International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 09 | Sept - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Deep Learning Approach for Lung Disease Detection by Image Processing

Aman Tiple[1,2,*], Shlok Kadam[2]

[1]*Vasishth Genomics Research Lab Pvt. Ltd. Bengaluru, India*
[2]*Institute of Biosciences and Technology, MGM University, Chh. Sambhajinagar, India*

---***---

Abstract - In this research, the detection of lung diseases is targeted for ILD (interstitial lung disease) emphysema consolidation and fibrosis by image processing approaches. High resolution CT scans of the lung are analyzed using a combination of several techniques that include deep learning models such as convolutional neural networks (CNNs), image segmentation, feature extraction and classification methods. CNNs improve the accuracy of a variety of manual image feature extraction methods for disease pattern recognition. In this study, transfer learning with pre-trained models is used to improve the classification of lung disease patterns in medical images. An impressive usage of U-Net for multi-scale feature extraction can refine the detection even more. This project demonstrates the potential for integrating these technologies into clinical settings, aiding radiologists in early diagnosis and improving patient outcomes.

***Key Words*:** ILD, CNN, Image Processing, CT,

1.INTRODUCTION

The phrase "interstitial lung disease" (ILD) refers to a broad category of lung conditions. The bulk of these conditions result in the lung tissues gradually becoming scarred, which ultimately causes the interstitial to thicken. ILD is a broad category of several lung diseases that frequently present with similar clinical symptoms and unclear patterns. ILD patterns are found using high-resolution computed tomography (HRCT) images of the lung field regions. Radiologists can more easily recognise patterns in high-resolution pictures of the pulmonary vasculature, airways, interstitial lung, and lung alveoli thanks to HRCT. ILD tissues can be divided into many patterns according on how they look. Emphysema, consolidation, and fibrosis are the three typical patterns that are the subject of this investigation. Even for seasoned radiologists, it is a taxing task to review every HRCT area with the same precision and efficiency. As a result, a screening tool could help the radiologist get a differential diagnosis by offering an alternative viewpoint [1].

Most previous research has concentrated on enhancing the quality of the hand-crafted features for pathological region identification. Several classifiers have been used to classify ILD patterns, including k nearest neighbours (kNN), artificial neural networks, 5 and support vector machines (SVM). Other classifiers that have been used include local binary patterns, histograms based, and scale-invariant feature transform [2]. The manual extraction of hand-crafted features is the drawback of picture categorisation with conventional methods.

The limitations of hand-crafted features in many medical imaging applications have been explored through the use of deep-learning techniques. Convolution neural networks (CNNs), when combined with current generation graphics cards and technological advancements, yield superior outcomes in the classification and segmentation of natural images. Patch-based classification is the main focus of most deep learning efforts [3].

The weight sharing between hidden layers was examined in the Van et al. study. With the aid of gradient descent, supervised training was carried out on a fully connected neural network. Six ILD patterns were categorised by Anthimopoulos et al. using a CNN with five convolutional layers and a filter size of after extracting patch sizes of. The network achieved approximately 85.5% accuracy [4,5].

Radiologists' annotated areas of interest (ROIs) are used to aid with patch-based classification. The procedure takes a long time and is not as clinically preferred as it may be. For radiologists, identifying patterns at the section level will be more beneficial. A pre-trained AlexNet model was employed in the study by Gao et al., 18 for fine-tuning purposes. As opposed to patch-based classification, the classification work was finished on the entire lung segment. The input photos were resized to match the artificially generated AlexNet architectural design, which made use of various attenuation windows [6].

