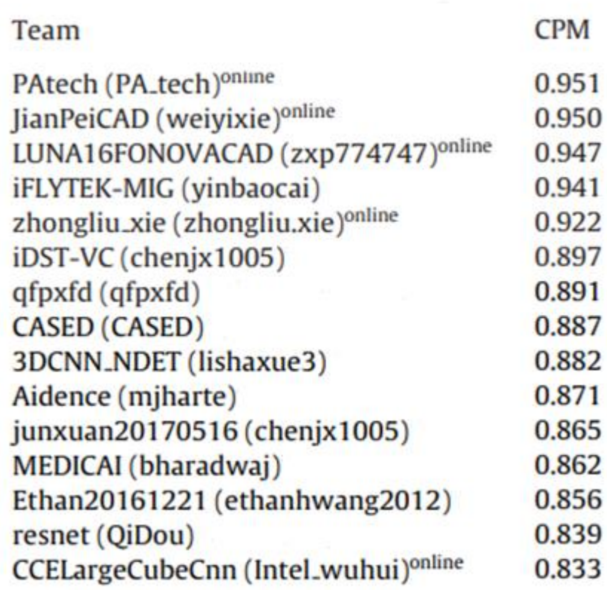
According to our findings, despite some of the CAD systems include GGO and part-solid nodules among the detected nodules, more efficient dedicated approaches have been also proposed not only to improve GGO detection , but also part-solid one, that also contains groundglass features. (Kim KG, 2005) (Ye X L. X., 2007).

However, the poor performance of our model trained including all the nodule classes can not be imputed just to the presence of non-solid nodules.

3.3. Performance comparison with literature's work

The technical results(Section 3.1) and the results on different class of nodules(Section 3.2) revealed the low performance of our implementation of Faster R-CNN. The obtained results in term of sensitivity are too low for including the implemented detection system in the clinical practice.

From Figure 41 we can appreciate the CPM score of leader state-of-the-art nodule detection systems. Also the result relative to our most performing experiment(Section 3.1.4), that achieves a CPM score of 0.0789, is very far from the literature detection systems.



Team	CPM
PAtech (PA_tech)$^{\text{online}}$	0.951
JianPeiCAD (weiyixie)$^{\text{online}}$	0.950
LUNA16FONOVACAD (zxp774747)$^{\text{online}}$	0.947
iFLYTEK-MIG (yinbaocai)	0.941
zhongliu_xie (zhongliu.xie)$^{\text{online}}$	0.922
iDST-VC (chenjx1005)	0.897
qfpxfd (qfpxfd)	0.891
CASED (CASED)	0.887
3DCNN_NDET (lishaxue3)	0.882
Aidence (mjharte)	0.871
junxuan20170516 (chenjx1005)	0.865
MEDICAI (bharadwaj)	0.862
Ethan20161221 (ethanhwang2012)	0.856
resnet (QiDou)	0.839
CCELargeCubeCnn (Intel_wuhui)$^{\text{online}}$	0.833

Figure 41. *CPM score comparison of the state-of-the-art approaches. Note that "online" means models with online descriptions available on LUNA16 competition website: https://luna16.grand-challenge.org/Results/.*

In order to understand if our Faster R-CNN implementation is a suitable system for lung nodule detection, we need to investigate more on parameters optimization and especially different training strategy should be implemented(Appendix B). Only a complete analysis and evaluation of parameters, network's architecture and training modality will definitively establish the adequacy of our modified Faster R-CNN in the field of lung nodule detection.

3.4. Analysis on computational cost of Faster R-CNN

For computations, the graphical processing units (GPU) used were NVIDIA Quadro P5000. The Processor is an Intel Xeon W-2123 CPU@3.60 GHz and has a RAM of 64.0 GB.

In our implementation the training of Faster R-CNN was based on the separate training of three networks:VGG16,RPN and Fast R-CNN. Each of these network required different training time, which depended also on the network's structure and batch size.

VGG16 fine-tuning required a time of approximately 4 hours. Replacing GAP layer with Max-pooling layers and Flatten caused a small increase of training time in order of minutes, due to a higher number of features for the training. (Section 2.4.1.1)

The most significant change regarded the RPN, where the use of the image-centric sampling strategy, with respect of having a larger batch of 512 with multiple image features, brought to an increase of training time from approximately 6 hours to approximately 22 hours (Section 2.4.1.3).

Fast R-CNN required a longer time to train, nearly equal to 3 days. While changes of batch composition and dimension relative to Section 2.4.1.2 did not change considerably the training time, the introduction of Max-pooling layers decreased the computational cost by means of a reduction of the number of features, causing a reduction of training time in order of hours.

Chapter 4. Conclusions and future developments

The aim of this project consisted in the implementation and investigation of an automatic system for detection of pulmonary nodules to optimize lung cancer screening based on LDCT. A deep learning model was exploited and specifically Faster R-CNN was adopted, widespread architecture in the field of object detection. Faster R-CNN consists in two substructures: the RPN dedicated to generating a series of region proposals and finally the Fast-RCNN which associate each proposed ROI to an object. Both RPN and Fast R-CNN share a backbone network, dedicated to feature extraction. In our implementation VGG16 was trained separately and for this reason is considered as a separate structure from RPN and Fast R-CNN. Each part of the detection system was trained separately.

