

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/336879291>

Lung Cancer Detection Based on CT Scan Images by Using Deep Transfer Learning

Article in *Traitement du Signal* · October 2019

DOI: 10.18280/ts.360406

CITATIONS

4

READS

645

3 authors:



Sajja Tulasi Krishna

Vignan University

5 PUBLICATIONS 14 CITATIONS

[SEE PROFILE](#)



Retz Mahima Devarapalli

Vignan University

3 PUBLICATIONS 7 CITATIONS

[SEE PROFILE](#)



Hemantha kumar Kalluri

Vignan University

40 PUBLICATIONS 74 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Biometrics [View project](#)

Lung Cancer Detection Based on CT Scan Images by Using Deep Transfer Learning

Tulasi Krishna Sajja, Retz Mahima Devarapalli, Hemantha Kumar Kalluri*

Vignan's Foundation for Science, Technology and Research Deemed to be University, Guntur 522213, India

Corresponding Author Email: khkcsevigman@gmail.com

<https://doi.org/10.18280/ts.360406>

Received: 13 April 2019

Accepted: 26 July 2019

Keywords:

convolutional neural network (CNN), lung cancer, transfer learning, AlexNet, GoogleNet, ResNet50

ABSTRACT

Lung cancer is the world's leading cause of cancer death. The convolutional neural network (CNN) has been proved able to classify between malignant and benign tissues on CT scan images. In this paper, a deep neural network is designed based on GoogleNet, a pre-trained CNN. To reduce the computing cost and avoid overfitting in network learning, the densely connected architecture of the proposed network was sparsified, with 60 % of all neurons deployed on dropout layers. The performance of the proposed network was verified through a simulation on a pre-processed CT scan image dataset: The Lung Image Database Consortium (LIDC) dataset, and compared with that of several pre-trained CNNs, namely, AlexNet, GoogleNet and ResNet50. The results show that our network achieved better classification accuracy than the contrastive networks.

1. INTRODUCTION

Lung cancer is one of the types of cancer; it causes abnormal growth of cells in the lungs. These types of cells are called malignant nodules. From the CT scan of lung images, deep learning techniques provide us with a method of automated analysis of patient scans. Globally, cancer is the major cause of death irrespective of gender. All types of cancers, Lung cancer dominates most cancer deaths [1]. Cancer leads to the transformation of regular cells into tumor cells in a multistage is a malignant tumor. The regular cells are grown uncontrollably and spread of abnormal cells. This uncontrollable development of tumors (cells) causes harmful cancer. This type of overgrowth that occurs in the lungs is called lung cancer. There are two main types of Lung cancer: small cell lung cancer and non-small cell lung cancer. Both types are lung cancer diseases; cancer cells form nodules in the lungs. Smoking is a major risk of lung cancer. Early detection and diagnosis of lung cancer may be life-saving. Fail to detect cancer cells in the lungs may spread cells to other areas of the body before a doctor detects them in the lungs. Screening with Low-Dose Spiral Computed Tomography (LDCT) has been shown to reduce lung cancer deaths. Proper treatment for lung cancer is based on whether the tumor is a small cell (13 %) or non-small cell (84 %).

An Automatic Detection System of Lung Nodules based on Multi-group Patch-Based Deep Learning Network [2], LIDC dataset is used where the input is multi-group 2D Lung CT Images. It involves three steps. Lung contours are repaired using a slope analysis method. Later, the vessel-like structure in a CT image is eliminated by applying the Frangi Filter. After that, the CNN structure is verified on two groups of images, one group contains original images, and the second group contains binary images generated through complex binarization processing to classify whether the nodule is cancerous or not. The researchers achieved 94 % sensitivity. Yang et al. [3] used the LIDC dataset. Based on the centroid

location of the malignant nodules, the researchers cropped the original images into smaller patches and used them as the cancer cases used Convolution Neural Networks. Fan et al. [4] proposed a method to detect nodules of lung CT Images using 3D Convolution Neural Networks along with traditional processing methods. The image is transferred from grayscale to color (RGB). Later, a series of morphological operations are performed. Finally, the connected area is the mask of the CT image. The researchers applied CNN and obtained 67.7 % accuracy.

