Preparing CT imaging datasets for deep learning in lung nodule analysis

Wang, Jingxuan; Sourlos, Nikos; Zheng, Sunyi; van der Velden, Nils; Pelgrim, Gert Jan; Vliegenthart, Rozemarijn; van Ooijen, Peter

Published in:
Heliyon

DOI:
10.1016/j.heliyon.2023.e17104

Heliyon 9 (2023) e17104

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon

Preparing CT imaging datasets for deep learning in lung nodule analysis: Insights from four well-known datasets

Jingxuan Wang [a,*], Nikos Sourlos [a], Sunyi Zheng [c], Nils van der Velden [a], Gert Jan Pelgrim [a], Rozemarijn Vliegenthart [a,d], Peter van Ooijen [b,d,**]

[a] Department of Radiology, University of Groningen, University Medical Center of Groningen, 9713GZ, Groningen, the Netherlands
[b] Department of Radiation Oncology, University of Groningen, University Medical Center of Groningen, 9713GZ, Groningen, the Netherlands
[c] School of Engineering, Westlake University, Xihu District, 310030, Hangzhou, China
[d] Data Science Center in Health (DASH), University of Groningen, University Medical Center of Groningen, 9713GZ, Groningen, the Netherlands

ARTICLE INFO

Keywords:
Deep learning
Lung nodule dataset
Data access and download
Data annotation
Data preprocessing

ABSTRACT

Background: Deep learning is an important means to realize the automatic detection, segmentation, and classification of pulmonary nodules in computed tomography (CT) images. An entire CT scan cannot directly be used by deep learning models due to image size, image format, image dimensionality, and other factors. Between the acquisition of the CT scan and feeding the data into the deep learning model, there are several steps including data use permission, data access and download, data annotation, and data preprocessing. This paper aims to recommend a complete and detailed guide for researchers who want to engage in interdisciplinary lung nodule research of CT images and Artificial Intelligence (AI) engineering.

Methods: The data preparation pipeline used the following four popular large-scale datasets: LIDC-IDRI (Lung Image Database Consortium image collection), LUNA16 (Lung Nodule Analysis 2016), NLST (National Lung Screening Trial) and NELSON (The Dutch-Belgian Randomized Lung Cancer Screening Trial). The dataset preparation is presented in chronological order.

Findings: The different data preparation steps before deep learning were identified. These include both more generic steps and steps dedicated to lung nodule research. For each of these steps, the required process, necessity, and example code or tools for actual implementation are provided.

Discussion and conclusion: Depending on the specific research question, researchers should be aware of the various preparation steps required and carefully select datasets, data annotation methods, and image preprocessing methods. Moreover, it is vital to acknowledge that each auxiliary tool or code has its specific scope of use and limitations. This paper proposes a standardized data preparation process while clearly demonstrating the principles and sequence of different steps. A data preparation pipeline can be quickly realized by following these proposed steps and implementing the suggested example codes and tools.

* Corresponding author. Department of Radiology, University of Groningen, University Medical Center of Groningen, 9713GZ, Groningen, the Netherlands.

** Corresponding author. Department of Radiation Oncology, University of Groningen, University Medical Center of Groningen, 9713GZ, Groningen, the Netherlands.

E-mail addresses: j.wang02@umcg.nl (J. Wang), p.m.a.van.ooijen@umcg.nl (P. van Ooijen).

https://doi.org/10.1016/j.heliyon.2023.e17104
Received 25 May 2023; Received in revised form 6 June 2023; Accepted 7 June 2023
Available online 16 June 2023

1. Introduction

Lung cancer is the leading cause of cancer mortality and one of the most malignant tumors that threaten the health and life of people. In the latest American cancer statistics, lung cancer ranks second among cancers in terms of estimated new cases and mortality in both men and women [1]. The most common early manifestations of lung cancer are lung nodules, which can be found during chest CT examinations either as incidental findings or as part of a screening setup.

In recent years, many deep learning-based solutions have aimed for lung nodule evaluation in imaging exams [2]. Research involving AI has so far mainly focused on solving specific tasks in radiological evaluation, such as pulmonary nodule detection [3] and pulmonary nodule classification [4]. There are only a few reviews and studies describing how to prepare medical image datasets for deep learning. It has been pointed out that AI engineers should not forget to include domain experts to allow them to gain insights on how to obtain carefully curated image data [5]. To guide this process, Willemink et al. [6] introduced general principles for medical imaging data handling from ethical approval to data annotation. However, these are only the steps performed before preprocessing the data to be fed into AI algorithm. Adam et al. [7] constructed the largest brain CT imaging dataset for developing machine learning algorithms for detection and characterization of intracranial hemorrhage. According to the above studies, data preparation is the main factor that influences the performance of any deep learning method. There is no published work describing the whole preparation process of a CT image-based lung nodule dataset for any deep learning task.