Our main aim was to evaluate the performance of Faster R-CNN in the field of lung nodule detection. Taking inspiration from the version proposed by Shaoqing Ren et al. (2017), the network was adapted to be more suitable for the dataset available following the literature approach on lung nodule detection and at the same time we investigated how different parameters and structure’s characteristics, not investigated by literature, can influence the detection. Moreover a separate section was dedicated to the comprehension of the behaviour of Faster R-CNN on different classes of lung nodules.

An initial preprocessing step was applied limiting the anatomical structures to the parenchyma in order to improve speed and accuracy of the detection network. VGG16, the network chosen for feature extraction, was modified to have a feature map more appropriate to small objects and to avoid overfitting due to the limited number of samples available. In contrast to the original Faster R-CNN (Shaoqing Ren, 2017), which utilizes all the five convolutional blocks of VGG16 Net, we followed the approach of Jia Ding et al. (2017), adding a deconvolutional layer in order to increase the feature map resolution adapting the network for a small object detection problem. We investigated different VGG16 architecture in order to avoid overfitting: the introduction of GAP layers instead of Max-pooling layers followed by Flatten layers, proved to be necessary(Section 2.4.1.1).

However, additional strategies can be tested regarding both (i) the backbone network structure and (ii) solutions to avoid overfitting. For what concerns the point (i), the training of the deconvolutional layer weights could be improved concatenating part of the features coming from the contraction path; strategy frequently adopted in architectures that includes a combination of contraction and expansion paths (Olaf Ronneberger, 2015). Additionally, other backbone architectures already used in literature like ResNet or U-net could be tested to replace VGG16 (Yanfeng, 2019) (Hao Tang, 2019). The principal issue of backbone neural network is the high risk of overfitting (ii), related to their depth and so to their high numbers of parameters to train. A solution to this problem could be the extension of the dataset with the inclusion of new images that could allow a training from scratch instead of the use of pre-trained weights.

We realized that a parameter that need to be further investigated in order to improve the performance of the network was the stride value which determined the size of the base anchors and thus the range of detectable object size. We saw that the network is not still able to detect small nodules (size<13 pixels), even if we added the reduction of the stride from 16 to 8[a.u.] by means of the addition of the deconvolutional layer (Jia Ding, 2017).

As explained in section 2.3.2, the stride value is limited by the feature map dimension and so by the strategy applied at the backbone network level. To manage the stride value, the adaptation of the backbone architecture to reach a proper feature map size is not the only solution. Another simple trick could be that of enlarging the field of view of the network by increasing the input image resolution, but also more complex solutions have been proposed in the literature: the use of feature pyramid network(FPN) demonstrated to be an efficient strategy (Tsung-Yi Lin, 2017) to improve RPN. Taking multiple feature map from different depth of the backbone network, feature pyramid networks allow combining features with higher level of abstraction and reduced field of view, with more simple features where a larger field of view is preserved.

We noted that also the RPN training strategy can affect the network performance. The application of the image-centric sampling strategy, where each batch contains samples coming from a single image, brought to an improvement in terms of sensitivity with respect of having a larger batch of 512 samples with multiple image features(section 2.4.1.3). Even if a reduction of the batch size increased the training time(Section 3.4), the performance improved.

RPN provides as output a set of regions of interest (ROIs), along with the probability of the ROI of being background or foreground. However, the RPN makes a high-level estimation of regions that can contain the objects of interest. Following the reference code implementation (https://github.com/dongjk/faster_rcnn_keras, https://github.com/you359/Keras-FasterRCNN), we applied a four-step filtering (section 2.3.3) that reduces the initial number of ROIs, given then as input to Fast-RCNN, from 63504 to 300 ROIs.

This final number of ROIs is a parameter that have to be investigated in the future in order to verify if a more proper number instead of 300 exists in case of pulmonary nodules detection.

In the Fast R-CNN, the proposals generated by RPN are warped into squares by means of ROI pooling layer and finally input to the dense layers (Section 2.3.3). For each proposal Fast R-CNN return two outputs: the probability related to the different class (nodule vs. non-nodule) and the adjusted coordinates to better fit the lung nodule box coordinates.

The small object detection is surely the main issue in the construction of Fast R-CNN and opens some problematics also in the implementation of this part of the detection system. Specifically, the application of the ROI pooling layer, fixed to 7x7 pixels in Shaoqing Ren et al. (2017), lays the foundation on the dimension of the ROI with respect to feature map reference frame.

As already mentioned above,by means of the insertion of the deconvolutional layer in the feature extractor network, we fixed the stride value to 8[a.u.], followed the approach of Jia Ding et al. (2017). Although the reduction of the stride with respect to the original implementation stride which was equal to 16[a.u.], the size of tumors with respect to the feature map reference frame in our dataset is always inferior to 7x7 pixels. For this reason in the first instance we applied an interpolation to bring the dimension of these small ROIs equal to that fixed in the original implementation (Shaoqing Ren, 2017) but without applying the Max-pooling. We faced the problem by reducing the original 7x7 pixels dimension through a lower interpolation and with the insertion of a Max-pooling operation. Using 3x3 ROI lead to an improvement of the performance, while the additional reduction of the dimension of ROI did not improve the results(Section 2.4.1.4). The results of these experiments express furtherly the need for reducing the stride value in order to increase the

ROIs dimension with respect to the feature map reference frame and implement a real ROI pooling layer without a reduction of the ROI's size fixed by Shaoqing et al. (2017).