The lowest quality level of photos can be analysed via image processing. If entropy is a measure of information, then these actions diminish rather than increase the probability of picture information content. Enhancing pixel intensity through discrete-to-digital image conversion, pixel segmentation, mathematical pixel operations, and higher-quality image reconstruction is the primary need for processing [7]. The first stage in image analysis is preprocessing of CT scans, which is then followed by the segmentation process and various morphological procedures to identify cancer spots or cells in the image. Additionally, it can be used to calculate the percentage of lung afflicted by cancer, or the extent to which the cancer is spreading. The use of morphological operations essentially involves comparing the dimensions and morphology of the cancerous cell to those of a normal cell. The images of the infected cells are then projected onto a greyscale image at the highest intensity [8].

For pulmonary imaging, high-resolution X-ray computed tomography (CT) is the gold standard. High spatial and high temporal resolution, good contrast resolution for the pulmonary structures and surrounding anatomy, and the capacity to capture an entire three-dimensional (3-D) volume of the human thorax in a single breath hold are all possible with CT, depending on the scanner technology [9]. Applications for pulmonary CT imaging include lung parenchyma density analysis, airway analysis, and analysis of the mechanics of the lung and diaphragm. Lung segmentation is a prerequisite for all of these uses of quantitative analysis. The number of volumetric studies of the lung is growing due to the introduction of multisite spiral CT scanners, so it's imperative to create quick, precise algorithms that require little to no human intervention to determine the exact borders of the lung [10].

Several teams have created methods for segmenting lung CT images with computer assistance. To estimate regional gas and tissue volumes in the lungs of normal people, manually drawn borders were employed. But manual techniques take a lot of work. Image segmentation in an X-ray CT can be guided by the inherent contrast between the surrounding high-density chest wall and the low-density lungs [11]. The boundaries of the left and right lungs were located using 2-D edge tracking. Others have used manually provided seed sites in 3-D region growing. Many semi-automatic methods involve some human intervention to modify the segmentation that results or choose threshold values [12,13].

Recently, a knowledge-based, automatic technique for segmenting CT images of the chest was introduced. Their technique uses low-level image processing techniques to be guided by anatomic knowledge stored in a semantic network. The majority of lung CT segmentation algorithms in the literature use pixel-based techniques. The first concept in pixel-based approaches is to remove bones and fat [14]. The lung parenchyma appears on CT scans as low-intensity pixels due to its extremely low density. This characteristic is used to keep the two lungs apart from the surrounding tissue. The ultimate objective of CAD systems is the diagnosis of lung illnesses, and numerous studies have demonstrated that CAD systems are successful at identifying small pulmonary nodules on CT. revealed that a CAD system was used to detect a significant portion of lung tumours that were overlooked [15]. In this study we are going to identify the lung diseases using various image processing techniques. The objectives of this study are:

- To detect disease using basic image processing technique,
- Usage the various tools to in image analysis
- To introduce fast Fourier transform

2. MATERIALS AND METHODOLOGY

2.1. Materials: Python programming language, Python libraries

- NumPy: It is a python library which is used for working with arrays and also has functions for working in domain of linear algebra, Fourier transform, and matrices.
- Matplotlib: Matplotlib is a comprehensive library for creating static, animated, and interactive visualization capabilities, plot scatter plot. Histogram and many more.
- Pandas: It has data structure for efficient data manipulation and has functions for analyzing, cleaning, exploring data and can handle wide range of data formats.
- Scikit-learn: It is most useful and robust library for machine learning in Python and also provides a selection of efficient tools for machine learning and statistical modeling.
- Open CV: Open-Source Library for Computer Vision. They offer a sophisticated user interface for functional photos and movies. OpenCV is frequently used for applications like object detection in images and videos, among many other things.