Both public and private data collections have their specific scope. The well-known LIDC-IDRI public dataset [8] can be used for lung nodule detection and classification. Another public dataset LUNA16 [9] is designed for lung parenchyma segmentation, lung nodule detection, and false positive reduction in the detection of pulmonary nodules. In clinical practice, observing the growth of lung nodules is an important indicator of lung cancer; therefore, public dataset NLST [10] and private dataset NELSON studies [11] are suitable for lung nodule follow-up evaluation because of the presence of follow-up scans.

Accurate annotation of lung nodules in CT scans is a crucial step for the development and evaluation of deep learning algorithms. Some publicly available datasets such as LIDC-IDRI and LUNA16 provide annotations, while obtaining valid annotations for other datasets may be a challenging and time-consuming process. In this regard, it is important to establish rigorous annotation protocols and utilize both experts and automated tools to ensure high-quality annotations.

Only in this way can subsequent data preprocessing be carried out on the CT scan. A CT scan is normally stored in DICOM format which contains pixel/voxel data and header. The pixel/voxel data refers to an image composed of integer values. A DICOM header provides information related to the image, such as slice location, image orientation, window level/width, etc [12]. However, despite the use of the DICOM standard, some challenges such as discontinuous slices, inappropriate Field of View (FOV) settings, and inappropriate default window level/width settings still need to be solved via data preprocessing.

This work aims to introduce a complete preparation process from acquiring CT scans towards developing a dataset that can be used for deep learning-based lung nodule research. Multiple complex and necessary processes should be performed in cooperation with professionals with different roles and responsibilities. This study can be useful for engineers and data managers without a medical background to obtain the basic knowledge of CT scans and preparation steps. Additionally, medical professionals (e.g., radiologists and pulmonologists) can obtain a basic understanding of some image processing techniques and steps. Moreover, each data preparation step has included example codes or tools, which help novice researchers quickly implement their lung cancer research in AI.

2. Methods

2.1. Large-scale datasets

Nowadays, the requirements for datasets in deep learning research are becoming increasingly stringent. Researchers should select data based on various aspects such as dataset size, data storage structure, data annotation, etc. Four well-known large-scale datasets that can be used for various lung nodule tasks are described in Table 1, which comprise three public datasets (LIDC-IDRI, LUNA16, and NLST) and a private dataset (NELSON).

The LIDC-IDRI includes 1018 chest spiral CT scans from 1010 patients and annotations collected during a two-phase annotation process using four experienced radiologists [8]. The CT images are available in DICOM format. The annotations are stored as Extensive Markup Language (XML) files recording two-phase lung nodule annotation results. In the first phase, the radiologists independently detected and labelled the type (nodule≥3 mm, nodule < 3 mm, non-nodule≥3 mm), axial location of approximate three-dimensional

Table 1
Data availability of four datasets.

	LIDC-IDRI	LUNA16	NLST	NELSON
Total number	1018 scans/1010 patients	888 scans	54,000 participants	7557 participants
Format of CT images	DICOM	MHD and Raw	DICOM	DICOM
Slice thickness of CT images	0.5–5 mm	≤2.5 mm	1.0–3.2 mm	1.0 mm
Demographic description	–	–	+	+
Follow-up	–	–	+	+
Pixel based annotation of lung nodule	+	+	–	–

+Available; -: Not available.

center-of-mass, and characteristics (subtlety, internal structure, spiculation, lobulation, sphericity, solidity, margin, and likelihood of malignancy) of the lung nodules. In the second phase, each radiologist independently reviewed their own and the other three radiologists' annotations and gave their final review. This two-phase annotation ensures the accuracy and authority of the annotation results.

The LUNA16 is a challenge competition dataset derived from the LIDC-IDRI dataset, to develop lung nodule detection and false positive reduction algorithms [9]. To standardize the evaluation of models and algorithms, the LUNA16 initiative included new rules to select the data and reconstructed CT images, nodule annotations, and auxiliary information. The LUNA16 dataset opted not to use the DICOM format because the DICOM standard is primarily designed for clinical use. Therefore, the dataset provides 888 scans in. mhd (MetaIO Header) and. raw file formats, which are more flexible and easier to work with for research purposes. Additionally, masks for lung parenchyma segmentation, the pixel coordinates of the nodules (coordX, coordY, coordZ), and the diameter of each nodule are supplemented.

The NLST was designed to determine whether low-dose spiral CT screening for lung cancer reduces lung cancer mortality in a high-risk population relative to chest X-ray screening [10]. The NLST provides approximately 54,000 participants' baseline and follow-up scans. In addition, this study includes information such as patient demographics, smoking history, and scan acquisition dates. However, NLST does not include the coordinates of nodules on CT scans.

The NELSON trial investigated whether low-dose multi-detector CT screening for lung cancer in high-risk patients reduces lung cancer mortality [11]. NELSON includes over 7557 patients' baseline and follow-up scans, lung nodule location, and associated annotation labelled by radiologists. The CT image and annotation are presented in DICOM and secondary capture images (images with burned in annotations captured from the screen saved DICOM format) respectively.

2.2. Data preparation

Each lung nodule dataset preparation basically follows the following four processes (Fig. 1).

