Machine Learning Methods and Models for Recognizing Lung Inhomogeneity from Computed Tomography Scans

G. R. Shakhmametova[1](✉), E. S. Stepanova[1], A. A. Akhmetshin[1], and R. Kh. Zulkarneev[2]

[1] Ufa University of Science and Technology, K.Marx str. 12, Ufa, Russia
shahmametova.gr@ugatu.su
[2] Bashkir State Medical University, Lenin str. 3, Ufa, Russia

Abstract. The article considers the use of deep neural networks for the recognition of lungs inhomogeneity from computed tomography (CT) scans, as well as the determination of the amount of inhomogeneity, which will help to diagnose respiratory diseases at an early stage. The recognition process is performed in several steps: lung segmentation on a CT slice, segmentation of inhomogeneous areas on the corresponding CT slice, pattern recognition, determination of the volume of inhomogeneity and generation of a report with a detailed description of the slices. The article describes the architecture of the neural network and the process of its training to identify inhomogeneous areas on a CT scan of the respiratory organs, as well as the developed algorithm for calculating the volume of inhomogeneity. To implement the recognition stage, a lightweight convolutional neural network (CNN) is used. The developed approach provides for finding areas containing inhomogeneous areas on scan slices, highlighting them in the image and determining the volume of inhomogeneity. Datasets from the public database MosMedData and NSCLC were used as data. The IOU and Dice metrics on the test data were 0.82 and 0.92, respectively. The software is implemented in the Python 3.6 programming language using the Django 2.2 framework and is cross-platform. For machine learning algorithms, PyTorch 1.9.0 packages were used. An analysis of the effectiveness of the results obtained showed a recognition accuracy of 91% according to the Dice metric. #CSOC1120.

Keywords: Machine Learning · Neural Networks · Convolutional Neural Networks · Lung Inhomogeneity · CT Scans Recognition

1 Introduction

Today, information technology and machine learning methods are actively used in medicine, including the recognition of functional diagnostic images, which include CT scans. According to a study by Lunit, an artificial intelligence system was able to correctly recognize more than 97% of images when detecting malignant tumors [1]. In medical diagnostics, a second opinion is also important, both for the physician and the

R. Silhavy and P. Silhavy (Eds.): CSOC 2023, LNNS 722, pp. 339–350, 2023.
https://doi.org/10.1007/978-3-031-35311-6_35

patient. For example, lung cancer is the leading cause of death among cancers worldwide [2]. Lung cancer accounts for more than 90% of all pulmonary neoplasms and 28% of all deaths resulting from human tumor disease. Its early identification in the form of inhomogeneity has an increasing importance in the diagnosis of respiratory diseases. When the disease is detected early on lung CT, the likelihood of a favorable outcome is much higher. But identifying the disease at an early stage is difficult enough for a novice or inexperienced specialist. To help the specialist, reduce the workload on a radiologist, increase the speed of CT scan analysis, and improve the quality of analysis, it is proposed to apply machine learning and deep neural network methods to recognize areas of inhomogeneity in the lungs on CT scans.

This article proposes an algorithm for recognizing patterns of inhomogeneous areas of the lungs and determining the amount of inhomogeneity of each part of the lung. The algorithm is based on a deep learning mathematical model. A combination of the MosMedData dataset [3] and NSCLC [4], containing computer tomograms of lungs with markup by radiologists, was used to train the model.

2 Related Works

Research in the field of lung X-ray and CT scan recognition is now actively pursued in many countries. As a rule, researchers set a narrow task, for example, recognition of coronavirus infection, segmentation of nodules, recognition of fibrous tissue, etc. In light of the severe epidemiological situation, work on specific software has accelerated. Most software is designed for hospitals or private clinics and targets a specific list of pathologies.

Researchers from the Department of Radiology at Wuhan Huangpi Hospital have developed the COVNet framework for COVID-19 detection from CT scans [5], which is based on the ResNet50 architecture. CovNet extracts a set of features from each slice of the CT scan and then combines them to produce a final feature map. It is then fed to a fully connected network with SoftMax activation to determine the probability of COVID-19.

