Lung Cancer Detection from Computed Tomography (CT) Scans using Convolutional Neural Network

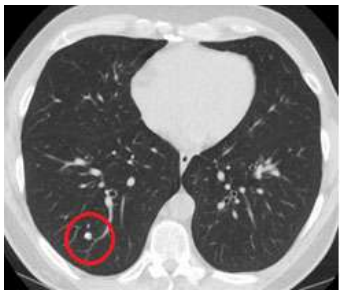
M.Bikromjit Khumancha*, Aarti Barai†, C.B. Rama Rao‡
Department of Electronics and Communication Engineering, National Institute of Technology Warangal
Email: *bikramjitk8@gmail.com, †aarti.barai.10@gmail.com, ‡cbrr@nitw.ac.in

***Abstract*—With increasing patients of Lung Cancer every year, it is important to detect Lung Cancer so as to give proper medical treatments. Low dose CT Scan imgages are used for the detection of Lung Cancer. The first step is to detect the pulmmonary nodules in Lungs. LUNA16 data has 888 CT scans with annotated nodules in the CT scans. The annotation has coordinates of the lung nodules. A $32 \times 32 \times 32$ cube is made around the nodules with nodule as center. A 3D Covolutional Neural Network (CNN) is used to detect nodules using these cubes. For Lung Cancer detection, Data Science Bowl 2017 kaggle competition data is used. It has 1595 CT scans. Lung nodules are predicted on this data using the nodule detector by running on the CT scans as grids. A ROI mask for lungs is apllied on the CT scan using Image Processing. The predicted nodules coordinates are used to make cubes around nodules as the same size as before and a second 3D CNN is used to predict cancer using it. The model achieves a precision and recall of 89.24% and 82.17% respectively.**

I. Introduction

One of the major reasonsfor non-accidental death is cancer. It has been proved that lung cancer is the topmost cause of cancer death in men and women worldwide. The death rate can be reduced if people go for early diagnosis so that suitable treatment can be administered by the clinicians within specified time. Cancer is, when a group of cells go irregular growth uncontrollably and lose balance to form malignant tumors which invades surrounding tissues. The stage of a cancer refers to how extensively it has metastasized. Stages 1 and 2 refer to cancers localized to the lungs and latter stages refer to cancers that have spread to other organs. Current diagnostic methods include biopsies and imaging, such as CT scans [1]. By raising an early flag, lung cancer can then be identified at a time when it is far more curable.

Traditionally, image processing algorithms have been used to extract features from images to distinguish between malignant and benign tumors. But distinguishing features between malignant and benign tumors is a complex task which also requires development of hand crafted features from the images [2,3]. A CNN has the ability to extract features by itself in a hierarchical mannerby use of multiple convolutional and max pooling layers [4]. The traditional methods were used on 2D slices of CT Scan. This,however has a drawback. Tha valuable spatial information or correltion between between slices is lost if we use only a single slice for detection.



(a) Cancerous Nodule.

Fig. 1: A slice of CT Scan containing cancerous nodule.

We aim to detect the presence of early or non-early stage cancer in CT scan image of Lungs with the help of Image Procesing and 3D CNN. The system will take the 3D CT Scans as input and give an output, whether the patient is cancerous or not. However, we cannot directly apply CNN on a CT Scan as it will be difficult to detect a small nodule in the large volume of 3D CT Scan. The solution is to reduce the search region by cropping out 3D cubes of small volume containing nodules from the CT Scan for training a nodule detector [5]. After the nodule detection is done, another 3D CNN cn be trained for identifiction of cncerous and non-cancerous nodules. Figure 1 shows a slice of CT Scan containing a cancerous nodule.

II. Data

A. Lung Nodule Detection

The data used for nodule detection is from a competition called LUNA16 Grand challenge [6]. It has 888 CT scans. It is a subset of the LIDC/IDRI dataset from the Cancer Imaging Archive. The dataset has scans in which the slices are less than or equal to 2.5 mm. It contains annotations which were done by 4 experienced radiologists. All nodules which are greater than 3 mm and accepted by 3 out of 4 radiologists has been included while nodules smaller than that has been excluded. It also has nodules for false positive rediuction. The annotation file is a csv file with each finding per line. Each line has SeriesInstanceUID of the scan, the x,y and z coordinate of the

nodule in world coordinates and the corresponding diameter in mm. The annotation has 408186 candidates.