We have also tested different batch composition of Fast R-CNN to balance negative and positive class (section 2.4.1.2). Training the Fast-RCNN with a fixed batch of 16 ROI samples achieves the best performance among the multiple experiments.

Afterwards we have investigated the performance of the network on the three type of nodules: solid, part-solid and GGO. As we expected, the network performed better on solid and part-solid with respect to GGO nodules. The worst performance could be related to a bias of the net towards a particular lesion class. So, we have defined three different datasets for solid, part-solid and non-solid nodules respectively and we have trained and validated the network on each class of nodule. In these experiments we have considered only nodule with box dimensions above 16 pixels in order to simplify the problem of small object detection and better focus the model's limitations with respect to the lesion type. Only training the network on GGO and part-solid, both characterized by groundglass features, led to a slight improvement on the corresponding class of nodules(Section 2.4.2). While in the case of solid no improvement can be appreciated. A possible solution could be the integration of a dedicated approach for GGO and part-solid nodules to improve the results on this class.

Despite the experiments and studies conducted in order to adapt Faster R-CNN for lung nodule detection, our network performance remains poor and far from the results of the literature(Section 3.3). Further studies on parameters, network's structure and especially on the training modality are needed in order to understand if our modified Faster R-CNN is a suitable method for lung nodule detection.

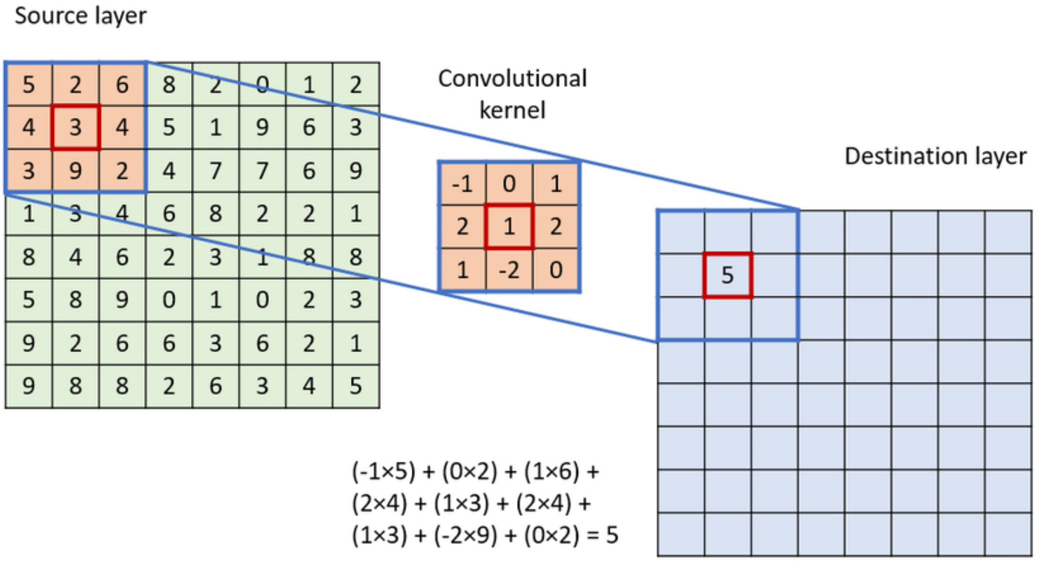
The general poor performance of our implementation could be improved by means of different structure and parameters optimization of the three networks, but more complex training techniques such as alternative training (appendix B) need to be tested. In this implementation the weights of the network are trained separately for VGG16, RPN and Fast R-CNN, in contrast to the original paper implementation. Here, RPN and Fast R-CNN do not share the convolutional layers of backbone neural network and, in this sense, there is no communication between the two networks. In a future implementation we will surely follow the four-step alternating training in order to improve the performance of the network.

Another approach that could be tested in the future was applied in the implementation from which we took inspiration (https://github.com/you359/Keras-FasterRCNN). It consists into an end-to-end training where the three networks are trained conjunctly. This approach is a simple technique that allows to share information between the three networks.

Future implementation could also take into account the integration of (i) merging operation of overlapping candidates in adjacent slices and (ii) false positive reduction step, in order to set up a complete CAD system.

Appendix A

Convolutional neural network (CNN) is a particular type of neural network designed for 2D images, although CNN can be used with 1D or 3D data. The most important and unique component of CNN is the convolutional layer, that performs a convolution operation on input image. Convolution levels extract ,by means of the use of filters, the features of the images whose content have to be analyzed. The filter is always smaller than the image analyzed and the multiplication applied between a filter-sized patch of the input image and the filter is a dot product. A dot product is the element-wise multiplication between the filter-sized patch of the input and filter, which is then summed, always obtaining a single value(Figure 42).