GE Healthcare, together with the University of Rouen and the Department of Nuclear Medicine of the Henri Becquerel Center, conducted a study that aimed to use the information contained in several linked tasks to improve both segmentation and COVID-19 classification [6]. An architecture was developed for three tasks: COVID-19 classification, COVID-19 segmentation, and image reconstruction to improve the representation of the extracted data. For all three tasks, an encoder was created that takes a CT scan as input. For the segmentation task, a decoder was created where the encoder output is sent. Another decoder was used for the image reconstruction task. For the classification task, the encoder output is sent to the fully connected network. The output of the binary classifier answers the presence or absence of COVID-19.

Researchers from Taif University in Saudi Arabia together with scientists from Cairo University in Egypt developed a lightweight model to detect lung nodules on CT scan for mobile devices [7]. They used 4 convolutional layers as the architecture. Each layer consists of consecutive convolutional blocks, a 1*1 connective convolutional block and max pooling operation, and nonlinear ReLU activation functions after each block. The

4 convolution blocks are followed by a global average pooling GAP operation and two fully connected layers with SoftMax activation function at the output for classification. The network determines whether the CT scan slice belongs to one of three classes: a benign mass, a malignant mass, or an image without masses.

CAD4TB is tuberculosis (TB) recognition software [8]. The result of the software is a texture heat map on the original X-ray image and the probability of tuberculosis. The processing time for one image is 30 seconds, but the main drawback of the soft-ware is its ability to work only with an X-ray image.

Imsight CT Analysis System is software for MRI machines [9]. The software is capable of detecting 19 types of thoracic abnormalities. When screening for coronavirus infection the accuracy is 99%, the processing time per image is 10-20 seconds. This software product is only available for use in hospitals in China.

At the European Congress of Radiology Annual Meeting (ECR-2017), a study was presented in which artificial intelligence was used for primary disease detection from X-rays [10]. The developers were La Fe Institute for Medical Research (Valencia, Spain) and Quibim. A computer system for respiratory disease detection (Computed-aided detection, CAD) based on convolutional neural networks was developed. The algorithm is able to recognize diseases from the chest radiograph with high accuracy, taking into account the sensitivity and specificity of the organism.

Analysis of studies in the field of recognition of human lung CT scans showed a lack of readily available solutions in the field of recognition with the necessary accuracy of lung inhomogeneity.

3 Problem Statement

A computed tomography (CT) scan consists of a certain number of images - slices. The number of slices depends on the characteristics of the CT scanner and can range from 30 to 300. The evaluation of a CT scan comes down to a segmentation task followed by a binary classification of image sections.

On a CT scan, inhomogeneity can be expressed in one of four patterns: consolidation, atelectasis, ground-glass, nodule [11]. Images of the patterns for recognition are shown in Fig. 1.

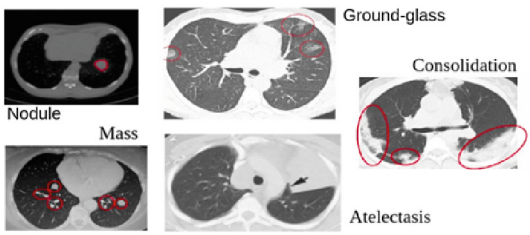

Fig. 1. Images of patterns for recognizing lung inhomogeneity

For each slice of the CT scan, it is necessary to determine the presence and location of lung tissue compaction and determine the percentage of inhomogeneity in each lung. This process is carried out in three stages:

1. Segmentation of the image of the lungs on the slice.
2. Recognition of inhomogeneous areas.
3. Determination of the percentage of lung inhomogeneity.

4 Proposed Solution

To recognize the four patterns of inhomogeneity, it is proposed to use a neural network trained on a dataset consisting of CT scans with marked inhomogeneous areas.

It was decided to use a combination of two datasets, MosMedData and NSCLC, as datasets. In the first dataset, 50 marked CT scans were used, and in the second dataset, 31 were used. The total size of the dataset was 3256 slices.