B. Cancer Detection

The data used for Cancer Detection is from the Kaggle competition Data Science Bowl 2017 [7]. It has 1595 CT scans with an annotation file which has the patient ID and a value 1 or 0 according to whether the patient has cancer or not. The images are in DICOM format. The dataset has 1176 non-cancerous and 419 cancerous patients.

III. Proposed Methodology for Nodule Detection

A. Preprocessing

The first step of preprocessing is to make the CT scans homogenous. Every scan is rescaled so that each voxel represents a volume $1mm \times 1mm \times 1mm$. Then pixel values in each image is converted to Hounsfield units (HU), which is a measurement of radiodensity. Some CT scan scanners have cylindrical scanning bounds, but the ouput image is a square. Those pixels which fall outside this bound get the fixed value -2000. These pixels are given a value 1. This is done to make all the unecessary pixels black and to get the lung region properly. Now, the pixel values are scaled between 0 and 1. The next step is to make sure all the CT scans have the same orientation. The last step is to zero-center the data by subtracting the mean of all the images [8].

B. Creating Training Data for Nodule Detection

The annotations provided with the LUNA16 dataset contain 408186 candidates. Some candidates are also provided for False Positive reduction. The annotation has the coordinates of the candidate and diameter in mm. The number of annotated nodules in LUNA16 dataset is originally low (1186). So we used the LIDC/IDRI annotations to get nodules [9,10,11]. This increased the number of nodules to 5,000.

TABLE I: LUNA16 labelsets

Description	Quantity
Nodule candidates	5,000
Non-nodules	400,000
Candidates for False Positive reduction	7,000

The CT scan images generally have sizes around $200mm \times 200mm \times 200mm$ while our nodules in early stages have size less than 10mm. So the amount of signal vs noise is around 1:1000,000. Due to this, a 3D CNN will not be able to learn from raw image data. This problem is solved by cropping small 3D cubes around the candidate coordinates. The size of the cube used is $32 \times 32 \times 32$. Figure 2 illustrates the cropping of 3D cubes from CT Scan. The low amount of nodules is handled by doing data augmentation (two loss-less data augmentation used-3D flipping and translation) [12].

After performing data augmentation, our dataset finally has 20,000 nodules,4000000 non-nodules and 7,000 candidates for false positive reduction. These are randomly shuffled and divided into training set,cross-validation set and test set. The train set, validation set and test set consist of 32,138,04 , 803,452 and 9,744 candidates respectively. The training is done on the raw image cubes of the lung CT scan.

TABLE II: Architecture of the 3D CNN used for nodule detection.