5	2	6	8	2	0	1	2
4	3	4	5	1	9	6	3
3	9	2	4	7	7	6	9
1	3	4	6	8	2	2	1
8	4	6	2	3	1	8	8
5	8	9	0	1	0	2	3
9	2	6	6	3	6	2	1
9	8	8	2	6	3	4	5

-1	0	1
2	1	2
1	-2	0

(-1×5) + (0×2) + (1×6) +
(2×4) + (1×3) + (2×4) +
(1×3) + (-2×9) + (0×2) = 5

Figure 42. *Convolution operations consists in element-wise multiplication between the filter-sized patch of the input and filter*

The aim of convolutional layers is to identify patterns, like curves, angles, circles or squares in the image with high precision. In the first level the filter represents a low level characteristic because it identifies simple objects such as curves or lines. In subsequent convolutional levels, the filters identifies more and more specific and sophisticated patterns.

The dimension of the kernel represent an important parameter to set inside the neural network. Also padding and stride deserve a particular attention.

Stride represent the amount of pixels by which the filter is shifted(Figure 43).

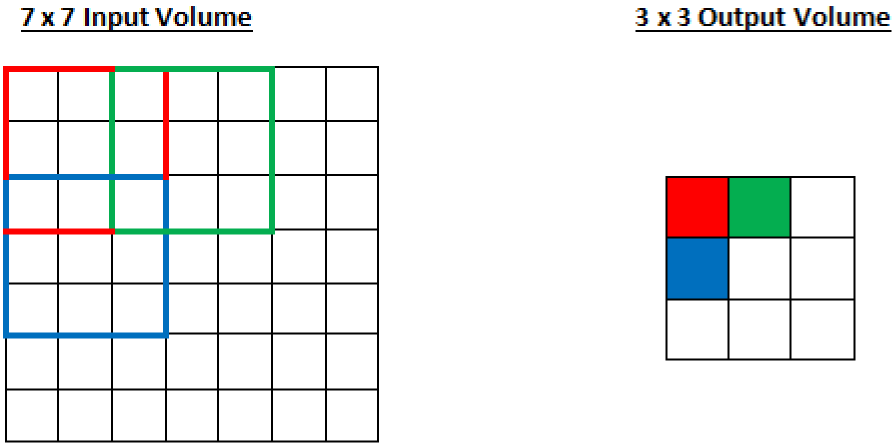
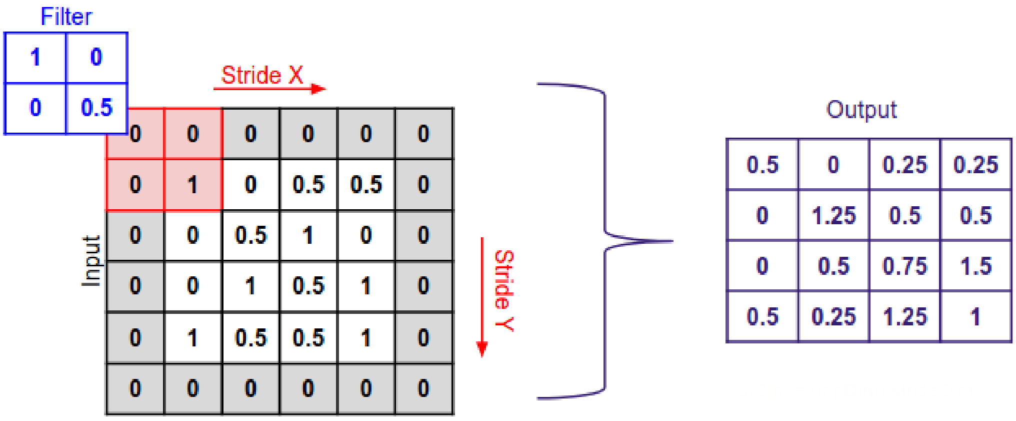


Figure 43. *The stride is equal to 2[a.u.], in fact the 3x3 kernel is moving by 2 pixel a time*

Padding consists in the insertion of additional layer at the border of an image in order to avoid the reduction of dimensionality that characterize a normal convolution and the consequent lost of information at the corners of the image. Typically, the values of the extra pixels are set to zero(zero padding)(Figure 44).



1	0
0	0.5

0	0	0	0	0	0
0	1	0	0.5	0.5	0
0	0	0.5	1	0	0
0	0	1	0.5	1	0
0	1	0.5	0.5	1	0
0	0	0	0	0	0

0.5	0	0.25	0.25
0	1.25	0.5	0.5
0	0.5	0.75	1.5
0.5	0.25	1.25	1

Figura 44. *Zero padding allows to maintain the same dimension*

Pool layers is applied after convolutional layer and reduce the dimensions of the 2D image by combining the outputs of neurons at one layer into a single neuron in the next layer.
The pooling layer acts on each feature map separately in order to form a new set of the same number of pooled feature maps.
Two common functions used in the pooling operation are:

- Average Pooling: compute the average value for each patch on the feature map.
- Maximum Pooling (or Max-pooling): compute the maximum value for each patch of the feature map(Figure 45).