Victor et al. [5] used deep transfer learning and obtained 88.41 % accuracy. Jan et al. [6] proposed a lung segmentation method based on morphological and circular filter. Later, CNN was used and achieved 84.6 % accuracy. Lyu et al. [7] developed Multi-Level CNN applied on the LIDC dataset and achieved 84.81 % accuracy.

Kumar et al. [8] proposed to use deep features extracted from an autoencoder along with a binary decision tree as a classifier to build a CAD system for lung cancer classification. Nodules are extracted from lung images using the information provided by different practitioners. The extracted nodules are then fed into autoencoder. Later features are extracted from layer four of the five-layer autoencoder. These features are then used for classification and achieved 75.01 % accuracy.

Several researchers have proposed different algorithms to detect lung cancer. Machine learning techniques have been used to detection and classification of the cancerous lesions in medical images, which can help radiologists make decisions, especially for the cases which are difficult to identify, improving the accuracy with efficiency. The literature survey [2-8] shows that there is a need to propose a method to improve accuracy by classification. The remaining part of the manuscript is organized as follows: In Section 2 describe the Convolutional Neural Network. Section 3 discusses the proposed method. Section 4 shows experimental results and discussion. Finally, the conclusion is made in Section 5.

2. CONVOLUTIONAL NEURAL NETWORK

In recent days deep learning [9] is one of the rising fields for classification and recognition. CNN is one of the most popular deep neural networks. The network contains the input layer, hidden layers, and the output layer. The hidden layer contains the convolutional layer, ReLU (Rectified Linear Unit), pooling layer, fully connected layers, and many more. By using these layers, the Convolutional Network was built. There are different pre-trained architecture models available in CNN [10] such as LeNet, AlexNet, GoogleNet, VGGNet, ResNet50, etc., The CNN architecture for lung cancer detection is shown in Figure 1.

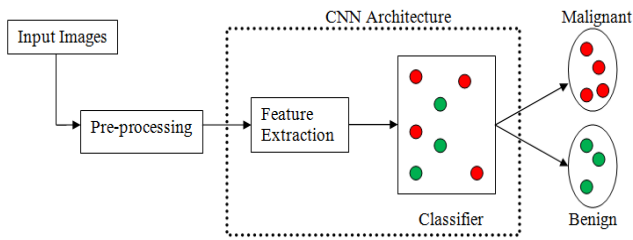


Figure 1. CNN architectures for lung cancer detection

In deep learning, the model trains with a large volume of data and learns model weight and bias during training. These weights are transferred to other network models for testing. The new network model can start with pre-trained weights [11]. A pre-trained model is already trained in the same domain. AlexNet is a much deeper neural network than the LeNet. In this network Rectified Linear Unit (ReLU) is used to add non-linearity, it speeds up the network. This network has five convolutional layers, three fully connected layers followed by the output layer, and also contains 62.3 million parameters. GoogleNet achieves good accuracy, but it required high computational power because the orders of calculations are very high. GoogleNet was replaced with average pooling after the last convolutional layer instead of fully-connected layers at the end; this will reduce the number of parameters. So far, while increasing the network depth automatically accuracy also increases. But some problems arise along with network depth in ResNet. The increased depth that required changing the weights, which raises the end of the network, the prediction becomes small at the initial layers. Another one is a huge parameter space it required. To prevent these problems residual modules come into the picture. ResNet50 and ResNet152 are example networks of ResNet.

3. PROPOSED NETWORK

AlexNet [12], GoogleNet [13], ResNet50 [14] are different architectures of the Convolutional Neural Network. By using these networks classify the CT scan images as benign or malignant. Each network contains an input layer, number of hidden layers, and an output layer [15]. The knowledge of these networks is utilized to classify the images effectively. This process is known as transfer learning [16].

AlexNet is the first Convolutional Network which contains eight layers; convolutional layer, ReLU, normalization, and max-pooling layer are set of layers, 5 sets of layers are used in this architecture followed with fully connected layers and dropout layers [17] finally softmax layer. This network

automatically extracts the distinctive features from input images and classifies the images. AlexNet was able to classify 1000 different classes; in this paper, this network is modified to classify the binary class, such as malignant or benign. This modified AlexNet classifies the images efficiently than existing methods.