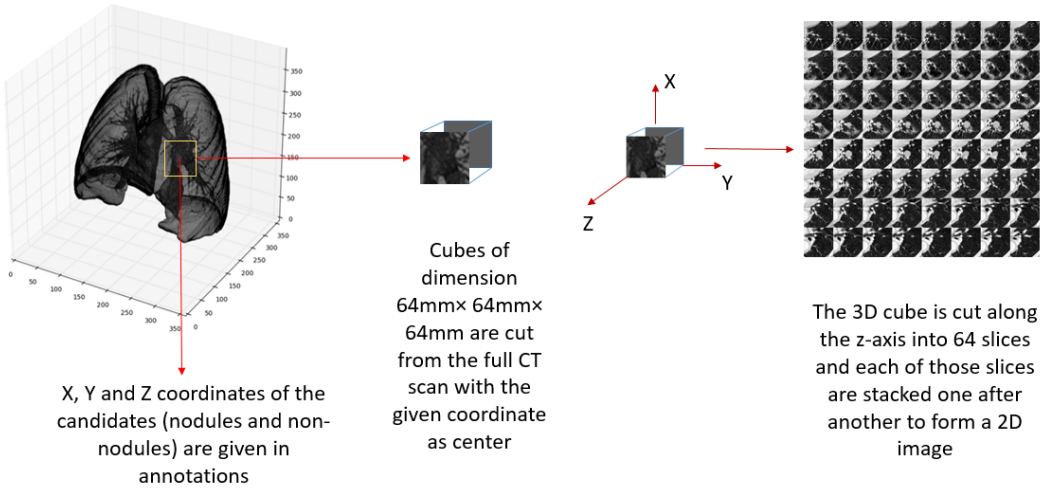
Layer	Parameters	Activation	Output
Input			$32 \times 32 \times 32$, 1
Average Pool	$2 \times 1 \times 1$		$16 \times 32 \times 32$, 1
3D Convolution	$3 \times 3 \times 3$	ReLU	$16 \times 32 \times 32$,64
Max Pooling	$1 \times 2 \times 2$		$16 \times 16 \times 16$,64
3D Convolution	$3 \times 3 \times 3$	ReLU	$16 \times 16 \times 16$,128
Max Pooling	$2 \times 2 \times 2$		$8 \times 8 \times 8$,128
3D Convolution	$3 \times 3 \times 3$	ReLU	$8 \times 8 \times 8$,256
3D Convolution	$3 \times 3 \times 3$	ReLU	$8 \times 8 \times 8$,256
Max Pooling	$3 \times 3 \times 3$		$4 \times 4 \times 4$,256
3D Convolution	$3 \times 3 \times 3$	ReLU	$4 \times 4 \times 4$,512
3D Convolution	$3 \times 3 \times 3$	ReLU	$4 \times 4 \times 4$,512
Max Pooling	$2 \times 2 \times 2$		$2 \times 2 \times 2$,512
3D Convolution	$2 \times 2 \times 2$	ReLU	$1 \times 1 \times 1$,64
3D Convolution	$2 \times 2 \times 2$	Sigmoid	$1 \times 1 \times 1$,1

A $64mm \times 64mm \times 64mm$ cube is shown in Figure 1. Each slice of the cube along the z-axis are stacked together and represented in a 8×8 grid image. Thickness of each slice is 1 mm. Figure 3:a is the image of a nodule. Figure 3:b is the image of a non-nodule. Each of these cubes are further cropped to $32mm \times 32mm \times 32mm$ before feeding to CNN.

C. Training the 3D CNN for Nodule Detection

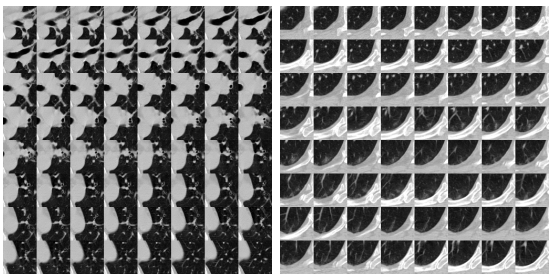
A 3D CNN is used for the nodule detection. The CNN has convolutinal layers with relu activation function for all except the last one. The last one has a sigmoid activation function which will gives the value of whether a candidate is a nodule or not. The model uses Convolutional layers with padding to keep the output size same as the input size. Max Pooling layers are used to downsample the input and reduce its dimension [13]. As we go deeper into the layers, the input size decreases and the number of filters used for each layer increases. The minibatch size used is 32. Since the data used for training is huge, generator is used to feed samples of the size of minibatch into the network without loading the whole dataset into the memory. The optimizer used is SGD (Stochastic Gradient Descent).Dropouts of 0.2,0.3 and 0.4 are used after second, third and fourth Max Pooling layer respectively, to prevent overfitting the model [14]. The Learning Rate initialized is 0.001, which is decreased to 0.0001 after fifth epoch using a learning rate scheduler. This deceasing of learning rate is done to converge our model properly to a local minima as higher learning rate might make the Gradient Descent skip the minima. The loss function used is Binary Crossentropy and the metrics used for validation are Binary Crossentropy and Binary Accuracy. The network is run for 12 epochs with the weights saved after each iteration in HDF5 format. The weights giving the best validation accuracy is finally chosen. The model is trained in Keras with Tensorflow backend in Google Colaboratory cloud.