2.2.1. Data use permission

Data use permission is required before accessing and downloading. For example, for the NELSON dataset, a data access board evaluates the proposed project based on its feasibility, safety and adherence to the data use regulations based on the legislation. In addition, ethical approval may be required from the medical ethics committee. The LIDC-IDRI and LUNA16 do not require data use permission. However, their public licenses determine the restrictions on the use of the data. Although NLST is also a public dataset, researchers still need to obtain permission from the data committee.

2.2.2. Data access and download

Data access and download procedures follow a similar pattern, but individual implementation of different datasets may vary. Based on the most common implementation steps (Fig. 2): 1) The user uses a client (web-accessible visual interface or command lines) to send a data request to the server through an application program interface (API); 2) the server extracts the images from the database; 3) the image data is returned from the database to the server, 4) the server returns the data to the client so that the user can access the data.

As a user, access and download functions are used more than other functions (create, upload, modify and delete data). LUNA16 provides an open-access website for downloading data [13]. LIDC-IDRI data can be queried and downloaded from its host website directly [14]. The NLST dataset can be accessed and downloaded through a personal user account [15]. For LIDC-IDRI and NLST, users

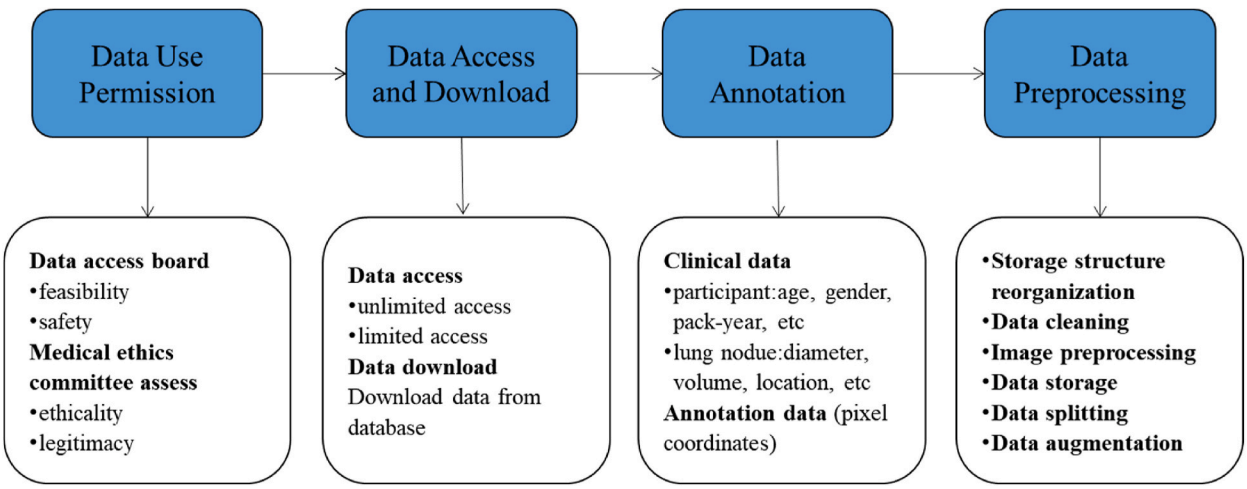

Fig. 1. Data preparation process.

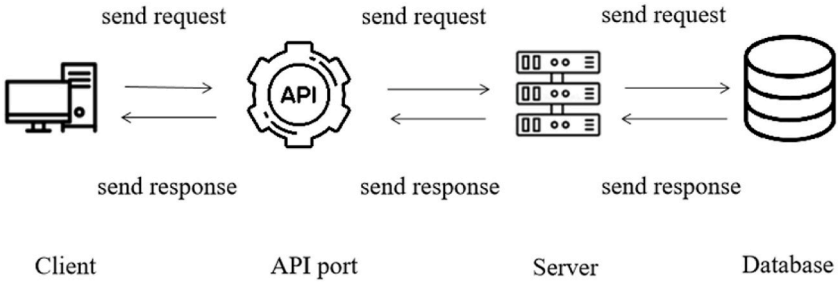

Fig. 2. The process of accessing and downloading data from the server.

can request the data through an image download application (e.g., NBIA Data Retriever). For the NELSON study, DICOM data are saved in the open-source imaging informatics platform XNAT (eXtensible Neuroimaging Archive Toolkit) [16]. Based on the hierarchical data store structure of XNAT [17], an automatic image download program has been designed and published on GitHub [18].

2.2.3. Data annotation

Data annotation includes the recording of findings either in metadata or in the image itself. For imaging data, data annotation is the process of labelling the region of interest (ROI) in the CT image. In general, the image annotation methods of nodules are divided into three categories: 1) marking the approximate centroid of the lung nodule, 2) drawing a rectangle bounding box around the lung nodule, or 3) drawing the contour of the lung nodule (also known as pixel-based annotation). Different annotation methods serve different deep learning tasks. If the lung nodule detection task utilizes an object detection algorithm to detect lung nodules, a rectangle bounding box around the nodule is needed. In addition to directly obtaining the bounding box through method 2), the coordinates of lung nodule obtained by method 1) and 3) can be used to calculate the bounding box. Method 1) has low operational complexity, because radiologists only need to mark one pixel of a nodule instead of multiple pixels. Additionally, it is less challenging for engineers to calculate and extract a rectangular bounding box by the coordinates of the lung nodule center. Only LIDC-IDRI and LUNA16 both provide pixel-based annotations (X-Y-Z coordinates) (method 3).