2.2. Creating Lung diseases dataset: It is a collection of medical images that can be used for various applications such as disease diagnosis, treatment planning, and research. Image processing techniques can be used to extract important features from these images and classify them into different categories based on the presence or absence of lung disease. For Dataset Creation:

- Collect the images: You can obtain medical images of the lungs from various sources such as hospitals, clinics, and research databases. Make sure the images are of high quality and resolution.
- Preprocessing: The first step in image processing is preprocessing. This involves removing noise, normalizing the images, and enhancing the contrast to improve image quality.
- Segmentation: The next step is to segment the lung region from the image. This can be done using various segmentation techniques such as thresholding, region growing, or active contours.
- Feature extraction: Once the lung region is segmented, features can be extracted from the image using various methods such as texture analysis, shape analysis, or intensity-based features.
- Classification: Finally, the extracted features can be used to classify the images into different categories based on the presence or absence of lung disease. Machine learning algorithms such as support vector machines (SVM) or convolutional neural networks (CNN) can be used for this purpose.

2.3 Description of the Dataset: This step involves the source and description of data, which forms the dataset that will be used to form a model. It includes Importing essential packages, loading images in Python environment using OpenCV library, listing directories within specific paths, Counting images in directories, Visualizing Distribution of Images. For images of lungs and possibly also have some pictures with people having lung disease and without.

2.4. Multi-scale Feature Extraction using U-Net & Model Training: A convolutional network, which we will refer to as the U-Net, is applied at multiple scales of a given image for feature extraction. Multi-scale feature extraction also allows to capture different details in images, which is essential for correct identifying of patterns related to lung disease. We first create a pre-trained model, by training U-Net on huge dataset. The thing that finally happened is the pre-trained model learning general features of data which can be transformed into a domain like lung infection detection in computer vision.

2.5. Transfer Learning with Lung Disease Images: Pre-trained model used is further trained using only the lungs images so that it learn a better representation for lung specific patterns and disease detection, Transfer Learning is making use of knowledge the model has learned from one task, to a new and separate related but similar domain.

2.6. Implementation: This last step consists of using the model that we trained to process new pictures and locate lung disease. This might also cover putting the model into an actual setting, let's say a clinical environment.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 09 | Sept - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

3. RESULTS

3.1. HPFs and LPFs Filters: After the initial imports, we define a 3x3 kernel and a 5x5 kernel, and then we load the image in grayscale. After that, we want to convolve the image with each of the kernels. There are several library functions available for such a purpose. NumPy provides the convolve function; however, it only accepts one-dimensional arrays. Although the convolution of multidimensional arrays can be achieved with NumPy, it would be a bit complex. SciPy's ndimage module provides another convolve function, which supports multidimensional arrays. Finally, OpenCV provides a filter2D function (for convolution with 2D arrays) and a sepFilter2D function (for the special case of a 2D kernel that can be decomposed into two one-dimensional kernels). The preceding code sample illustrates the ndimage.convolve function. We will use the cv2.filter2D function in other samples in the Custom kernels – getting convoluted section of this chapter. Our script proceeds by applying two HPFs with the two convolution kernels we defined. Finally, we also implement a different method of obtaining an HPF by applying a LPF and calculating the difference between the original image. Le ve start with the follow

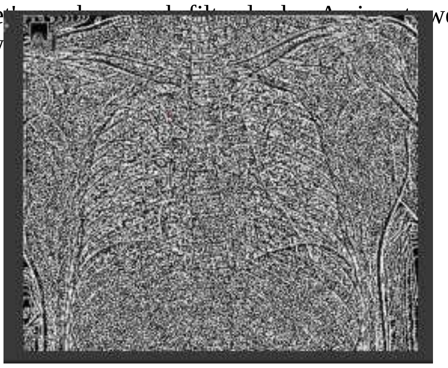

Fig -1 (a)

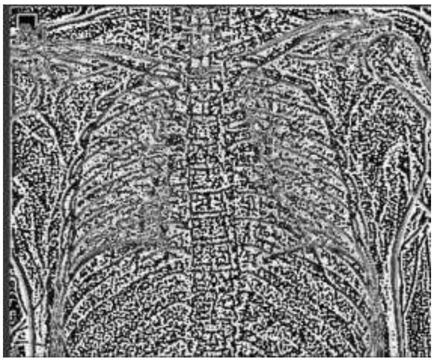

Fig -1 (b)