A visualisation of the CNN architecture is shown in Figure 4 and its detailed description is given in Table II.



(a) Nodule.

Fig. 2: Illustration showing the cropping of 3D cubes of size $64mm \times 64mm \times 64mm$.



(a) Nodule. (b) Non-nodule.

Fig. 3: $64mm \times 64mm \times 64mm$ cubes cropped out from the lung region with the candidate as center.The slices along z axis are represented as a grid image. The white portion corresponds to lung walls.

TABLE III: Confusion Matrix of Lung Nodule detection.

		Prediction outcome		
		n	**p**	**total**
actual value	**n′**	TN=9233	FP=65	N′ = 9298
	p′	FN=47	TP=399	P′ = 446
	total	P=9280	N=464	

IV. Results of Nodule Detector

The trained model of nodule detector is tasted on 9744 candidates. The results obtained are shown in the confusion matrix shown in Table III.

The precision and recall obtained is 89.46% and 85.99% respectively. The trained network on LUNA16 dataset is prone to False Positive prediction.

V. Proposed Methodology for Cancer Detection

A. Predicting Nodules in Kaggle Dataset

The nodule detector CNN model obtained before is now used to detect nodules on the Kaggle dataset and after that the predicted nodules will be used for training a model to predict whether it is cancerous or not. Since our nodule detector predicted nodules in cubes of size $32mm \times 32mm \times 32mm$, we run the nodule detector through the whole 3D CT scan assuming it to be a grid made of cubes with strides of 12mm. The 12mm stride ensures that whole nodule is covered in any one of the cubes that go through the CT scan. If the probability of nodule is greater than a threshold(0.6), it is classified as a nodule, otherwise not. The predicted nodule coordinates (center of the cube in which the nodule is found) is saved for each CT scan in a CSV file. Thus for each patient, we have coordinates of the nodules. Now, the next task is to predict cancer using it. Figure 5 shows the predicted nodules by our trained nodule detector on 2D slices of CT Scan. The slice having the predicted z-coordinate is selected and a white circle encircling the nodule is drawn using the x and y coordinates.

B. Segmentation of CT Scan Images

Image segmentation, which aims at automated extraction of object boundary features, plays a fundamental role in understanding image content for searching and mining in medical image archives. A challenging problem is to segment regions with boundary insufficiencies, i.e., missing edges and/or lack

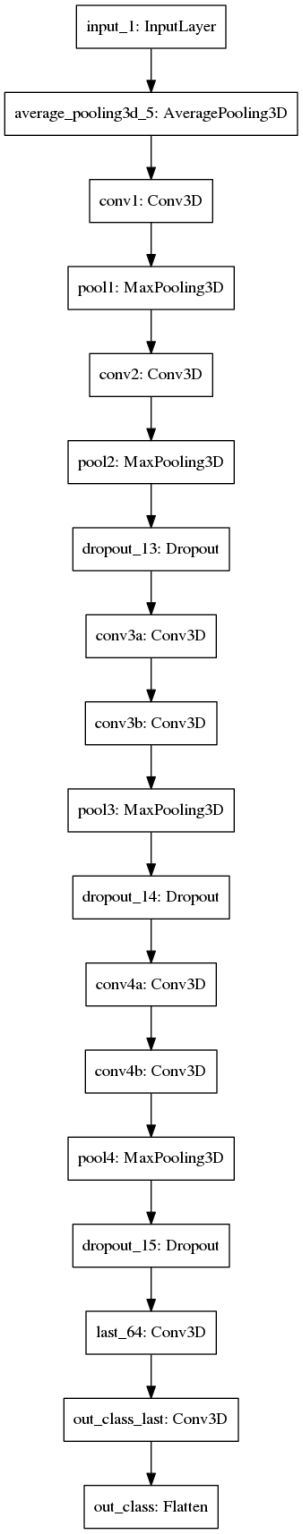
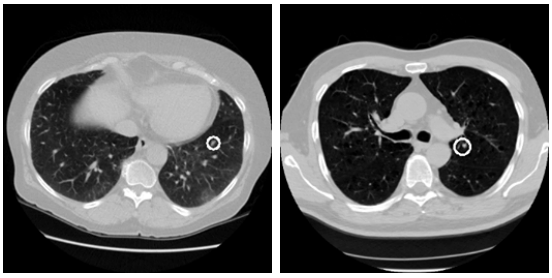