A series of experiments were conducted with different deep neural network architectures and approaches. The architectures experimented with were UNet [12], SegNet [13], XNet [14], AttUnet [15], and a modification of the network with a Dilated Convolution approach [16]. All networks were trained from scratch; for the UNet architecture, in addition to learning from scratch, a transfer learning approach was also used.

During the experiments, regularization in the form of Dropout layers was added to the network, the loss function was modified, as well as the image preprocessing algorithm (image size and normalization) and augmentation. Various optimizers were also used. As a result of the experiments, the values shown in Table 1 were obtained.

Table 1. Experimental results

Architecture	IOU	Dice
SegNet	62	75
XNet	68	77
AttUNet	64	74
UNet	63	74
Dilated Conv UNet	65	77

During the training of UNet architecture using transfer learning the following metrics values were obtained: IOU - 40, Dice - 50.

During the analysis of the experimental results, the XNet network was chosen because it showed good convergence on a small number of epochs. This architecture was also used to solve the problem of segmentation of bones and soft tissues on X-ray images [15] and it showed a value of F1 metric equal to 0.92, and a value of accuracy metric equal to 0.92. Its advantages are that it can be trained on a small dataset and that the model is lightweight, which reduces the recognition time.

5 Realization

The model training pipeline consists of 6 stages: image loading, preprocessing, augmentation, network setup, training, and evaluation of metrics on a test set (see Fig. 2).

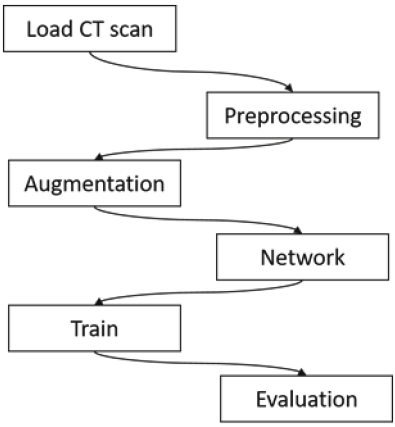

Fig. 2. The model training pipeline.

5.1 Loading a CT Scan

The SimpleITK library is used to load a CT scan. After loading the CT scan we get a tensor with a certain number of slices, the number of slices depends on the raw data.

5.2 Preprocessing

The CT scan must be preprocessed before the slices are fed. Each CT slice is preprocessed independently of the others. Pre-processing consists of two steps:

(1) resizing the image to 160*160 pixels;
(2) image normalization.

5.3 Augmentation

Augmentation is used to increase the dataset and stabilize the network, which not only increases the size of the dataset, but also trains the model to correctly process slices on noisy data. The following transformations were used as augmentations: shift, rotate, and random crop. For image augmentations the library Albumentations [17] was used, which provides many options for changing the image.

5.4 Neural Network

The proposed model includes 14 convolutional blocks with ReLU activation and batch normalization, 5 layers of MaxPooling2d and 5 layers of UpSample. The final layer is a convolution block with batch normalization and SoftMax activation function. Also, Dropout layers were added for regularization [18]. The architecture is presented in detail in Fig. 3.