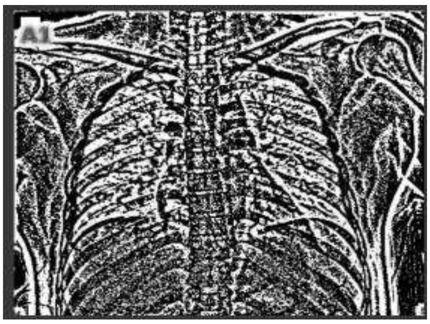

Fig -1 (b)

Fig -1: (a, b, c): Applying HPF and LPF filters on high resolution lung scans

3.2. Edge detection with Canny: OpenCV offers a handy function called Canny (after the algorithm's inventor, John F. Canny), which is very popular not only because of its effectiveness, but also because of the simplicity of its implementation in an OpenCV program since it is a one-liner:

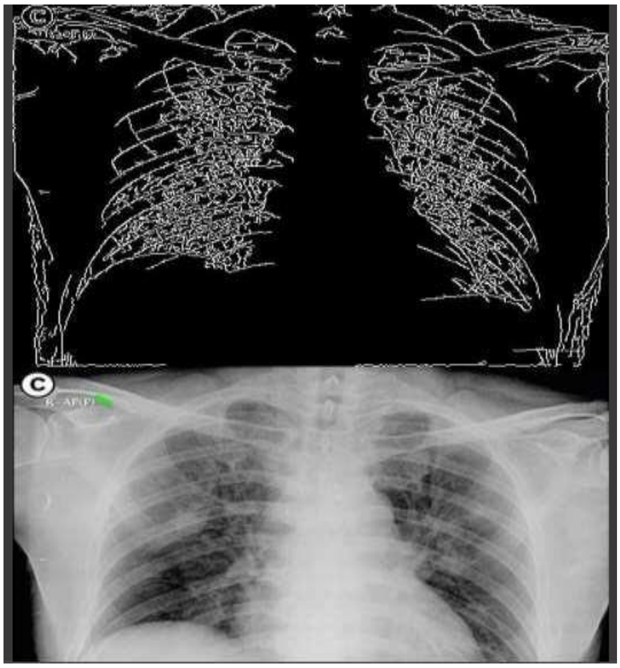

Fig -2: Identification of Edges

3.3. Bounding box, minimum area rectangle, and minimum enclosing circle: This function – like all of OpenCV's drawing functions – modifies the original image. Note that it takes an array of contours in its second parameter so that you can draw a number of contours in a single operation. Therefore, if you have a single set of points representing a contour polygon, you need to wrap these points in an array, exactly like we did with our box in the preceding example. The third parameter of this function specifies the index of the contours array that we want to draw: a value of -1 will draw all contours; otherwise, a contour at the specified index in the contours array (the second parameter) will be drawn. Most drawing functions take the color of the drawing (as a BGR tuple) and its thickness (in pixels) as the last two parameters. The last bounding contour we're going to examine is the minimum enclosing circle