Fig. 4: Architecture of the 3D CNN used for nodule detection.

of texture contrast between regions of interest (ROIs) and background. The segmentation of lungs is an important step beacause the region of interests lies inside the lungs. Our main task is to create a mask to separate the lung region from



(a) p1. (b) p2.

Fig. 5: Nodules predicted by the nodule detector on 2D slices of CT scan (Encircled by a white circle).

the other areas of the image. The segmentation used has the following steps-

1) Binarization: The first step of segmentation is binarization.Binarization is the process of converting a pixel image to a binary image. First the image is converted to grayscale. An important task of this step is to select the threshold for binarization. It is found that Radiodensity(HU) for lung tissue is -500. So, a value of 604(-400 HU) is used as a threshold to separate lung First itemtissue from others [8]. The Radiodensity(HU) of different materials are given in Table IV.

TABLE IV: Radiodensity(in HU) of various substances in a CT Scan [15]

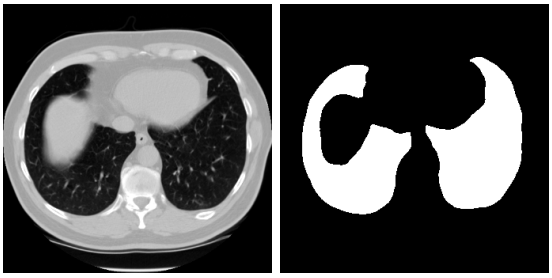
Substance	**Radiodensity(HU)**
Air	-1000
Lung Tissue	-500
Water and Blood	0
Bone	700

2) Clear Border: The next step is to remove the blobs connected to the boundary of the lungs. This done done using clear border.Clear border suppresses structures that are lighter than their surroundings and that are connected to the image border.

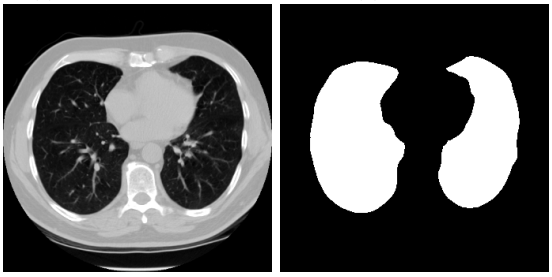
3) Labelling the Image: The next step is to label the image. A connected component in a binary image is a set of pixels that form a connected group. All the different connected components are assigned different labels. After labelling we keep the labels with the two largest areas.

4) Erosion: The next step is to perform erosion. Erosion is one of the popular techniques of morphological image processing. It is typically applied on binary images. The basic effect of the operator on a binary image is to erode away the boundaries of regions of foreground pixels (i.e. white pixels, typically). Thus areas of foreground pixels shrink in size, and holes within those areas become larger. This step is important beacause it separates the lung nodules from the blood vessels.

5) Closure Operation: The next operation is a Closing operation which is a dilation followed by an erosion. The basic effect of dilation on a binary image is to gradually enlarge the

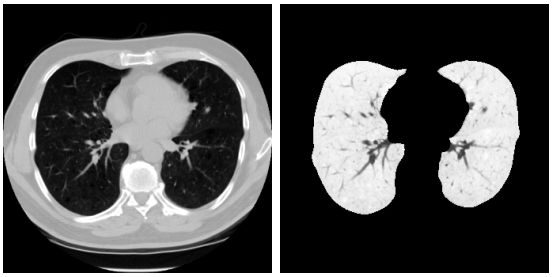


(a) CT Scan sclice A. (b) Mask A.



(c) CT Scan slice B (d) Mask B.