When pixel-level annotations are not available, readers (e.g., radiologists) have to generate this information with manual annotations. The number of readers depends on various factors such as the desired level of agreement between annotations, the complexity of the annotations, and the availability of resources such as time and funding. Generally, at least 2 readers should annotate a dataset to ensure the accuracy and reliability of the annotations, and a third reader to decide if there are discrepancies. It is also important to consider the experience and expertise of the readers.

One approach to generate manual pixel-level annotations is to use specialized software designed for this purpose, such as ITK-SNAP [19] or 3D Slicer [20]. These freely available tools allow radiologists to manually segment and label ROIs within CT scans. A more extensive overview of open-access tools for image annotation can be found in Ref. [21] which also provides the capability of each tool. In the appendix, the use of 3D Slicer to perform image annotation of lung nodules is demonstrated.

2.2.4. Data preprocessing

2.2.4.1. Storage structure reorganization. When CT scans are stored without proper storage structure organization, the resulting dataset will not be clean and concise thus hindering researchers to quickly query the scans based on the folder and file naming without having to access the DICOM header. For example, the public dataset mentioned in this paper appears to be well-organized and easily readable, owing to the fact that it was cleaned and pre-processed to a certain extent prior to its publication. Our recommendation is to name scans/images with uniquely identifiable numbers such as patient ID, seriesUID (the unique identifier for the CT series), and instance number or a combination of these identifiers, similar to how LIDC-IDRI or LUNA16 named their dataset. For data with follow-up, the scan date can be added to differentiate scans in different periods, such as in the NLST. An example code for storage structure reorganization can be found in Ref. [22].

2.2.4.2. Data cleaning. Data cleaning is the process of fixing or removing errors, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset [23]. In lung nodule research, data cleaning involves processing both image and non-image data to ensure high-quality data.

A CT scan typically includes hundreds of DICOM slices, but there may be instances where one or more slices are missing from the scan. Missing slices can be identified using the DICOM header by checking if slice number or position are consecutive. In addition, the entire thorax contours may be beyond the FOV of the CT image, due to e.g., radiographer error. In some cases, a lung nodule may be located near the edge of the lung. If the entire lung contour is incomplete, the nodule may be partially or completely "clipped" or cut off from view. All in all, scans with incomplete lung contours or with discontinuous slices should be carefully examined and potentially discarded to ensure the validity of the data.

For non-image data, there is numerous descriptive information of patient and lung nodules in metadata. Some problems should be

dealt with, such as removing duplicate patient id, handling missing values of lung nodule diameter, and handling outliers of pack years. We recommend a Python tutorial (including code) for data cleaning [24].

2.2.4.3. Image preprocessing. Preprocessing of the images is essential to meet specific requirements for input data, such as resizing and normalization, which have been shown to improve the accuracy of the model [25]. Fig. 3 shows the different preprocessing steps that are usually performed in different areas of lung nodule research. And Table 2 provides links to examples of code that can be used to perform the steps.

2.2.4.3.1. Lung window/level setting. During lung nodule screening, radiologists adjust window/level settings to distinguish different tissues in the thorax, including lung parenchyma, bones, and blood vessels. Window/level settings consist of Hounsfield Units (HU), window width (WW), and window level/center (WL). HU is a quantitative unit that measures radiodensity in CT images calibrated against water (0 HU) and air (−1024 HU). WW defines the range of HU values, and WL is the center HU value of WW.

The pixel data of the DICOM images is not expressed in HU values, therefore, before applying HU-based manipulation such as the lung window setting, a linear transformation from pixel values to HU values should be performed with two values (tag: Rescale Intercept and Rescale Slope) in the DICOM header.

Default window/level values (tag: Window Width and Window Center) can also be retrieved from the DICOM header, but they may not be the optimal values for a specific purpose. For example, the common lung WW value ranges from 1500 HU to 1600 HU, and the common lung WL value ranges from −700 HU to −500 HU, while values of 350 HU for WW and 50 HU for WL are suitable for mediastinal/soft tissue structures [26,27]. Data scientists could be mistakenly use a default DICOM header setting of WW at 350HU and WL at 50HU to construct the images to train the deep learning model for nodule detection while they should have used a lung setting. Therefore, default values should not be used without consideration and consultation of an expert/radiologist.

2.2.4.3.2. Resampling. Resampling is performed to achieve consistent voxel sizes across different CT scans. When acquiring CT images from different scanners or using different CT protocols on the same scanner, variations in pixel size and slice thickness can occur [28], leading to increased variability in the features of two-dimensional (2D) images or three-dimensional (3D) volumes. Additionally, thicker slices have less image detail and can negatively impact the 3D reconstruction of anatomical structures [29]. For example, in lung nodule recognition tasks (e.g., false positive reduction, classification), it is common to use 3D images to improve accuracy by increasing spatial information [30].