Input	input	(None, 160, 160, 1)	Conv2d + BatchNorm2d + ReLU	input	(None, 80, 80, 256)
	output	(None, 160, 160, 1)		output	(None, 80, 80, 128)
Conv2d + BatchNorm2d + ReLU	input	(None, 160, 160, 64)	MaxPolling2d	input	(None, 80, 80, 128)
	output	(None, 160, 160, 64)		output	(None, 40, 40, 128)
MaxPolling2d	input	(None, 160, 160, 64)	Conv2d + BatchNorm2d + ReLU	input	(None, 40, 40, 128)
	output	(None, 80, 80, 64)		output	(None, 40, 40, 256)
Conv2d + BatchNorm2d + ReLU	input	(None, 80, 80, 64)	MaxPolling2d	input	(None, 40, 40, 256)
	output	(None, 80, 80, 128)		output	(None, 20, 20, 256)
MaxPolling2d	input	(None, 80, 80, 128)	Conv2d + BatchNorm2d + ReLU	input	(None, 20, 20, 256)
	output	(None, 40, 40, 128)		output	(None, 20, 20, 512)
Conv2d + BatchNorm2d + ReLU	input	(None, 40, 40, 128)	Conv2d + BatchNorm2d + ReLU	input	(None, 20, 20, 512)
	output	(None, 40, 40, 256)		output	(None, 20, 20, 512)
MaxPolling2d	input	(None, 40, 40, 256)	Upsample	input	(None, 20, 20, 512)
	output	(None, 20, 20, 256)		output	(None, 40, 40, 512)
Conv2d + BatchNorm2d + ReLU	input	(None, 20, 20, 256)	Conv2d + BatchNorm2d + ReLU	input	(None, 40, 40, 512)
	output	(None, 20, 20, 512)		output	(None, 40, 40, 256)
Conv2d + BatchNorm2d + ReLU	input	(None, 20, 20, 512)	Upsample	input	(None, 40, 40, 512)
	output	(None, 20, 20, 512)		output	(None, 80, 80, 512)
Upsample	input	(None, 20, 20, 512)	Conv2d + BatchNorm2d + ReLU	input	(None, 80, 80, 512)
	output	(None, 40, 40, 512)		output	(None, 80, 80, 128)
Conv2d + BatchNorm2d + ReLU	input	(None, 40, 40, 256)	Upsample	input	(None, 80, 80, 256)
	output	(None, 40, 40, 256)		output	(None, 160, 160, 256)
Upsample	input	(None, 40, 40, 512)	Conv2d + BatchNorm2d + ReLU	input	(None, 160, 160, 512)
	output	(None, 80, 80, 512)		output	(None, 160, 160, 64)
Conv2d + BatchNorm2d + ReLU	input	(None, 80, 80, 512)	Conv2d + BatchNorm2d + SoftMax	input	(None, 160, 160, 64)
	output	(None, 80, 80, 128)		output	(None, 160, 160, 1)

Fig. 3. The final XNet architecture

5.5 Model Training

To determine the percentage of inhomogeneity and generate a report, it is necessary to segment the lungs; for this purpose, the Lungmask library [19] was used, which is based on the UNet network. Lung segmentation is performed in parallel with the operation of the main network for segmentation of inhomogeneous areas. ReLU was used as the activation function, and Softmax was used on the last layer. Adam with a learning rate of 0.0001 was chosen as the network optimizer. Binary Cross Entropy was selected as the loss function.

The training, validation, and test sets were formed to train the model, the ratio of which was 80:10:10, respectively. The training time was 200 epochs. The PyTorch framework was used for training. The output of the network determines the probability of belonging to a class. Class 0 - no inhomogeneous section detected, class 1 - inhomogeneous section on the slice detected.

5.6 Model Evaluation

As a response, the model gives a mask for each CT slice of the image. After training, it is necessary to determine segmentation metrics. Dice and IoU are used as metrics. On the test set, the value of the Dice and IoU metrics was 91% and 84%, respectively.

5.7 Report Generation

The final step is to apply a mask to the image that was on the slice before segmentation. This is done by obtaining the contours of inhomogeneous areas on the mask using the OpenCV library, after which the highlighted contours are superimposed on the CT slice by multiplying the binary mask with the image.

Figure 4 shows the result of the algorithm of recognition and highlight of inhomogeneous areas on a slice of CT scan.