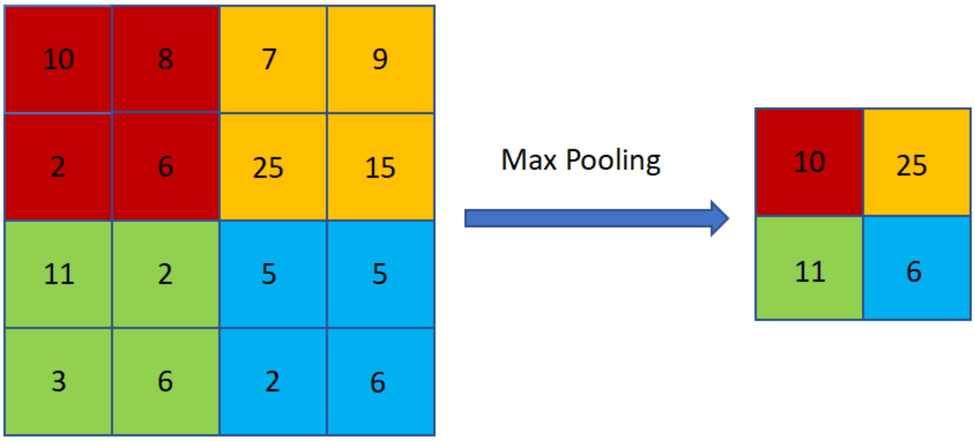


Figure 45. *Max pooling operation calculate the maximum value for each filter size patch of the original image*

Finally, fully connected layer takes place in the last part of the network. This level basically takes as input a vector(Input layer) and generates a dimensional vector N(Output) where N is the number of classes.

For instance, if you want a digit sorting program, N will be 10 since 10 are the numbers (0,1,2,3,4,5,6,7,8 and 9). Each number in this vector of dimension N represents the probability related to a certain class (Figure 46).

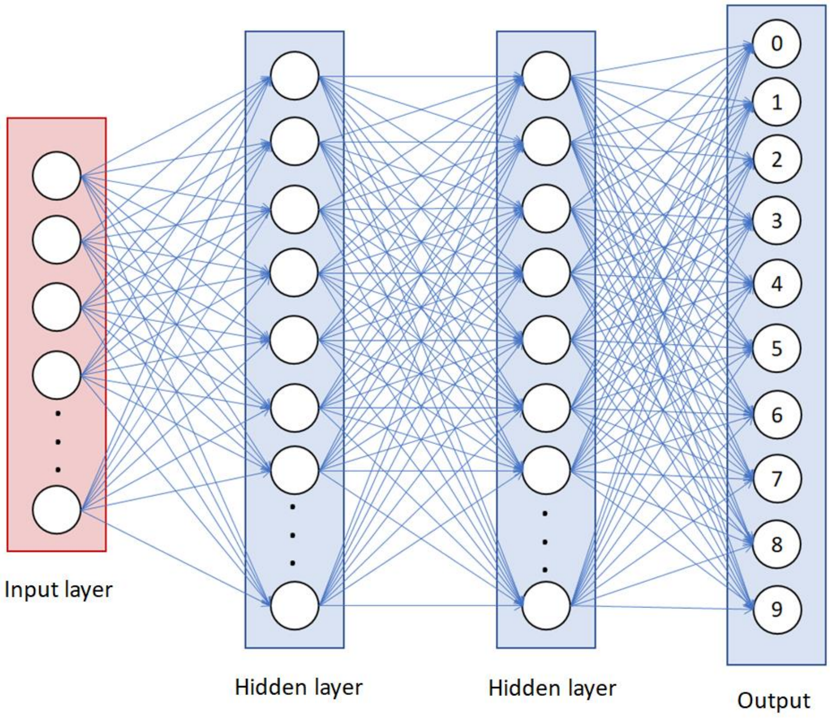


Figure 46. *Fully connected layers for digit classification*

Appendix B

In the original paper (Shaoqing Ren, 2017) is implemented a 4-Step Alternating Training(Figure 47), that consists in a 4-step training algorithm to learn shared features by means of alternating optimization. In the first step, RPN is trained. RPN is initialized with ImageNet-pre-trained weights and fine-tuned for the region proposal task. In the second step, Fast R-CNN is trained using the regions proposed by the RPN(first step). Also Fast R-CNN is initialized with ImageNet-pre-trained weights. Until now RPN and Fast R-CNN do not share convolutional layers. In the third step, Fast R-CNN is used to initialize RPN training procedure, but the shared convolutional layers are fixed and only the layers unique to RPN are fine-tuned. Now the two networks share convolutional layers. At last, keeping the shared convolutional layers fixed, the unique layers of Fast R-CNN are fine-tuned. RPN and Fast R-CNN share the same convolutional layers and constitute a unified network.