Resampling can be achieved using various image processing software or libraries. One common method is to use interpolation techniques, such as linear interpolation or cubic interpolation [31], to transform the pixel data of the original image into a new grid with a fixed spacing between voxels. The spacing can be determined based on the desired voxel size for the specific task or the input requirement of the deep learning model. The resampling process can also involve adjusting the image orientation and alignment to ensure the images are in a consistent anatomical position.

2.2.4.3.3. Lung parenchyma segmentation. Lung parenchyma segmentation is performed to separate the chest wall, mediastinum, bronchi, and heart from the lungs in CT images, which can reduce the interference of structures outside the lung parenchyma on lung nodule detection. LUNA16 provides lung parenchyma masks for segmentation. In general, datasets do not contain segmented lung parenchyma; therefore one of the following two methods can be used: 1) manual or semi-automatic software, e.g., 3D slicer; 2) automatic method [32,33], such as adaptive thresholding [34], region growing [35] and Unet [36].

2.2.4.3.4. Lung nodule extraction. For lung nodule classification or false positive reduction, ROIs are small patches containing lung

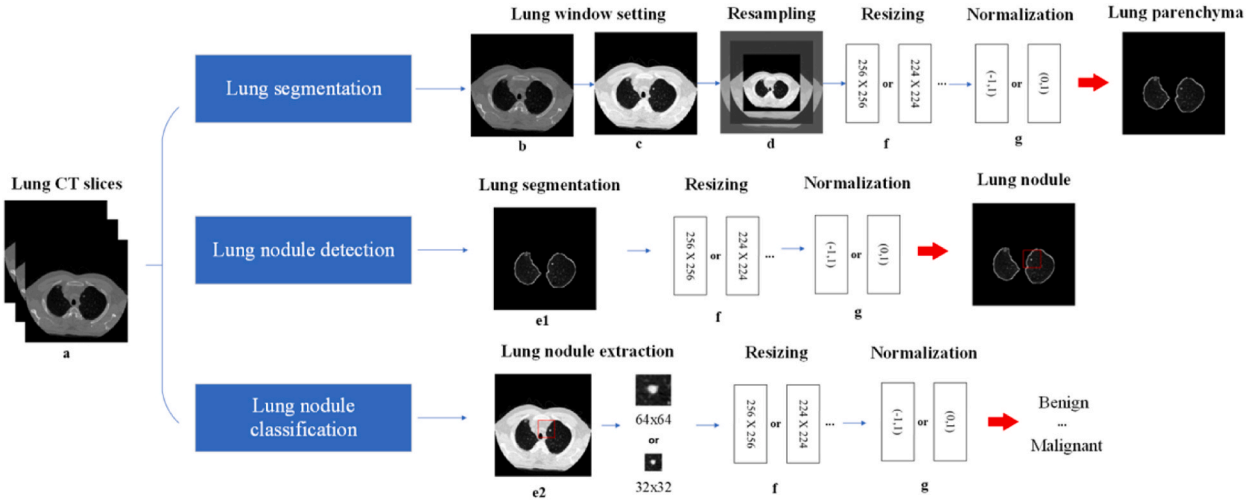

Fig. 3. Image preprocessing steps may be involved in different tasks. (a: original lung; b: lung HU setting; c: lung window/level setting; d: lung resampling; e1: lung parenchyma segmentation; e2: lung nodule extraction (32 × 32 and 64 × 64 are the example size); f: resizing lung (256 × 256 and 224 × 224 are the example size); g: normalizing).

Table 2
List of example codes in image preprocessing.

	Code	Core function/program name	Explanation
Lung window/level setting	https://vincentblog.xyz/posts/medical-images-in-python-computed-tomography	transform_to_hu ( ) window_image ( )	Transforms the pixel values to the HU Changes CT image into lung window
Resampling	https://github.com/Wxy-24/Dicom_preprocessing/blob/master/dcm_preprocessing.py	Resample ( )	Changes voxels to uniform size
Lung parenchyma segmentation	https://github.com/paulmtree/Lung-Segmentation-Project	DICOMProcessing.py	Filters out the lung and trachea and bronchi
		RegionGrowerLarge.py	Extracts the trachea and bronchi
		BronchialGrower.py	Uses the smaller region growing algorithm to extract the bronchioles from the lungs
	https://github.com/JoHof/lungmask	U-net	Default segmentation model
Lung nodule extraction	https://github.com/wxlearncoding/Lung-nodule-extraction	extract_lung_nodule_from_center ( )	Extracts lung nodule using approximate centroid
		extract_lung_nodule_by_boundingbox ( )	Extracts lung nodule based on the bounding box
	https://pylidc.github.io	pylidc	Extracts lung nodule using pixel coordinates of the contour
Resizing	https://www.kaggle.com/code/crawford/resize-and-save-images-as-numpy-arrays-128x128	proc_images ( )	Resizes images according to target size (width, height)
Normalization	https://www.kaggle.com/code/akh64bit/full-preprocessing-tutorial/notebook#Normalization	Normalization	Normalizes grey scale images to a range of 0–1

nodule rather than the whole lung. The choice of lung nodule extraction method depends on different annotation methods. If the lung nodule is annotated by the approximate centroid and bounding box, the nodule and its background can be extracted together in that bounding box. In the case of the pixel-wise annotation of the nodule volume or boundary, nodules can be extracted without surrounding tissue or within a bounding box that includes surrounding tissue. To preserve the complete characteristics of the nodule, the largest (outermost) contour should be used to extract the nodule.