GoogleNet contains 22 hidden layers. The depth of the neural network is larger than the AlexNet. Because of the increased depth, the network correctly classifies the samples more efficiently. This network also automatically extracts features from input images and classifies the images. GoogleNet was able to classify 1000 different classes; in this paper, this network is also modified to classify the binary class, such as malignant or benign. This modified GoogleNet classifies the images efficiently than existing methods.

ResNet50 contains 50 hidden layers. The depth of the network is larger than the GoogleNet. Because of the increased depth, the network correctly classifies the samples more efficiently. This network also automatically extracts features from input images and classifies the images. ResNet50 was able to classify 1000 different classes; in this paper, modified this network to classify the binary class as malignant or benign. This modified ResNet50 classifies the images efficiently than existing methods.

In this paper, the researchers proposed a network with convolutional layers, pooling, normalization layer, fully connected layer, and dropout layers. This network was built with so many layers; because of this plenty of layers, the network is considered a deep network. Moreover, bigger models lead overfitting and if we keep on increasing the layers computational cost is also increases with respect to layers. First, to reduce the computational cost, replace the densely connected architecture to sparsely connected architecture. Densely connected architecture means the network is built-in sequential order. Sparsely connected architecture means the network is built with the aggregation of some of the layers to minimize the number of input channels and reduce the number of convolutions. The dense and sparse networks are shown in Figure 2(a) and Figure 2(b), respectively.

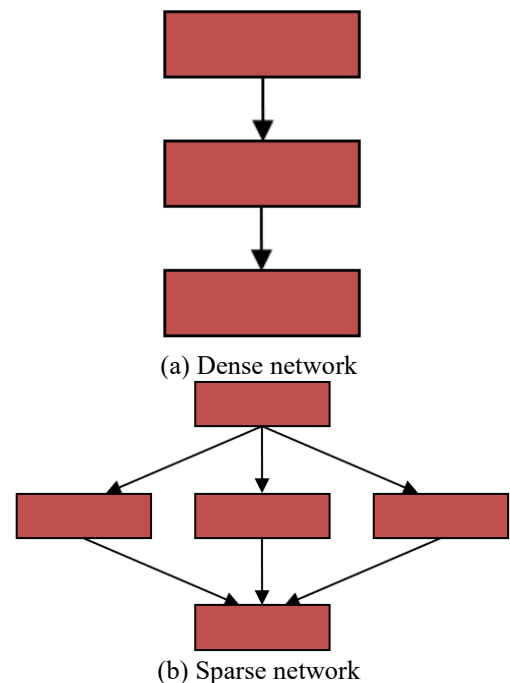


Figure 2. Network architectures

The proposed network is built with the help of a sparse network. The sparse network contains a total of 27 layers deep. The proposed architecture summary is shown in Figure 3. In between dropout and input image layer 22 layers are used which includes convolutional, max pooling and sparse layers, those are used for computation. In the proposed network, used

60 % dropout layer is present before the inception network because of this dropout layer avoids the overfitting.

The default dropout layer is 50 %, but in the proposed approach which was experimented with 60 % dropout neurons to reduce the over learning. The proposed approach achieves the highest classification accuracy.