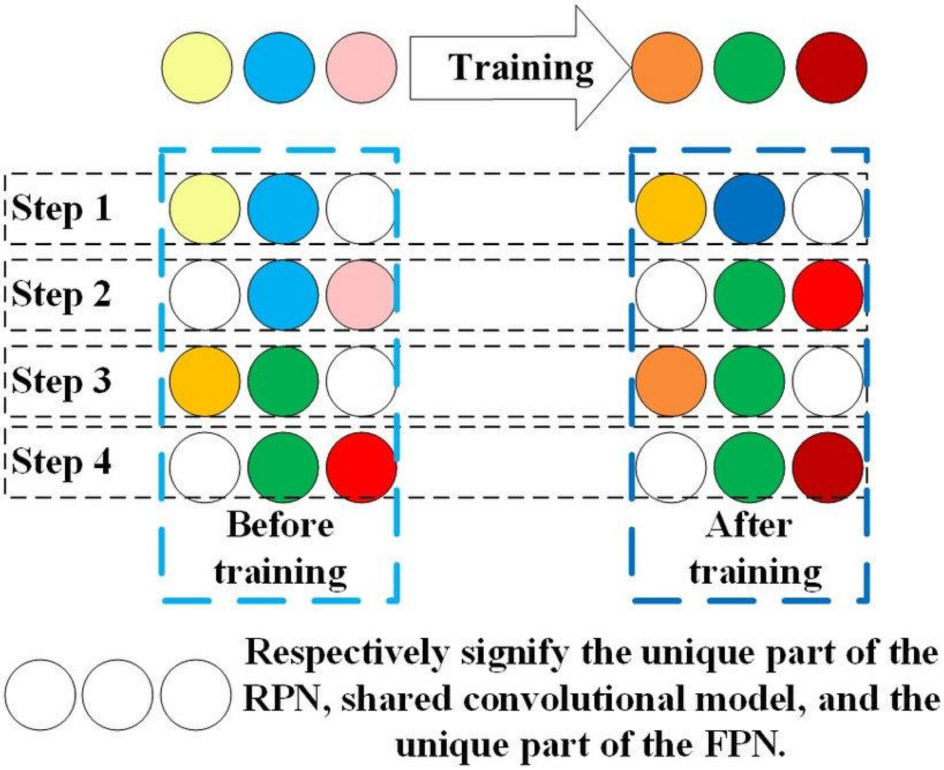

Training
Step 1
Step 2
Step 3
Step 4
Before training
After training
Respectively signify the unique part of the RPN, shared convolutional model, and the unique part of the FPN.

Figure 47. *Alternating training workflow for Faster R-CNN*

Bibliography

A. Riccardi, T. S. (2011). Computer-aided Detection of Lung Nodules via 3D Fast Radial Transform, Scale Space Representation, and Zernike MIP Classification . *Medical Physics*.

Aberle DR, A. A. (2011). Reduced lung-cancer mortality with low-dose computed tomographic screening.

Armato SG, G. M. (1999). Computerized detection of pulmonary nodules on CT scans. *Radiographics*.

Armato SG, M. G.-G. (2004). The Lung Image Database Consortium Research Group, F: Lung image database consortium: Developing a resource for the medical imaging research community1. *Radiology*.

Arnaud Arindra, A. T. (2016). Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the LUNA16 challenge.

Ashwin S, K. S. (2012). Efficient and reliable lung nodule detection using a neural network based computer aided diagnosis system;. pp. 135–142.

Ayman El-Baz, A. E.-G. (2013). Automatic Detection of 2D and 3D Lung Nodules in Chest Spiral CT Scans.

Broyelle, A. (2018). Automated Pulmonary Nodule detection on computed tomography images with 3d deep convolutional neural network.

Carole A. Ridge, A. Y.-P. (2015). Differentiating between Subsolid and Solid Pulmonary Nodules at CT: Inter- and Intraobserver Agreement between Experienced Thoracic Radiologists. *Radiology*.

Cascio D, M. R. (2012). Automatic detection of lung nodules in ct datasets based on stable 3d mass-spring models. *Comput Biol Med*.

Chao Tong, B. L. (2019). Pulmonary Nodule Detection Based on ISODATA-Improved Faster RCNN and 3D-CNN with Focal Loss.

Chen H, Z. J. (2012). Performance comparison of artificial neural network and logistic regression model for differentiating lung nodules on {CT} scans. *Expert Syst Appl*.

Council, N. R. (2016). . Health risks from exposure to low levels of ionizing radiation. *National accademy Press*.

Cristiano Rampinelli, 1. P. (2017). Exposure to low dose computed tomography for lung cancer screening and risk of cancer: secondary analysis of trial data and risk-benefit analysis.

Cristiano Rampinelli, D. O. (2102). Low-dose CT: technique, reading methods and image interpretation.

Da Silva GLF, D. S. (2017). Lung nodules diagnosis based on evolutionary convolutional neural network. *Multimedia Tools and Applications*, 19039–19055.

Diciotti S, L. S. (2010). The LoG characteristic scale: A consistent measurement of lung nodule size in ct imaging. *Med Imaging IEEE Trans*.

DJ., B. (2004). Radiation risks potentially associated with low-dose CT screening of adult smokers for lung cancer. *Radiology*.

Dou Q, C. H. (2016). Multilevel contextual 3-D CNNs for false positive reduction in pulmonary nodule detection. *EEE Transactions on Biomedical Engineering*, 1558–1567.

Fangzhou Liao, M. L. (2015). Evaluate the Malignancy of Pulmonary NodulesUsing the 3D Deep Leaky Noisy-or Network.

Fischbach F, K. F. (2003). Detection of pulmonary nodules by multislice computed tomography: improved detection rate with reduced slice thickness. *Eur Radiol*, 2378-2383.

Freddie Bray BSc, M. P. (2018). Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *acs journal*.

Gavrielides MA, Z. R. (2010). Information-theoretic approach for analyzing bias and variance in lung nodule size estimation with ct: A phantom study. *Med Imaging IEEE Trans.*

Giger, M. A. (1990). Computerized Detection of Pulmonary Nodules in Digital Chest Images - Use of Morphological Filters in Reducing False-Positive Detections. *Med Phys* .