2.2.4.3.5. Resizing. Deep learning models were (pre-)trained on images of certain dimensions which dictate a pre-defined size of the input image. Therefore, resizing is necessary when working with deep learning models that require input images to comply with this pre-defined size. Depending on the task, the resizing can be applied to different ROI. The input images for a lung nodule detection model can be either the original CT image or lung parenchyma. Similarly, for a lung nodule classification model, the input images are lung nodule patches categorized into different classes. Common image interpolation methods, such as nearest neighbour, bilinear interpolation, or bicubic interpolation, can be used to estimate new pixel values during resizing [31]. It is important to note that resizing can affect the quality of the image, and choosing an appropriate interpolation method is crucial to preserve important features and avoid artifacts.

2.2.4.3.6. Normalization. Normalization is the process of rescaling pixel or voxel values in medical images to a fixed range, such as 0 to 1 or -1 to 1, which is typically required before feeding the images into a deep learning model. This is because deep learning models may not perform optimally if the input pixel values have a wide range of values, and normalization can help to ensure that the model receives input data that is consistently scaled to a range. Moreover, normalization can improve the speed of gradient descent convergence during training [37].

2.2.4.4. Data storage. After data preprocessing, data storage and data splitting need to be considered depending on the task to be performed. Table 3 shows the links of the example codes of data storage and splitting.

For saving and viewing purposes, lung nodules may be saved in JPEG/JPG and PNG format. JPEG/JPG often uses a lossy compression algorithm, whereas PNG uses a lossless compression algorithm that helps to retain the full image information [38]. Because of this difference in compression methodology, PNG preserves the original image pixel values more closely than JPEG/JPG. For lung nodule classification, different categories of nodules are usually saved in different folders named using the nodule category. To use a lung nodule detection algorithm, researchers need to prepare data storage in one of the two popular formats (Pascal VOC [39] and Microsoft COCO [40]). These formats provide a standardized way to organize and store images and annotations, making it easier

Table 3
List of example codes in data storage.

	Code	Core function/program name	Explanation
Pascal VOC format	https://www.kaggle.com/code/dschettler8845/vinbigdata-convert-annotations-to-pascal-voc-xml/notebook	create_xml_file ( )	Creates Pascal VOC dataset for object detection
Microsoft COCO format	https://www.kaggle.com/code/alejopaullier/how-to-create-a-coco-dataset	create_coco_format_json ( )	Creates Microsoft COCO dataset for object detection
	https://github.com/Tony.607/voc2coco	voc2coco.py	Converts Pascal VOC to Microsoft COCO format

Table 4
List of example codes in data splitting.

	Code	Core function name	Explanation
Data splitting	https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html https://github.com/dnfwlxo11/cocosplit_train_test_valid	sklearn.model_selection.train_test_split cocosplit_train_test_valid.py	Splits arrays or matrices into random train and test subsets Data splitting for data storage in Microsoft COCO format

to train and evaluate models.

2.2.4.5. Data splitting. The data after preprocessing should be divided into training, validation, and test set. The training set is used to train and fine-tune the model parameters. The validation set is used to monitor the state of the model and the convergence and to determine the best hyperparameters. The test set is used to evaluate the generalization capability of the model.

Data splitting is generally achieved through methods such as hold-out split, k-fold split, stratified k-fold split, etc [41]. The choice of the splitting method is, among other factors, based on the overall size of the dataset. Hold-out split is a simple and widely used method that can be employed when the data is large enough (i.e., millions of samples). In this method, the splitting ratio is generally recommended to be 8:2 or 7:3 for the training and test sets, respectively [42]. However, for dataset with limited samples, k-fold cross-validation can be used to maximize the usage of available data for training, validation, and testing. In this method, the entire dataset is divided into k equally sized folds, and the model is trained and evaluated k times on different folds, with each fold serving as the validation set once and the remaining folds serving as the training set. When dealing with imbalanced datasets, the stratified split should be employed to ensure that each split has a balanced distribution of each class. For example, if the ratio of class 1 to class 2 in the original dataset is 5:1, then the same split ratio (5:1) should be maintained in the training set, validation set, and test set, respectively.

Table 4 shows the links of the example codes of data splitting. In practice, the splitting of training and validation sets is usually done during the model training, either through techniques such as data generators or by manually specifying the split during training.