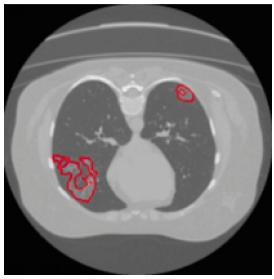

Fig. 4. Highlight of inhomogeneous areas on a CT slice

After the model response is received and the mask is applied to the slice of the CT scan, a report is generated consisting of the following items:

1. Calculated percentage of inhomogeneity in each lung. The percentage of inhomogeneity is calculated as the ratio of the number of pixels in each lung that are counted from the mask obtained in the XNet response, to the number of pixels in each lung that are counted from the mask obtained using the Lungmask library (CT scan slice with segmented lungs) (Eq. 1).

$$Inhomogeneity = \frac{\sum_i^N \sum_j^H \sum_k^W M1}{\sum_i^N \sum_j^H \sum_k^W M2} \tag{1}$$

where M1 – binary mask obtained using XNet;
M2 – binary mask of segmented lungs, obtained using Lungmask library;
N – the number of slices in CT scan;
H, W – the dimensionality of the slice that is fed into the model.

To calculate the percentage of inhomogeneity in each lung, using the Lungmask library, we obtain a mask with 3 classes (right lung, left lung and background).

According to Eq. 1, we calculate the percentage of inhomogeneity for each lung (Eq. 2).

$$Inhomogeneity_{left} = \frac{\sum_i^N \sum_j^H \sum_k^W M1}{\sum_i^N \sum_j^H \sum_k^W M2} \quad (2)$$

where M1 – binary mask obtained with XNet. Only the part of the mask that belongs to the left lung is taken into account;

M2 – left lung mask, obtained with Lungmask library.

An example of the report is shown in Fig. 5.

Chest CT scan analysis

Date: 05.06.2021

Patient data

A patient: ---
Date of birth: ---
Low inhomogeneity

Analysis results

Complaints: ---

The percentage of inhomogeneity of the lungs: 18.662%
The percentage of inhomogeneity of the left lung: 13.75%
The percentage of inhomogeneity of the right lung: 4.913%

With the volume of inhomogeneity close to zero, it should be assumed that the lungs are healthy.

Attention!
The results of the analysis should not be taken as the only correct result. In order to take further action, a MANDATORY doctor's consultation is required! In case of a disappointing result, in no case should you give in to panic or give up. The program may be wrong, so you need to contact specialists for further advice, and preferably several.

Fig. 5. Lung inhomogeneity percentage report

2. List of slices of a CT scan with highlighted areas of inhomogeneity (see Fig. 6).

In Fig. 6 slices of the CT scan contain white blackouts, which the model detected and highlighted. This type of inhomogeneity is represented by the frosted glass class.

3. Explanation of why these areas were highlighted (see Fig. 7).

The recognition system also explains the highlighted areas on the slices, explaining why they were highlighted and what exactly needs to be paid attention to.

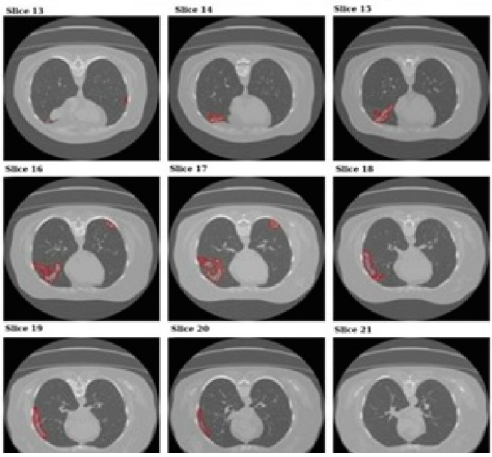

Fig. 6. The result of dividing the image into areas

Explanations

There are 39 slices in total.

Areas of inhomogeneity were not found on the slice 1-13
Areas of inhomogeneity of the Frosted glass type are defined on sections 14-20 are highlighted with a red outline.
Areas of inhomogeneity were not found on the slice 21-25
Areas of inhomogeneity of the Frosted glass type are determined on sections 26-28 are highlighted with a red outline.
Areas of inhomogeneity were not found on the slice 29-39

Fig. 7. Example of the report

6 Results

CT scans from the publicly available MosMedData database were used to test the model. The set contains a total of 1110 CT scans, among which are:

(1) healthy lungs (254 CT scans);
(2) lungs with lesions <= 25% (684 CT scans);
(3) lungs with 25 to 50% lesions (125 CT scans);
(4) lungs with 50 to 75% lesions (45 CT scans);
(5) lungs with lesions over 75% (2 CT scans).