Giger, M. D. (1988). Image Feature Analysis and Computer-Aided Diagnosis inDigital Radiography. 3. Automated Detection of Nodules in Peripheral Lung Fields. *Med Phys* .

Girshick, R. (2015). “Fast R-CNN”. *IEEE International Conference on Computer vision*.

Giulia Veronesi, M. P. (2014). Diagnostic Performance of Low-Dose Computed. 1.

Gomatrhi M, T. P. (2010). A computer aided diagnosis system for detectionof lung cancer nodules using extreme learning machine. *Int J Eng Sci Technol.*

Gulshan, V. P. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA*.

H Ashraf, A. D. (2010). *Combined use of positron emission tomography and volume doubling time in lung cancer screening with low dose ct scanning.*

Hao Tang, D. R. (2019). Pulmonary nodule detection using 3D deep convolutional neural networks.

Heber MacMahon, D. P.-P. (2017). Guidelines for Management of Incidental Pulmonary Nodules Detected on CT Images: From the Fleischner Society 2017. *Radiology*.

J.R.R. Uijlings, K. v. (2012). Selective Search for Object Recognition.

Jia Ding, A. L. (2017). Accurate Pulmonary Nodule Detection in Computed Tomography Images Using Deep Convolutional Neural Networks.

Kim KG, G. J. (2005). Computeraided diagnosis of localized ground-glass opacity in the lung at CT: initial experience. *Radiology 237(2)*, 657–661 .

Konstantinos Loverdos, A. F. (2019). Lung nodules: A comprehensive review on current approach and management.

Kumar SA, R. J. (2011). Robust and automated lung nodule diagnosis from ct images based on fuzzy systems. *Process Automation, Control and Computing (PACC),*.

Lee Y, H. T. (2001). Automated detection of pulmonary nodules in helical ct images based on an improved template-matching technique. *IEEE Trans Med Imaging*.

Liu Y, Y. J. (2010). A method of pulmonary nodule detection utilizing multiple support vector machines. *Computer Application and System Modeling (ICCASM)*.

Lo SC, F. M. (1993). Automatic lung nodule detection using profile matching and back-propagation neural network techniques. *Journal of Digital Imaging*.

Lodwick GS, K. T. (1963). The coding of roentgen images for computer analysis as applied to lung cancer. *Radiology*.

M. Callister, B. D. (2015). British Thoracic Society guidelines for the investigation and management of pulmonary nodules. *Thorax*.

M.Callister. (s.d.). The Fleischner Guideline / Lung-RADs. *Journal of Thoracic Oncology*.

Macedo Firmino, c. a. (2014). Computer-aided detection system for lung cancer in computed tomography scans: Review and future prospects.

Maisonneuve P, B. V. (2011). Lung cancer risk prediction to select smokers for screening CT--a model based on the italian COSMOS trial. *Cancer Prev Res (Phila)* .

Masaharu Sakamoto, H. N. (2016). Cascade neural networks with selective classifier and its evaluation usinglung x-ray CT-images.

Maurizio Infante, e. (2015). Long-term follow-up results of the DANTE trial, a randomized study of lung cancer screening with spiral computed tomography. *Am J Respir Crit Care Med.*

McCunney RJ, L. J. (2014). Radiation risks in lung cancer screening programs: a comparison with nuclear industry workers and atomic bomb survivors. *Chest* .

McKee BJ, H. J. (2014). Experience with a CT screening program for individuals at high risk for developing lung cancer. *Am Coll Radiol.*

McWilliams A, T. M. (2013). Probability of cancer inpulmonary nodules detected onfirst screening CT. *N Engl J Med.*

Messay T, H. R. (2010). A new computationally efficient {CAD} system for pulmonary nodule detection in {CT} imagery. . *Med Image Anal.*

Min Lin1, 2. Q. (2014). Network In Network.

Murphy K, S. A. (2007). Automated detection of pulmonary nodules from low-dose computed tomography scans using a two-stage classification system based on local image features. *Proc SPIE*.

Naidich DP, R. H. (1993). Variables affecting pulmonary nodule detection with computed tomography: evaluation with three-dimensional computer simulation. *J Thorac Imaging*, 291-299.

Olaf Ronneberger, P. F. (2015). U-Net: Convolutional Networks for Biomedical Image Segmemtation.

olei Zhou, A. K. (s.d.). Learning Deep Features for Discriminative Localization. 2924-2925.

Orozco HM, O. V. (2012). Lung nodule classification in frequency domain using support vector machines. *Information Science, Signal Processing and Their Applications (ISSPA)*.

Qiang Li, S. S. (s.d.). Selective enhancement filters for nodules, vessels, and airway walls in two- and three-dimensional CT scans. *Radiation imaging physics*.

Radiology, A. C. (s.d.). Lung CT Screening Reportingand Data System (Lung-RADS.

Retico, A. (2013). Computer-aided detection forpulmonary nodule identification:improving the radiologist’s performance *Imaging Med.*

Ross Girshick, J. D. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation.

Rubin GD, L. J. (2005). Pulmonary nodules on multidetector row CT scans: performance comparison of radiologists. *Radiology*, 274283.

S. G. Armato III, M. L. (1999). Three-dimensional approach to lung nodule detection in helical CT.

S. Matsumoto, Y. O. (2008). Computer-aided detection of lung nodules on multidetector row computed tomography using three-dimensional analysis of nodule candidates and their surroundings. *Radiation Medicine*.