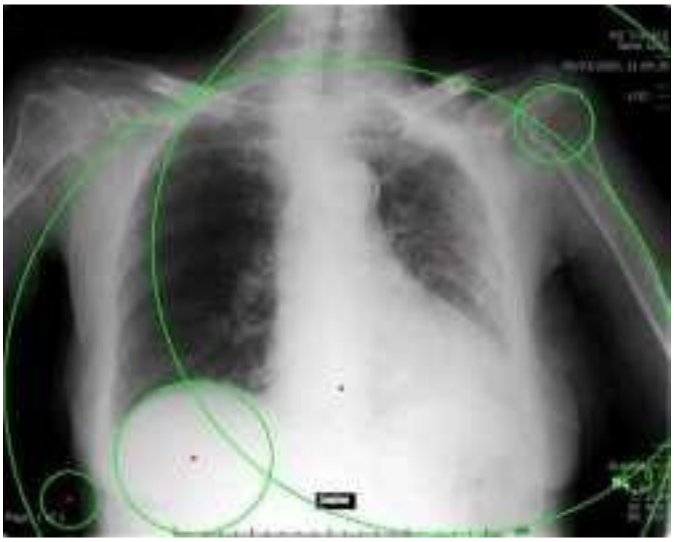

Fig -3: Clear lung Scan with minimum enclosing circle

3.4. Foreground detection with the Grab Cut algorithm: Calculating a disparity map is a useful way to segment the foreground and background of an image, but Stereos is not the only algorithm that can accomplish this and, in fact, Stereos is more about gathering three-dimensional information from two-dimensional pictures than anything else. Grab Cut, however, is a perfect tool for foreground/background segmentation.

The Grab Cut algorithm consists of the following steps:

- A rectangle including the subject(s) of the picture is defined.
- The area lying outside the rectangle is automatically defined as a background.
- The data contained in the background is used as a reference to distinguish background areas from foreground areas within the user-defined rectangle.
- A Gaussian Mixture Model (GMM) models the foreground and background, and labels undefined pixels as probable background and probable foreground.
- Each pixel in the image is virtually connected to the surrounding pixels through virtual edges, and each edge is assigned a probability of being foreground or background, based on how similar it is in color to the pixels surrounding it. The final part of our script displays the images side by side:

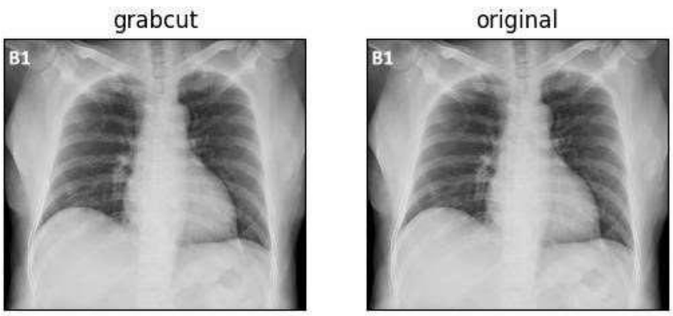

Fig -4: Foreground detection with the Grab Cut algorithm

3.5. Detecting Harris corners: Let's start by finding corners using the Harris corner detection algorithm. We will do this by implementing an example. If you continue to study OpenCV beyond this book, you will find that chessboards are a common subject of analysis in computer vision, partly because a checkered pattern is suited to many types of feature detection, and partly because chess is a popular pastime, especially in Russia, where many of OpenCV's developers live.

Fig -5: Foreground detection with the Grab Cut algorithm

3.6. Detecting DoG features and extracting SIFT descriptors: You will notice how the corners are a lot more condensed; however, even though we gained some corners, we lost others! In particular, let's examine the Variante Ascari chicane, which looks like a squiggle at the end of the part of the track that runs straight from northwest to southeast. In the larger version of the image, both the entrance and the apex of the double bend were detected as corners. In the smaller image, the apex is not detected as such. If we further reduce the image, at some scale, we will lose the entrance to that chicane too

This loss of features raises an issue; we need an algorithm that works regardless of the scale of the image. Enter Scale-Invariant Feature Transform (SIFT). While the name may sound a bit mysterious, now that we know what problem we are trying to solve, it actually makes sense. We need a function (a transform) that will detect features (a feature transform) and will not output different results depending on the scale of the image (a scale-invariant feature transform). Note that SIFT does not detect keypoints (which is done with the Difference of Gaussians (DoG); instead, it describes the region surrounding them by means of a feature vector.

A quick introduction to the DoG is in order. Previously, in Chapter 3, Processing Images with OpenCV, we talked about low pass filters and blurring operations, and specifically the cv2.GaussianBlur() function. DoG is the result of applying different Gaussian filters to the same image. Previously, we applied this type of technique for edge detection, and the idea is the same here. The final result of a DoG operation contains areas of interest (keypoints), which are then going to be described through SIFT.