Fig. 6: Two slices of CT Scan and their corresponding masks.



(a) An original slice of CT Scan (b) Image after applying mask

Fig. 7: Application of mask on lung CT Scan

boundaries of regions of foreground pixels (i.e. white pixels, typically). Closing fills holes in the regions while keeping the initial region sizes. This step is done to keep the blobs connected to the walls of the lungs.

6) Filling Holes: The next step is to fill the holes inside the binary mask of lungs.

The Figure 6 shows original CT scan images and their corresponding masks obtained after segmentation.

C. Applying Mask on the Original CT Scan Image

The mask obtained in the previos step is multiplied with original CT Image for all the slices of the CT scan. Figure 7 shows the original image and image after applying mask on the slice.

D. Making Training Data for Cancer Detection

The coordinates that were saved after nodule detection in Kaggle Dataset are taken as center and training cubes of size $32mm \times 32mm \times 32mm$ are cropped from the images obtained after applying the masks. Figure 8 shows two such training

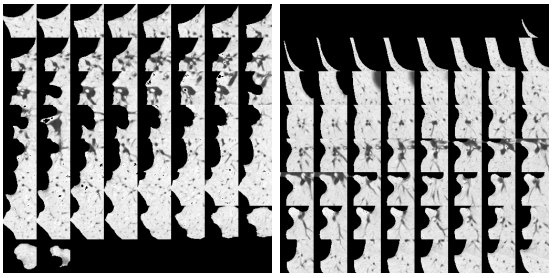


Fig. 8: Two $64mm \times 64mm \times 64mm$ cubes cropped out from the segmented image with the predicted coordinate as center.The slices along z axis are represented as a grid image. The white portion corresponds to lung walls.

cubes of size $64mm \times 64mm \times 64mm$ with slices along z-axis represented on a grid on images.

Since the nodule detector has high probability of False Positive prediction, the predicted nodules for each patient is arranged in descending order according to predicted probability by the nodule detector and the top four nodules with highest probabibily for each CT Scan are taken. The training ,validation and test set consist of 3,569, 608 and 608 nodules respectively.

E. Training a 3D CNN for Cancer Detection

The model for detection of cancer is another 3D CNN. The model used is not very deep and has less number of filters for each convolutional layer to prevent overfitting due to low amount of training data. The model uses ReLU activation function for all the convolutional layers except the last one. The last convolutional layer uses a Sigmoid activation function as it is used for making cancer predictions. Dropouts of 0.2 and 0.2 are used after the first and second Max Pooling layers respectively, to prevent overfitting the model [14]. The loss function used is Binary Crossentropy and the metric used for evaluation is Binary Accuracy. The optimizer used is an Adam Optimizer. The Learning Rate is initialized with 0.001 and a time-based decay is used to reduce the Learning Rate after each epoch. This helps the model converge properly to a local minima. The size of the minibatch used is 16. A Generator is used to feed the data of minibatch size to the model without loading the whole training and validation dataset into the memory. The model is trained for 12 epochs. The architecture of the model is shown in Figure 7 and the detailed description is given in Table V.

VI. Results of Cancer Detection

The model is tested on a test set containing 608 nodules. It is found that the number of observations that are sampled in each iteration, the size of the so-called minibatch, effects the performance of the model. Different minibatch sizes 8,16,32,64 were tried, out of which minibatch size 16 performed best. Intially training with 30 epochs resulted in validation loss increasing after iterations around 10 indicating overfitting. So, after testing with different total number of epochs around 10, it

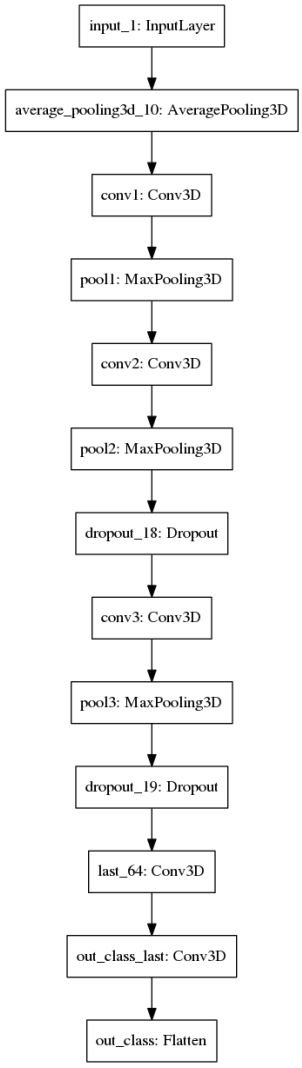


Fig. 9: Architecture of the 3D CNN for cancer detection.