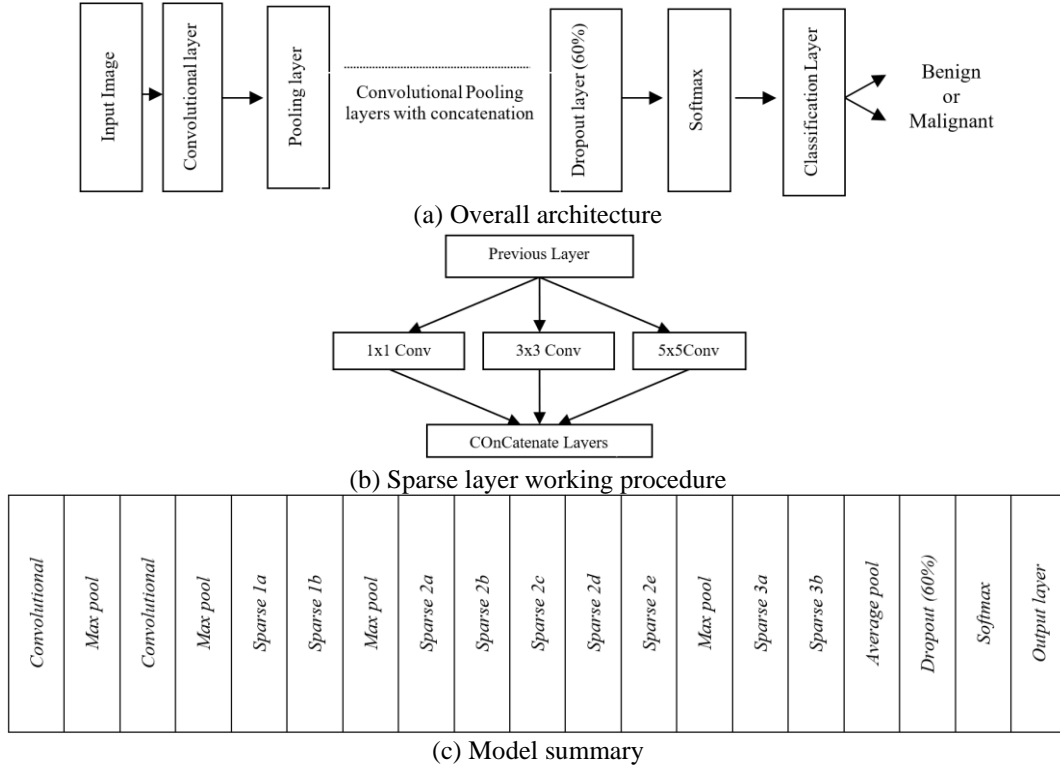


Figure 3. Proposed architecture

The working Procedure of each layer in the architecture is explained the following:

(1) Input Layer: This input layer accepts raw images and forwarded to further layers for extracting features.

(2) Convolution Layer: After the input layer, the next layer is the convolution layer. In this layer, the number of filters is applied to images for finding features from images. These features are used for calculating the matches at the testing phase.

Generally, the convolution is defined as a product of f and g object functions. The two function f and g over a range of $[0, t]$ is given in Eq. (1):

$$[f * g](t) = \int_0^t f(\tau)g(t - \tau)d\tau \quad (1)$$

where, $[f * g](t)$ indicates the convolution of f and g .

The Proposed network applied with input size as $227 \times 227 \times 3$ (color image). It uses 7×7 filters, with stride 2. After applying the convolution the output size is 111×111 . The convolution output size is computed using Eq. (2):

$$\left\lceil \frac{W-f+2p}{s} \right\rceil + 1 \quad (2)$$

where, $W \times H$ (Width \times Height) is 227×227 , filter (f) is 7×7 , stride (s) is 2 and padding (p) is 0. So, the output 111×111 is forwarded to the pooling layer. Similarly, all convolutional layers are computed inside the network. This computation is called abstract computation.

(3) Pooling: Extracted features are sent to the pooling layer.

This layer captures large images and reduces them, and reduces the parameters to preserve important information. It preserves the maximum value from each window. We applied the max-pooling; the output from the convolution layer is input matrix to pooling layer is calculated using Eq. (3):

$$\left\lceil \frac{I+2p-2}{s} \right\rceil + 1 \quad (3)$$

where, I is 111×111 , filter (f) is 3×3 , stride (s) is 1, and padding (p) is 0. To calculate the pooling layer output using Eq. (3). So, the output size from pooling layer is 56×56 (i.e. $\left\lceil \frac{111+0-2}{2} \right\rceil + 1$).

The same approach is applied for all pooling layers in the architecture.

(4) Sparse Layer: This layer is the combination of convolutional layers (conv 1×1 , conv 3×3 , conv 5×5) and the result from these layers are concatenated to the next layer of the model.

(5) Softmax Layer: This layer present just before the output layer. This layer gives the decimal probabilities to each class are calculated using Eq. (4). Those decimal probabilities are in between 0 and 1, which can predict n different classes, the feature will be stored into x , which is a column vector:

$$p(y = j|x, \theta) = \frac{e^{\theta_j^T x}}{\sum_{j=1}^k e^{\theta_j^T x}} \quad (4)$$

where, k is the target classes θ_j^T is a weight vector.