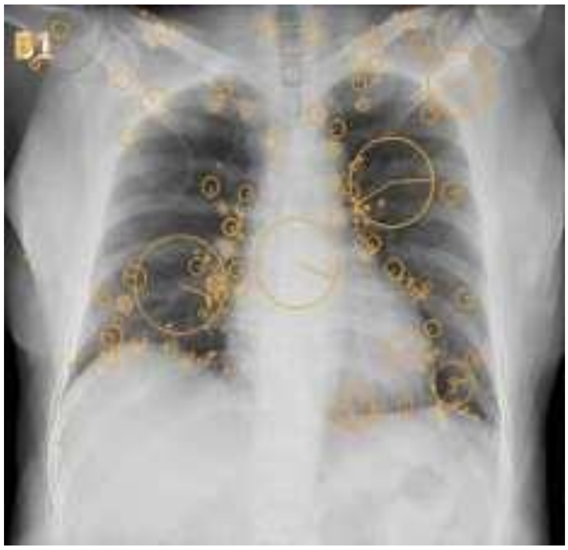

Fig -6: Detecting DoG features and extracting SIFT descriptors

3.7. Matching a logo in two images: The main points are quite clear: ORB aims to optimize and speed up operations, including the very important step of utilizing BRIEF in a rotation-aware fashion so that matching is improved, even in situations where a training image has a very different rotation to the query image. At this stage, though, perhaps you have had enough of the theory and want to sink your teeth in to some feature matching, so let's look at some code. The following script attempts to match features in a logo to the features in a photograph that contain the logo:

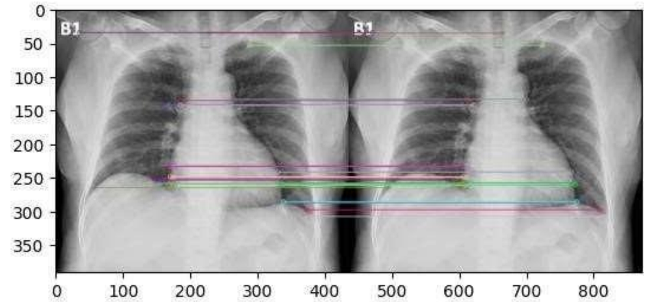

Fig -7: Foreground detection with the Grab Cut algorithm

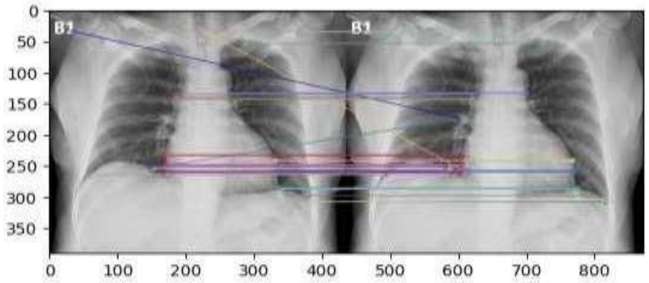

Fig -8: Filtering matches using K-Nearest Neighbors and the ratio test

3.8. Matching with FLANN: FLANN has a big toolbox, it knows how to choose the right tools for the job, and it speaks several languages. These features make the library fast and convenient. Indeed, FLANN's authors claim that it is 10 times faster than other nearest-neighbor search software for many datasets. As a standalone library, FLANN is available on GitHub at https://github.com/mariusmuja/flann/. However, we will use FLANN as part of OpenCV because OpenCV provides a handy wrapper for it. To begin our practical example of FLANN matching, let's import NumPy, OpenCV, and Matplotlib, and load two images from files.