TABLE V: Architecture of the 3D CNN used for cancer detection.

Layer	Parameters	Activation	Output
Input			$32 \times 32 \times 32$, 1
Average Pool	$2 \times 1 \times 1$		$16 \times 32 \times 32$, 1
3D Convolution	$3 \times 3 \times 3$	ReLU	$16 \times 32 \times 32$,32
Max Pooling	$1 \times 2 \times 2$		$16 \times 16 \times 16$,32
3D Convolution	$3 \times 3 \times 3$	ReLU	$16 \times 16 \times 16$,32
Max Pooling	$2 \times 2 \times 2$		$8 \times 8 \times 8$,32
3D Convolution	$3 \times 3 \times 3$	ReLU	$8 \times 8 \times 8$,32
Max Pooling	$2 \times 2 \times 2$		$4 \times 4 \times 4$,32
3D Convolution	$3 \times 3 \times 3$	ReLU	$2 \times 2 \times 2$,32
3D Convolution	$2 \times 2 \times 2$	Sigmoid	$1 \times 1 \times 1$,1

is found that 12 as number of epochs works the best. Different filter sizes are tried and it is found that a $2 \times 2 \times 2$ for

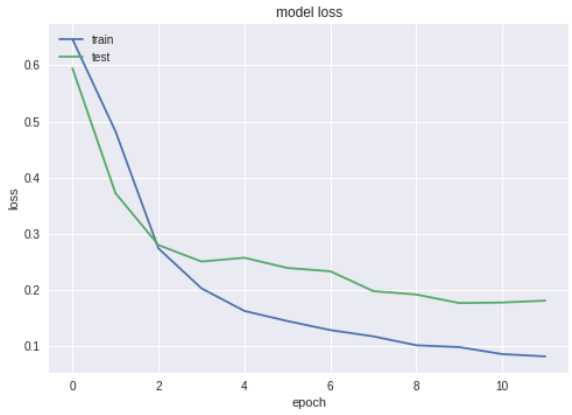


Fig. 10: Plot of loss function versus epoch for cancer detection.

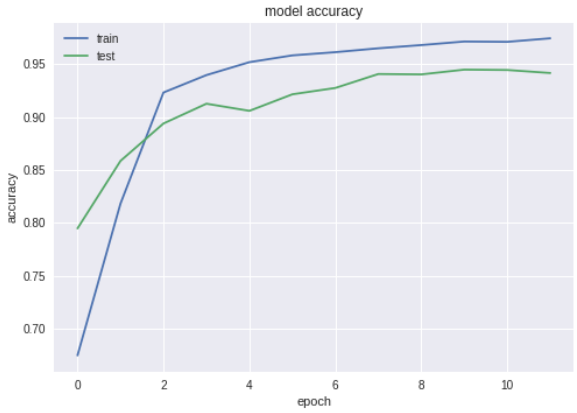


Fig. 11: Plot of accuracy versus epoch for cancer detection.

Max Pooling and $3 \times 3 \times 3$ for Convolutional layer works the best. Initializing with a very low learning rate along with decreasing it at each step and low number of filters for each layer also helped reduce overfitting. Our results suggest that the architecture of CNN that we have used is one of the optimal architecture for this task.

The plot of loss function versus epoch and accuracy versus epoch for training and validation set are given in the Figure 10 and Figure 11 respectively. From the loss function graph, it can be concluded that the model does not underfit as the loss is decreasing at each epoch. The gradual deacrese of loss at each epoch shows that proper selection of learning rate and model architecture prevented overftting. The confusion matrix is shown in the Table VI.