2.2.4.6. Data augmentation. Data augmentation is a technique that artificially expands the training dataset by generating more equivalent data from limited data [43]. Augmentation is widely used to solve a series of problems caused by the shortage of medical images. Chlap [44] summarized three augmentation techniques: 1) basic technique (geometric transformations, cropping, noise injection, etc); 2) deformable technique (randomized displacement field, spline interpolation, deformable image registration, etc; 3) deep learning-based technique (generative adversarial networks-based augmentation methods and other DL-based augmentation methods). Data augmentation can either be performed prior to training the model, resulting in a larger dataset with a mix of original and augmented samples, or it can be invoked from computer vision tools or libraries after data reading and before model training. In runtime, PyTorch's transformer provides image augmentation methods such as flipping, cropping, and adding noise.

3. Discussion

In this paper, we summarized the necessary data preparation process from data access to data augmentation for CT image-based lung nodule data in deep learning research.

3.1. Dataset application

Different datasets have different characteristics and applications, and it is important to choose the appropriate dataset for a specific research question. For instance, the LIDC-IDRI and LUNA16 datasets are suitable for beginners in lung nodule research as they provide pixel coordinates of lung nodules. However, they may not be ideal for specific subgroup analyses or follow-up studies due to the lack of demographic information and follow-up scans. In contrast, the NLST and NELSON datasets contain abundant data samples, demographic information, and follow-up scans, but lack pixel-level annotations. Researchers can still utilize these datasets if expert annotators are available to generate annotations when needed.

Public datasets are often preferred for deep learning applications as they are relatively clean and contain fewer errors and redundancies. One potential benefit of using public datasets is that they can reduce the obstacles caused by the lack of data annotation, another is to provide a platform for comparing the performance of deep learning models using the same dataset but with different implementations. However, public datasets may still have limitations. In contrast, private datasets may contain more data and clinical information, including comprehensive metadata, follow-up, and long-term outcomes. Although private datasets may also have issues like duplicate data and missing data, it is worth solving these problems. Overall, researchers should carefully consider the advantages and disadvantages of different datasets and choose the one that best suits their research needs.

3.2. Data annotation

In the early stages of lung nodule annotation, natural image annotation tools such as LabelMe were used [45]. However, LabelMe was not specifically designed for medical imaging as it only supported PNG or JPG file formats. This meant that format conversion of DICOM was necessary before lung nodules could be annotated, leading to increased operational complexity and time consumption. However, annotation tools that allow direct annotation in DICOM format are also available. Examples of such tools include 3D slicer and ITK-SNAP, which have a lung nodule labelling function. While the 3D slicer allows the pixel coordinates of annotations to be saved, this is not possible in ITK-SNAP. Chen et al. [46] also developed a web-based semi-automatic lung nodule annotation system called DeepLNAnno, which has shown good performance on real-world datasets. However, there is still room for improvement in this system, as window level/width cannot be adjusted, polygon annotation and diameter measurement of lung nodule are not available, and it is

not publicly available yet. These developments in annotation tools have made the process of lung nodule annotation more efficient and effective, but there is still a need for further improvement in the accuracy and efficiency of the process.

3.3. Data preprocessing

Data preprocessing is an essential step before building deep learning models. Although various data preprocessing methods are available through public frameworks or Python packages, researchers should ensure that these methods are suitable for their specific needs. For instance, open-source frameworks such as MONAI [47] and TorchIO [48] provide rich and robust data preprocessing methods for deep learning work. Given that the MONAI framework is currently under active development, researchers must ensure that they keep their version updated. TorchIO has only a few transformation methods developed for CT images, which limits its utility for specific applications.

To improve the generalizability of the deep learning model, data splitting and data augmentation methods are generally utilized in practice. Data splitting can be flexible for large datasets, but model training and testing with fewer data can have high variance in parameter estimation and performance metrics. In this case, cross-validation is a better choice than the hold-out method [49]. Data augmentation can overcome insufficient training data, avoid sample imbalance, avoid overfitting, and improve model robustness. However, it also inevitably introduces noise due to the difference between the generated data and the real data. Hence, researchers should use data augmentation as needed and with caution.

4. Conclusion

This paper takes four datasets as examples and introduces the preparation process of how to convert CT scans into pulmonary nodule data available for deep learning. Depending on the specific research question, not every step of data preparation will be used in practice. Researchers need to carefully select dataset, data annotation methods, and data preprocessing methods. To enable researchers to quickly understand each step of data preparation, we provide some example codes/tools for researchers who want to engage in pulmonary nodule research.

Author contribution statement

Jingxuan Wang: Conceived and designed the experiments; Performed the experiments; Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data; Wrote the paper.

Nikos Sourlos; Sunyi Zheng; Nils van der Velden; Gert Jan Pelgrim: Analyzed and interpreted the data; Wrote the paper.

Rozemarijn Vliegenthart; Peter van Ooijen; Conceived and designed the experiments; Wrote the paper.

Data availability statement

Data included in article/supp. Material/referenced in article.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

Jingxuan Wang is grateful for the PhD financial support from the China Scholarship Council (CSC file No. 202006540020) and the University of Groningen. This funding source had no role in the design, data collection, management, analysis, interpretation, preparation, review, approval of the manuscript, and decision to submit the manuscript for publication.