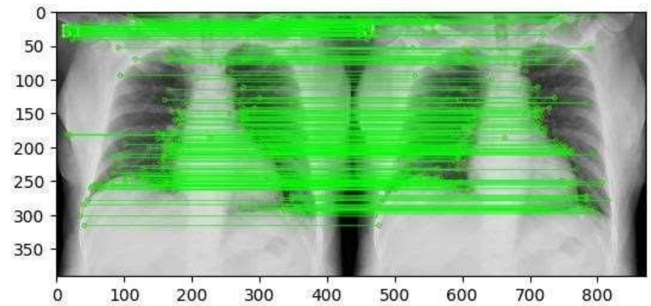

Fig -8: Filtering matches using K-Nearest Neighbors and the ratio test

In the present study, a deep-learning framework for the diagnosis of ILD was implemented for automatic identification cation of the presence of consolidation, emphysema, and fibrosis from a HRCT section. The pre-trained model was created using natural images and fine-tuned using the ILD database. Creation of a pre-trained model with natural images and subsequent transfer learning using a particular database provided acceptable results, where annotated images are scarce. In future, active learning will be explored to improve the accuracy of screening with minimal use of training images.

4. DISCUSSION

The project successfully demonstrated the potential of image processing techniques in the detection of lung diseases. The algorithm developed for this project was able to accurately identify and classify various lung diseases using chest X-ray images. The results showed that the proposed method achieved a high accuracy rate, which could lead to the development of an effective and reliable tool for early detection of lung diseases. The project highlights the importance of leveraging advancements in technology to improve the diagnosis and treatment of lung diseases, which remain a significant public health concern.

The system was able to analyze and process medical images of the lungs, identify abnormalities such as tumors and nodules, and classify them based on their type and severity. The accuracy of the system was evaluated using a dataset of medical images, and the results showed promising performance with a high level of accuracy. The developed system can potentially aid medical professionals in the early detection and diagnosis of lung diseases, leading to better patient outcomes and potentially saving lives. Further research can be conducted to improve the system's performance, extend its capabilities to other lung diseases, and integrate it with existing medical systems for broader use**.**

REFERENCES

1. Agarwala, S., Kale, M., Kumar, D., Swaroop, R., Kumar, A., Dhara, A. K., Thakur, S. B., Sadhu, A. and Nandi, D., “Deep learning for screening of interstitial lung disease patterns in high-resolution CT images,” Clin. Radiol. 75(6), 481.e1-481.e8 (2020).
2. Jacobs, C., Sánchez, C. I., Saur, S. C., Twellmann, T., De Jong, P. A. and Van Ginneken, B., “Computer-Aided detection of ground glass nodules in thoracic CT images using shape, intensity and context features,” Medical Image Computing and Computer-Assisted Intervention – MICCAI 2011. MICCAI 2011., 207–214 (2011).
3, Shin, H.-C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D. and Summers, R. M., “Deep Convolutional Neural Networks for Computer-Aided Detection: CNN architectures, dataset characteristics and transfer learning,” IEEE Trans Med Imaging 35(5), 1285–1298 (2016).
4. Van Tulder, G. and De Bruijne, M., “Learning Features for Tissue Classification with the Classification Restricted Boltzmann Machine,” [Medical Computer Vision: Algorithms for Big Data. MCV 2014.], 47–58 (2014).
5. Anthimopoulos, M., Christodoulidis, S., Ebner, L., Christe, A. and Mougiakakou, S., “Lung pattern classification for interstitial lung diseases using a deep convolutional neural network,” IEEE Trans Med Imaging 35(5), 1207–1216 (2016).
6. Gao, M., Bagci, U., Lu, L., Wu, A., Buty, M., Shin, H.-C., Roth, H., Papadakis, G. Z., Depeursinge, A., Summers, R. M., Xu, Z. and Mollura, D. J., “Holistic classification of CT attenuation patterns for interstitial lung diseases via deep convolutional neural networks,” Comput Methods Biomech Biomed Eng Imaging Vis 6(1), 1–6 (2016).
7. Choi, Y. J., Kim, J. K., Kim, N., Kim, K. W., Choi, E. K. and Cho, K.-S., “Functional MR imaging of prostate cancer,” Radiographics 27(1), 63–75 (2007).
8. Gawade, P. and Chauhan, R. P., “Detection of lung cancer using image processing techniques,” Int. J. Adv. Technol. Eng. Explor. 3(25), 217–222 (2016).
10. Hu, S., Hoffman, E. A. and Reinhardt, J. M., “Automatic lung segmentation for accurate quantitation of volumetric X-ray CT images,” IEEE Trans. Med. Imaging 20(6), 490–498 (2001).
11. Memon, N. A., Mirza, A. M. and Gilani, S. A. M., “Limitations of lung segmentation techniques,” [Proceedings of the European Computing Conference], 753–766 (2009).
12. Denison, D. M., Morgan, M. D. and Millar, A. B., “Estimation of regional gas and tissue volumes of the lung in supine man using computed tomography.,” Thorax 41(8), 620–628 (1986).
13. Hoffman, E. A. and Ritman, E. L., “Effect of body orientation on regional lung expansion in dog and sloth,” J. Appl. Physiol. 59(2), 481–491 (1985).
14. Hedlund, L. W., Anderson, R. F., Goulding, P. L., Beck, J. W., Effmann, E. L. and Putman, C. E., “Two methods for isolating the