4. EXPERIMENTAL RESULTS AND DISCUSSION

4.1 LIDC dataset

To improve research and development activities, the Lung Image Database Consortium (LIDC) [18] was initiated by the National Cancer Institute (NCI). The LIDC database was created with three categories of objects to be marked by four radiologists: Nodules greater than or equal to 3 mm in diameter, of presumed histology, Nodules less than 3 mm in diameter of an indeterminate nature, non-Nodules that are less than 3 mm but are benign. The database contains 1008 patient records. The sample CT scan images of the LIDC dataset are shown in Figure 4.

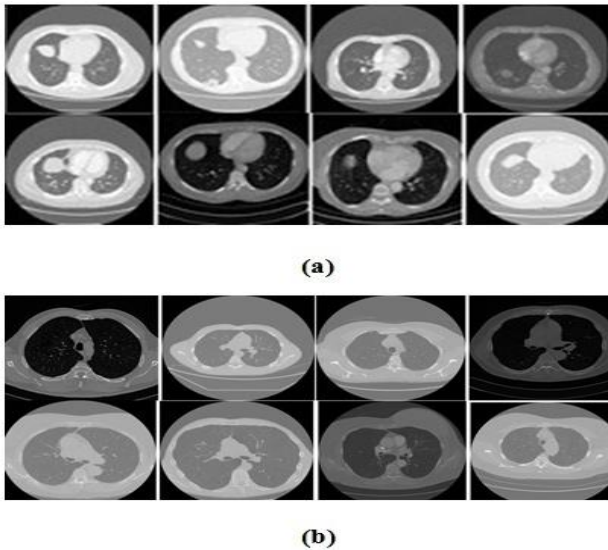


Figure 4. Sample images of a) Malignant b) benign from LIDC dataset

4.2 Experimental setup

4.2.1 Preprocessing

For this experiment, Lung Image Database Consortium (LIDC) dataset [19] is used. This data set contains the computed tomography (CT) scans of 1018 patients. It also contains an XML file for each patient, which contains the individual annotations marked by the four radiologists. To work with the dataset, several CT slices of each patient are retrieved based on the XML file and are placed in a directory. For the binary classification, the malignancy characteristics of the annotations given in the XML file is considered. If the Malignancy rating is greater than 3, all the patient slices are considered as malignant. And if the Malignancy rating is less than or equal to 3, those slices are considered as benign slices.

The LIDC dataset [19] images are available in DICOM format. These medical images are stored with the .dcm format. To do the experiment for effective classification of images, images are converted into the .jpg format along with the same labels. The converted images are stacked in two separate directories named as benign and malignant. No further preprocessing methods are applied on images; raw images are fed to the network directly.

4.2.2 Performance metrics

A confusion matrix is a very flexible and feasible visual representation of the performance of architecture with a binary class or multiclass. Basic terminology to measure the performance [20] of the model:

True Positive (TP): A CT scan image of a person is predicted as benign and its ground truth also benign.

True Negative (TN): A CT scan image of a person is predicted as Malignant and its ground truth also malignant.

False Positive (FP): A CT scan image of a person is predicted as Malignant but actually, it is benign.

False Negative (FN): A CT scan image of a person is predicted as benign but it is malignant.

Table 1. Accuracies for CNN networks with different samples

Network Architectures	80 % Training samples		90 % Training samples	
	Validation accuracy (%)	Testing accuracy (%)	Validation accuracy (%)	Testing accuracy (%)
AlexNet	100	89	100	90.87
GoogleNet	99.84	95.42	98	94
ResNet50	100	97.42	100	96
Proposed Net	100	99.03	100	99.00

Table 2. Comparison between existing systems with proposed system on LIDC Dataset

LIDC Database Information					Experimental Results					
S. No	No. of samples	Training	Testing	Authors	Existing Method	Results	AlexNet	GoogleNet	ResNet50	Proposed Method
					Filter, Convolution Neural Networks					
1	1006	90 %	10 %	Jiang et al. [2]		Sensitivity 94 %	Sensitivity 94.00 %	Sensitivity 98.00 %	Sensitivity 92.00 %	Sensitivity 100.00 %

Accuracy is the most common measure to evaluate the model. So, it is not only a suitable metric for model evaluation. Along with accuracy, sensitivity is also used, to measure the proposed model. Based on this confusion matrix in Table 3 to evaluate our proposed network is considered the following measures:

(1) Sensitivity: Measure the True values that are correctly predicted as true. It is also called as True Positive Rate, TPR, and calculated by using Eq. (5).

$$\text{Sensitivity} = \frac{TP}{TP + FN} \quad (5)$$

Table 3. Confusion matrix

	Benign	malignant
Benign	TP	FP
malignant	FN	TN

(2) Accuracy: Accuracy is the ratio of a number of correct predictions to the total number of predictions and it is calculated using Eq. (6).

$$\text{Accuracy} = \frac{TP+TN}{TP+FN+FP+TN} \quad (6)$$

4.2.3 Experimental results

Experimental work and analysis were carried over on MATLAB 2018a software. All experiments were run on a computer system with Windows 10 of the 64-bit operating system. The system consists of random-access memory (RAM) DDR4 of 32GB and Intel Xeon W-2135 3.726MHz 8.256c CPU. This system is working with Nvidia GEFORCE GTX1080Ti 11GB GDDR5X Graphics card. The experiments utilize the GPU capacity of the system.

The experiments were trained with the Stochastic Gradient Descent (SGD) algorithm [21] for optimization with initial Learning rate as 0.001, for every epoch the learning rate is decreased by 10 times. 25 epochs are used for training. First, experiments are done with AlexNet, while 80 % training and 20 % testing samples are used when the network gave 100 % validation accuracy, 89 % testing accuracy, again experiment is done with GoogleNet while 80 % training and 20 % testing samples are used when the network gave 99.84 % validation accuracy, 95.42 % testing accuracy. And also the experiments are done with ResNet50 while 80 % training and 20 % testing samples are used when the network gave 100 % validation accuracy, 97.42 % testing accuracy. Our proposed Net achieves 100 % validation accuracy and 99.03 % testing accuracy.

Likewise, the experiments were conducted with 90 % training and 10 % testing samples AlexNet, GoogleNet, ResNet50, ProposedNet achieved 100 %, 98 % 100 %, 100 % validation accuracies respectively and also the network architectures got 90.87 %, 94 %, 96 %, and 99 % testing accuracies respectively. Proposed Net achieves the highest classification accuracy than the AlexNet, GoogleNet, and ResNet50 which are mentioned in Table 1. From these results, we might know ResNet50 is the best among the pre-trained networks (AlexNet, GoogleNet, and ResNet50). The existing methods are placed in Table 2 for the purpose of comparison.

Jiang et al. [2] proposed a Filter, Convolutional Neural Networks method they tested on 1006 samples of LIDC dataset with 90 % training and 10 % testing they got 94 % sensitivity. The same number of training samples was used in GoogleNet, AlexNet, ResNet50, and the proposed network. The obtained results are 94 %, 98 %, 92 %, and 100 % sensitivity respectively. Victor et al. [5] used the CNN-ResNet50 with SVM-RBF and obtained 88.41 % accuracy on LIDC 1536 samples with 80 % training and 20 % testing. The same number of training samples was used with AlexNet, GoogleNet, ResNet50, and the proposed network. The obtained results are 89 %, 95.42 %, 97.42 % and 99.03 % accuracies, respectively.

5. CONCLUSIONS

In this paper, the researchers have proposed a deep neural network based on GoogleNet with a maximum dropout ratio to reduce the processing time. This network reduces the overfitting at the time of learning by using the dropout layer. In the proposed method, 60 % of neurons are at a fully connected layer, which is a higher drop than the existing GoogleNet. Experiments were conducted using the three pre-trained CNN architectures such as AlexNet, GoogleNet, and ResNet50 on LIDC pre-process dataset. Among the three pre-trained architectures, ResNet50 produced the highest accuracy. The proposed network achieved the highest accuracy than the pre-trained architectures and the state-of-the-art methods. In future the proposed network performance is test on different dropout ratios and without dropout and also need to verify the importance of the inception layers added to the network and how many inception layers are sufficient for achieve better performance.