S. Ozekes, O. O. (2008). Computerized Lung Nodule Detection Using 3D Feature Extraction and Learning Based Algorithms. *Journal of Medical Systems* .

Setio AAA, C. F. (2016). Pulmonary nodule detection in CT images: f alse positive reduction using multi-view convolutional networks. *IEEE Transactions on Medical Imaging*, 1160–1169.

Shao H, C. L. (2012). Shape-based computer-aided detection of lung nodules in thoracic ct images. *Computer Science and Network Technology (ICCSNT)*.

Shaoqing Ren, R. G. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence* .

Societ, F. (2017). guidelines for Management of incidental Pulmonary nodules Detected on cT images. *Radiology:*.

Suzuki K, I. S. (2003). Massive training artificial neural network (mtann) for reduction of false positives in computerized detection of lung nodules in low-dose computed tomography. *Med Phys.*, 1602–1617.

Swensen SJ, S. M. (1997). The probability of ma-lignancy in solitary pulmonary nodules. Application to small radio-logically indeterminate nodules. *Arch Intern Med*.

T. Hara, M. H. (2005). Nodule detection in 3D chest CT images using 2nd order autocorrelation features. *Engineering in Medicine and Biology 27th Annual Conference*.

Tajbakhsh N, S. K. (2016). Comparing two classes of end-to-end machine-learning models in lung nodule detection and classification: MTANNs vs. CNNs. *Pattern Recognition*, 476–486.

Tan M, D. R. (2011). A novel computer-aided lung nodule detection system for ct images. *Med Phys*.

Team, T. N. (2011). Reduced Lung-Cancer Mortality with Low-Dose Computed Tomographic Screening. *the new england journal of medicine*.

Teramoto A, F. H. (2013). Fast lung nodule detection in chest ct images using cylindrical nodule-enhancement filter. *Int J Comput Assist Radiol Surg*.

Thakur, R. (s.d.). Step-by-Step R-CNN Implementation From Scratch In Python.

Tsung-Yi Lin, P. D. (2017). Feature Pyramid Networks for Object Detection.

U. Pastorino, M. S. (2015). Prolonged lung cancer screening reduced 10-year mortality in the MILD trial: new confirmation of lung cancer screening efficacy.

Veronesi, G. (2007). Lung cancer screening with low-dose computed tomography: A non-invasive diagnostic protocol for baseline lung nodules.

W. Suiyuan, W. J. (2012). Pulmonary Nodules 3D Detection on Serial CT Scans. *Third Global Congress on Intelligent Systems*.

W.-J. Choi, T.-S. C. (2014). Automated pulmonary nodule detection based on three-dimensional shape-based feature descriptor. *Computer Methods and Programs in Biomedicine*.

Wang S, Z. M. (2017). Central focused convolutional neural networks: developing a data-driven model for lung nodule segmentation. *Medical Image Analysis*, 172–183.

Wood, D. E., Kazerooni, E., & Baum, S. L. (2015). Lung Cancer Screening, Version 1.2015.

Wormanns D, L. K. (2005). Detection of pulmonary nodules at multirow-detector CT: effectiveness of double reading to improve sensitivity at standard-dose and low-dose chest CT. *Eur Radiol*, 14-22.

Xia Huanga, W. S.-L. (2019). Fast and fully-automated detection and segmentation of pulmonary nodules in thoracic CT scans using deep convolutional neural networks. *Computerized Medical Imaging and Graphics*.

Xie Y, Z. J. (2017). Fusing texture, shape and deep model-learned information at decision level for automated classification of lung nodules on chest CT. *Information Fusion*, 102–110.

Xu X-W, D. K. (1997). Development of an improved CAD scheme for automated detection of lung nodules in digital chest images. *Med Phys.* .

Y. Mekada, T. K.-i.-i. (2003). Detection of small nodules from 3D chest X-ray CT images based on shape features. *International Congress Series* .

Yanfeng Li, L. Z. (2019). Lung Nodule Detection With Deep Learningin 3D Thoracic MR Images.

Yashin Dicente, O. A. (2015). Efficient and fully automatic segmentation of the lungsin CT volumes.

Ye X, L. X. (2007). Efficient computer-aided detection of groundglass opacity nodules in thoracic CT images. 4449–4452.

Ye X, L. X. (2009). Shape-based computer-aided detection of lung nodules in thoracic ct images. . *Biomed Eng IEEE Trans.* .

Ying Ru Zhao, X. X. (2011). NELSON lung cancer screening study. *Cancer imaging*.

Zisserman, K. S. (2015). Very deep convolutional networks forlarge-scale image recognition. *International Conference on LearningRepresentations*.

Ringraziamenti

Ringrazio Prof. Guido Baroni per avermi dato l'opportunità di trattare degli argomenti così interessanti.
Ringrazio Ing. Noemi Garau che mi ha seguito durante questi mesi e ringrazio Dott. Ing. Chiara Paganelli per avermi indirizzato e supportato in questo periodo.
Un ringraziamento spetta all'Istituto Europeo di Oncologia per lo scambio dei dati.
Ringrazio la mia famiglia e i miei amici per essermi sempre stati accanto.