Marked CT scans are provided for scans with 25% lesion volume, for the rest only the volume of the lesion is known. The marked CT scans were included in the dataset, which was further divided into training, validation, and test sets.

Intersection over Union (IOU) and Dice metrics were used as an assessment of the accuracy of the model for segmenting inhomogeneous sections. On the test set, the model showed 84% and 91% on the IOU and Dice metrics, respectively.

For unmarked CT scans, testing was performed, that predicted lesion grade:

(1) healthy lungs (<=2%);
(2) degree of lesion 2 to 25%;
(3) degree of lesion 25 to 50%;
(4) degree of lesion 50 to 75%;
(5) degree of lesion >75%.

The results of confusion matrix are presented in Table 2.

Table 2. Confusion Matrix

Reality/Prediction	Positive	Negative
Positive	148 (TP)	85 (FN)
Negative	47 (FP)	207 (TN)

Using the resulting confusion matrix, Precision, Recall, and F-score metrics were calculated (Equations 3-5):

$$Precision = \frac{TP}{TP + FP} = \frac{148}{148 + 47} = 0.75 \tag{3}$$

$$Recall = \frac{TP}{TP + FN} = \frac{148}{148 + 85} = 0.64 \tag{4}$$

$$F = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * 0.75 * 0.64}{0.75 + 0.64} = 0.68 \tag{5}$$

7 Discussions

At the moment, the developed algorithm for recognizing lung inhomogeneity and the corresponding software is implemented as a telemedicine service, the client part of which is written using the React and Antd libraries [20], and the server part is written in Python 3.6 using the Django framework. The server has several modules, each of which is engaged in its own task – loading, preprocessing, recognition, post-processing, analyzing the results and generating a report with subsequent sending to the user (see Fig. 8).

It should be noted that the developed model is not always able to recognize large inhomogeneous areas, which are clearly visible to the human eye. This is due to the peculiarities of the dataset on which the network was trained. The original dataset contained CT scans with mostly small inhomogeneous areas. By supplementing the dataset, this problem can be solved.

Thus, now the model is not able to fully replace a specialist radiologist, but it can significantly help the doctor in decision support when detecting small inhomogeneous

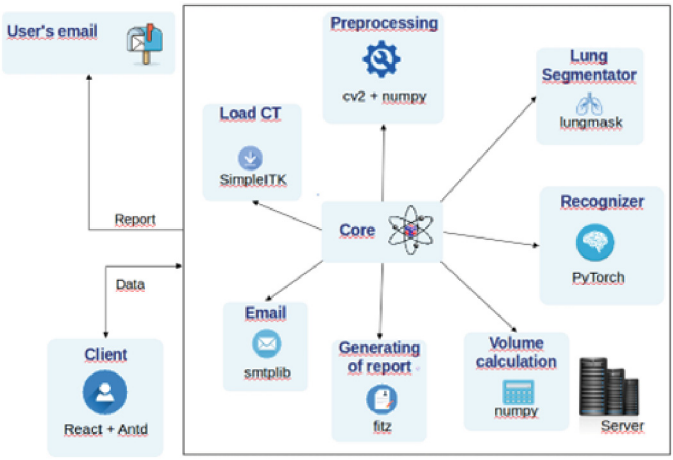

Fig. 8. The architecture of the lung inhomogeneity recognition service

areas in the lungs, in which then pathologies, including malignant ones at early stages, can be detected with a high degree of probability.

The developed algorithms can also be implemented as a separate module of the clinical decision support system for the prevention and treatment of bronchopulmonary diseases.