lung area of a CT scan for density information.," Radiology 144(2), 353–357 (1982).
15. Kalender, W. A., Fichte, H., Bautz, W. and Skalej, M., "Semiautomatic evaluation procedures for quantitative CT of the lung," J. Comput. Assist. Tomogr. 15(2), 248–255 (1991).
16. Armato, S. G., Li, F., Giger, M. L., MacMahon, H., Sone, S. and Doi, K., "Lung cancer: Performance of automated lung nodule detection applied to cancers missed in a CT screening program," Radiology 225(3), 685–692 (2002).
17. El-Baz, A., Soliman, A., McClure, P., Gimel'farb, G., El-Ghar, M. A. and Falk, R., "Early assessment of malignant lung nodules based on the spatial analysis of detected lung nodules," 2012 9th IEEE International Symposium on Biomedical Imaging (ISBI) (2012).
18. Gangeh, M. J., Sørensen, L., Shaker, S. B., Kamel, M. S., De Bruijne, M. and Loog, M., "A Texton-Based approach for the classification of lung parenchyma in CT images," Med Image Comput Comput Assist Interv, 595–602 (2010).
19. Korfiatis, P. D., Karahaliou, A. N., Kazantzi, A. D., Kalogeropoulou, C. and Costaridou, L. I., "Texture-Based identification and characterization of interstitial pneumonia patterns in lung multidetector CT," IEEE Trans Inform Technol Biomed 14(3), 675–680 (2010).
20. Park, S. C., Tan, J., Wang, X., Lederman, D., Leader, J. K., Kim, S. H. and Zheng, B., "Computer-aided detection of early interstitial lung diseases using low-dose CT images," Phys Med Biol 56(4), 1139–1153 (2011).
21. Song, Y., Cai, W., Kim, J. and Feng, D. D., "A multistage discriminative model for tumor and lymph node detection in thoracic images," IEEE Transactions on Medical Imaging 31(5), 1061–1075 (2012).
22. Xu, R., Hirano, Y., Tachibana, R. and Kido, S., "Classification of diffuse lung disease patterns on High-Resolution Computed Tomography by a Bag of Words approach," Med Image Comput Assist Interv, 183–190 (2011).
23. Yao, J., Dwyer, A., Summers, R. M. and Mollura, D. J., "Computer-aided diagnosis of pulmonary infections using texture analysis and support vector machine classification," Acad Radiol 18(3), 306–314 (2011).