The performance of the system gives satisfactory results and the precision and recall of the model are 89.24% and 82.17% respectively.

VII. Conclusion

The 3D structure of nodules are used to extract features by 3D CNN to predict lung cancer. By using CNNs, the tedious task of manually extracting features can be eliminated. The

TABLE VI: Confusion Matrix of Lung Cancer detection.

		Prediction outcome n	**Prediction outcome** p	**Prediction outcome** total
actual value	**n′**	TN=386	FP=20	$N' = 406$
	p′	FN=36	TP=166	$P' = 202$
	total	P=422	N=186	

obtained precision and recall obtained for the model are decent considering that less labeled data than most state-of-the-art CAD systems are used. Further experimentations can be done by using widely used CNN architectures such as VCG [16], ResNet [17], and GooLeNet [18]. Also, the model is saved at the best validation accuracy. Other metrics, such as F1 can alsobe used. Other experimentations include making the network deeper and more extensive hyperparameter tuning. Finally, many models with different architectures and input sizes can be trained and an ensemble of these classifiers can be used to improve the cancer prediction.

VIII. References

References

[1] Tests and diagnosis, *http:// www.mayoclinic.org/diseases-conditions/lung-cancer/basics/tests-diagnosis/con-20025531*.

[2] B. v. Ginneken, A. A. A. Setio, C. Jacobs, and F. Ciompi, *Off-the-shelf convolutional neural network features for pulmonary nodule detection in computed tomography scans*, 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), 2015.

[3] J. Kuruvilla and K. Gunavathi, *Lung cancer classification using neural networks for CT images*,Computer Methods and Programs in Biomedicine, vol. 113, no. 1, pp. 202-209, 2014.

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, *Gradient-based learning applied to document recognition*, Proceedings of the IEEE, vol. 86, no. 11, pp. 22782324, 1998.

[5] J. d. Wit *2nd place solution for the 2017 national datascience bowl.*

[6] LUNA16, *Lung nodule analysis 2016. https://luna16.grand-challenge.org/.*

[7] Kaggle, Data Science Bowl 2017." *https://www.kaggle.com/c/data-science-bowl-2017.*

[8] A. Chon, N. Balachandar, and P. Lu , Data Science Bowl 2017." *Deep Convolutional Neural Networks for Lung Cancer Detection.*

[9] S. G. Armato III, G. McLennan, L. Bidaut, M. F. McNitt-Gray, C. R. Meyer, A. P. Reeves, B. Zhao, D. R. Aberle, C. I. Henschke, E. A. Hoffman et al. *he lung image database consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on ct scans*, Medical physics, vol. 38, no. 2, pp. 915931, 2011.

[10] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle et al. *The cancer imaging archive (tcia): maintaining and operating a public information repository*, 2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI), 2015.

[11] Data from lidc-idri,2015. *http://dx.doi.org/10.7937/K9/TCIA.2015.LO9QL9SX.*

[12] J. Lemley,S. Bazrafkan and P. Corcoran. *Smart Augmentation Learning an Optimal Data Augmentation Strategy*, arXiv:1703.08383v1 [cs.AI] 24 Mar, 2017.

[13] A. Krizhevsky, I. Sutskever, and G. E. Hinton. *mageNet Classification with Deep Convolutional Neural Networks.*

[14] N. Srivastava, G. Hinton ,A. Krizhevsky ,I. Sutskever ,and R. Salakhutdinov *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*, Journal of Machine Learning Research 15 (2014) 1929-1958.

[15] H. Lepor *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*, Prostatic Diseases, vol. 2000. W.B. Saunders Company, 2000.

[16] K. Simonyan, A. Zisserman *Very Deep Convolutional Networks for Large-Scale Image Recognition*, Computer Science, 2014.

[17] K. He, X. Zhang, S. Ren, Jian Sun *Deep Residual Learning for Image Recognition*, The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, 2016.

[18] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich *"Going Deeper With Convolutions*, The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1-9, 2015.