8 Conclusion

The result of the study is the development of a model for the detection of inhomogeneous areas on CT lung scans. The implementation involves 3 stages: segmentation of lungs on a slice of CT lung scan for further calculation of the volume of inhomogeneity, segmentation of inhomogeneous areas on a CT slice and the generation of a report. A convolutional neural network based on XNet architecture was developed and trained for segmentation of inhomogeneous areas, which showed the best results in the experiments to select the network architecture. The test results showed 84% and 91% on IOU and Dice metrics, respectively.

The model is used in a developed telemedicine service that can assist the diagnostician in the analysis of CT lung scans. The service can be used for decision support and it saves time in image recognition and reduces the influence of human factors such as fatigue, inattention or lack of experience.

Further research in this area is planned in the area of expanding the recognizable patterns of inhomogeneity of the lungs and improving recognition accuracy. It is also planned to implement the obtained solutions in the form of an embedded module of a decision support system for the prevention and treatment of bronchopulmonary diseases.

Acknowledgment. The reported study was funded by RSF, project number 22-19-00471.

References

1. Nvidia. https://blogs.nvidia.com/blog/2022/07/21/lunit-healthcare-ai-ipo/. Accessed 01 Mar 2023
2. World Health Organization. https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death. Accessed 01 Mar 2023
3. Morozov, S. P., Andreychenko, A. E., et al.: MosMedData: Chest CT scans with COVID-19 related findings dataset, arXiv preprint arXiv:2005.06465 (2020). https://doi.org/10.48550/arXiv.2005.06465
4. Bakr, S., et al.: A radiogenomic dataset of non-small cell lung cancer. Sci. Data **5**, 180202 (2018). https://doi.org/10.1038/sdata.2018.202
5. Li, L., et al.: Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT. Radiology **296**(2), 200905 (2020). https://doi.org/10.1148/radiol.2020200905
6. Amyar, A., Modzelewski, R., Ruan, S.: Multi-task deep learning based CT imaging analysis for COVID-19: Classification and segmentation. MedRxiv preprint (2020). https://doi.org/10.1101/2020.04.16.20064709
7. Mehedi, M., Ghulam, M., et al.: Light deep model for pulmonary nodule detection from CT scan images for mobile devices. Wirel. Commun. Mobile Comput. **2020**, 8893494 (2020). https://doi.org/10.1155/2020/8893494
8. Delft Imaging Project. https://www.delft.care/cad4tb/. Accessed 01 Mar 2023
9. KR. https://36kr.com/p/1725371727873. Accessed 01 Mar 2023
10. AuntMinnie. https://www.auntminnie.com/index.aspx?sec=ser&sub=def&pag=dis&ItemID=116825. Accessed 24 Sept 2022
11. Radiology Assistant. https://radiologyassistant.nl/chest/chest-x-ray/lung-disease. Accessed 01 Mar 2023
12. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
13. Badrinarayanan, V., Kendall, A., Cipolla, R.: SegNet: A deep convolutional encoder-decoder architecture for image segmentation. In: IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2481–2495 (2017). https://doi.org/10.1109/TPAMI.2016.2644615. PMID: 28060704.
14. Bullock, J., Cuesta-Lazaro, C., Quera-Bofarull, A.: XNet: A convolutional neural network (CNN) implementation for medical X-ray image segmentation suitable for small datasets. In: Medical Imaging 2019: Biomedical Applications in Molecular Structural and Functional Imaging, vol. 10953, p. 109531Z (2019)
15. Oktay, O., Schlemper, J., et al.: Attention U-Net: Learning where to look for the pancreas. arXiv preprint arXiv:1804.03999 (2018)
16. Wang, Z., Ji, S.: Smoothed dilated convolutions for improved dense prediction. Data Mining Knowl. Discov. **35**(4), 1470–1496 (2021). https://doi.org/10.1007/s10618-021-00765-5
17. Albumentations. https://albumentations.ai/. Accessed 01 Mar 2023
18. Srivastava, N., et al.: Dropout: A simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
19. Github. Library for segmentation lung. https://github.com/JoHof/lungmask. Accessed 01 Mar 2023
20. Ant Design. https://ant.design/. Accessed 01 Mar 2023