REFERENCES

- [1] Bray, F., Ferlay, J., Soerjomataram, I., Rebecca, L.S., Torre L.A., Jemal, A. (2018). Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA: A Cancer Journal for The Clinician, 68(6): 394-424. <https://doi.org/10.3322/caac.21492>
- [2] Jiang, H., Qian, W., Gao, M., Li, Y. (2018). An automatic detection system of lung nodule based on multigroup patch-based deep learning network. IEEE Journal of Biomedical and Health Informatics, 22(4): 1227-1237. <https://doi.org/10.1109/JBHI.2017.2725903>
- [3] Yang, H., Yu, H., Wang, G. (2016). Deep learning for the classification of lung nodules. arXiv preprint, arXiv:1611.06651.
- [4] Fan, L., Xia, Z., Zhang, X., Feng, X. (2017). Lung nodule detection based on 3D convolutional neural networks. In the Frontiers and Advances in Data Science (FADS). 2017 International Conference on IEEE, Xi'an, China. <https://doi.org/10.1109/FADS.2017.8253184>
- [5] Victor, R., Peixoto, S., Pires, S., Silva, P., Pedrosa, P., Filho, R. (2018). Lung nodule classification via deep transfer learning in CT lung images. In 2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS), Karlstad, Sweden. <https://doi.org/10.1109/CBMS.2018.00050>
- [6] Jin, X., Zhang, Y., Jin, Q. (2016). Pulmonary nodule detection based on CT images using Convolution neural network. 2016 9th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China. <https://doi.org/10.1109/ISCID.2016.1053>
- [7] Lyu, J., Ling, S.H. (2018). Using multi-level convolutional neural network for classification of lung nodules on CT images. In 2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Honolulu, HI, USA, pp. 686-689. <https://doi.org/10.1109/EMBC.2018.8512376>
- [8] Kumar, D., Wong, A., Clausi, D.A. (2015). Lung nodule classification using deep features in CT images. 2015 12th Conference on Computer and Robot Vision, Halifax,

- NS, Canada. <https://doi.org/10.1109/CRV.2015.25>
- [9] Krishna, S.T., Kalluri, H.K. (2019). Deep learning and transfer learning approaches for image classification. *International Journal of Recent Technology and Engineering (IJRTE)*, 7(5S4): 427-432.
- [10] Alom, Z., Taha, T., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M., Essen, B., Awwal, A., Asari, V. (2018). The history began from AlexNet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164*.
- [11] Pan, S.J., Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and data Engineering*, 22(10): 1345-1359. <https://doi.org/10.1109/TKDE.2009.191>
- [12] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6): 84-90. <https://doi.org/10.1145/3065386>
- [13] Szegedy, C., Liu, W., Jia, Y.Q., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA. <https://doi.org/10.1109/CVPR.2015.7298594>
- [14] He, T., Zhang, Z., Zhang, H., Zhang, Z., Xie, J., Li, M. (2019). Bag of tricks for image classification with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 558-567.
- [15] Liu, Y.H. (2018). Feature extraction and image recognition with convolutional neural networks. *Journal of Physics: Conference Series*, 1087(6): 062032. <https://doi.org/10.1088/1742-6596/1087/6/062032>
- [16] Hussain, M., Bird, J.J., Faria, D.R. (2019). A study on CNN transfer learning for image classification. In *UK Workshop on Computational Intelligence*, Springer, Cham, pp. 191-202.
- [17] Yang, J., Yang, G. (2018). Modified convolutional neural network based on dropout and the stochastic gradient descent optimizer. *Algorithms*, 11(3): 28. <https://doi.org/10.3390/a11030028>
- [18] Fedorov, A., Hancock, M., Clunie, D., Brochhausen, M., Bona, J., Kirby, J., Freymann, J., Pieper, S., Aerts, S., Kikinis, R., Prior, F. (2018). Standardized representation of the LIDC annotations using DICOM., *PeerJ Preprint*, e27378v1. <https://doi.org/10.7287/peerj.preprints.27378v2>
- [19] LIDC-IDRI. database available <https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI>, accessed on 18 March, 2019.
- [20] Hossin, M., Sulaiman, M.N. (2015). A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process*, 5(2): 1-11. <https://doi.org/10.5121/ijdkp.2015.5201>
- [21] Cui, X., Zhang, W., Tüske, Z., Picheny, M. (2018). Evolutionary stochastic gradient descent for optimization of deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 6048-6058. <https://arxiv.org/abs/1810.06773>