Appendix A

Based on the study [21], we selected 3D slicer (https://www.slicer.org/) as an annotation tool for lung nodule coordinates. We chose CT images from a lung cancer screening study as an example. In the 3D slicer, the default geospatial location is on the RAS (R: right, A: anterior, S: superior) coordinate system. After importing the scan in the [DICOM] module, we should adjust the lung window setting in the [Volumes] module or click on the "Adjust window/level" button on the toolbar. In the [Markups] module, there are several annotation methods. For lung nodule annotation, three methods are shown in the following description.

1) Marking the approximate centroid of the lung nodule

Use the *Point List* tool or ROI tool to mark the approximate centroid of the lung nodule (Fig. 1).

2) Drawing a rectangle bounding box around the lung nodule

Use the *ROI* tool and draw a bounding box for the lung nodule (Fig. 2).

3) Drawing the contour of the lung nodule

Use the Curve tool and draw a circle-like shape around the contour of the lung nodule on three axes (Fig. 3).

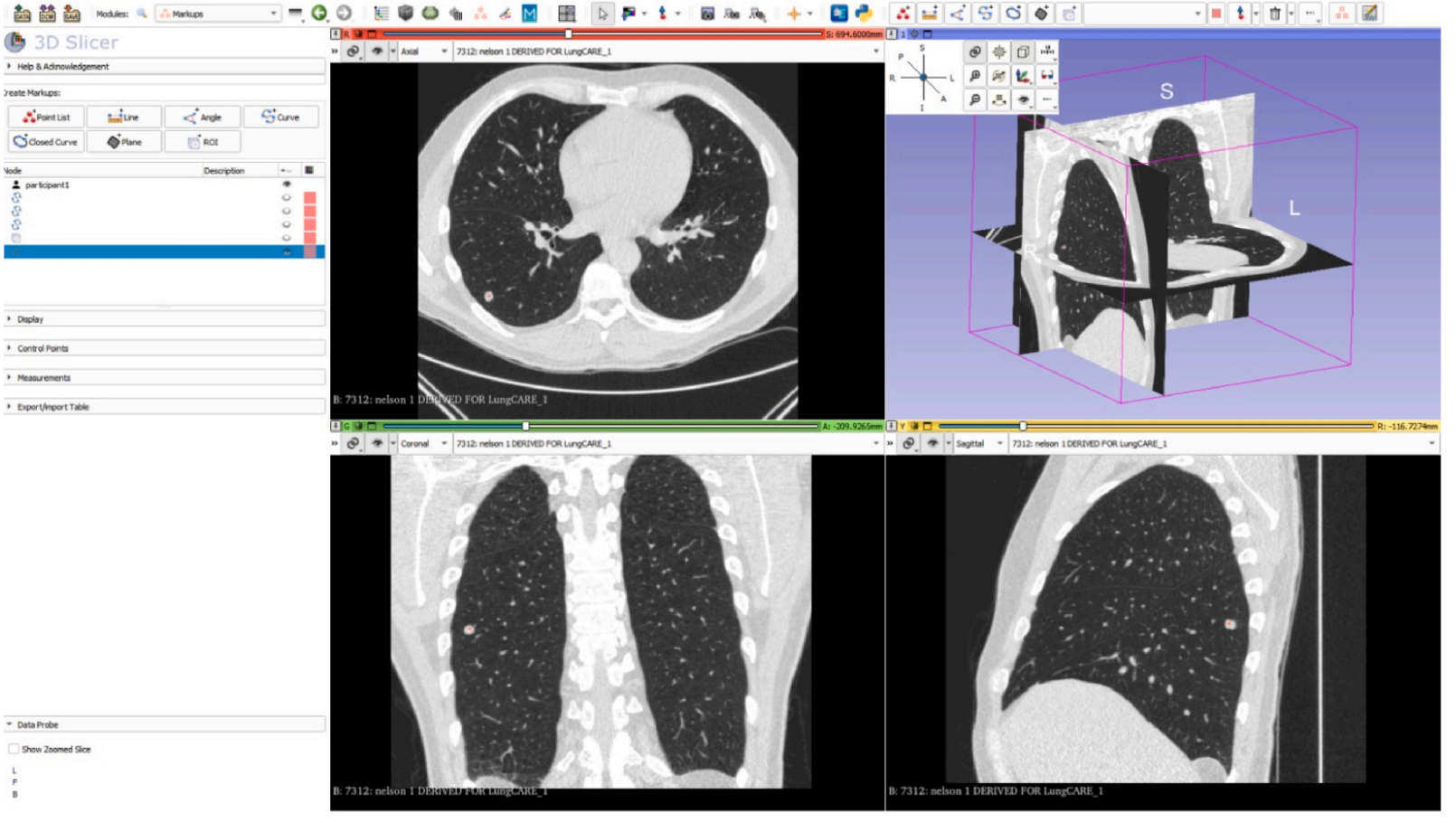

Fig.1. The approximate centroid of the lung nodule on three axes

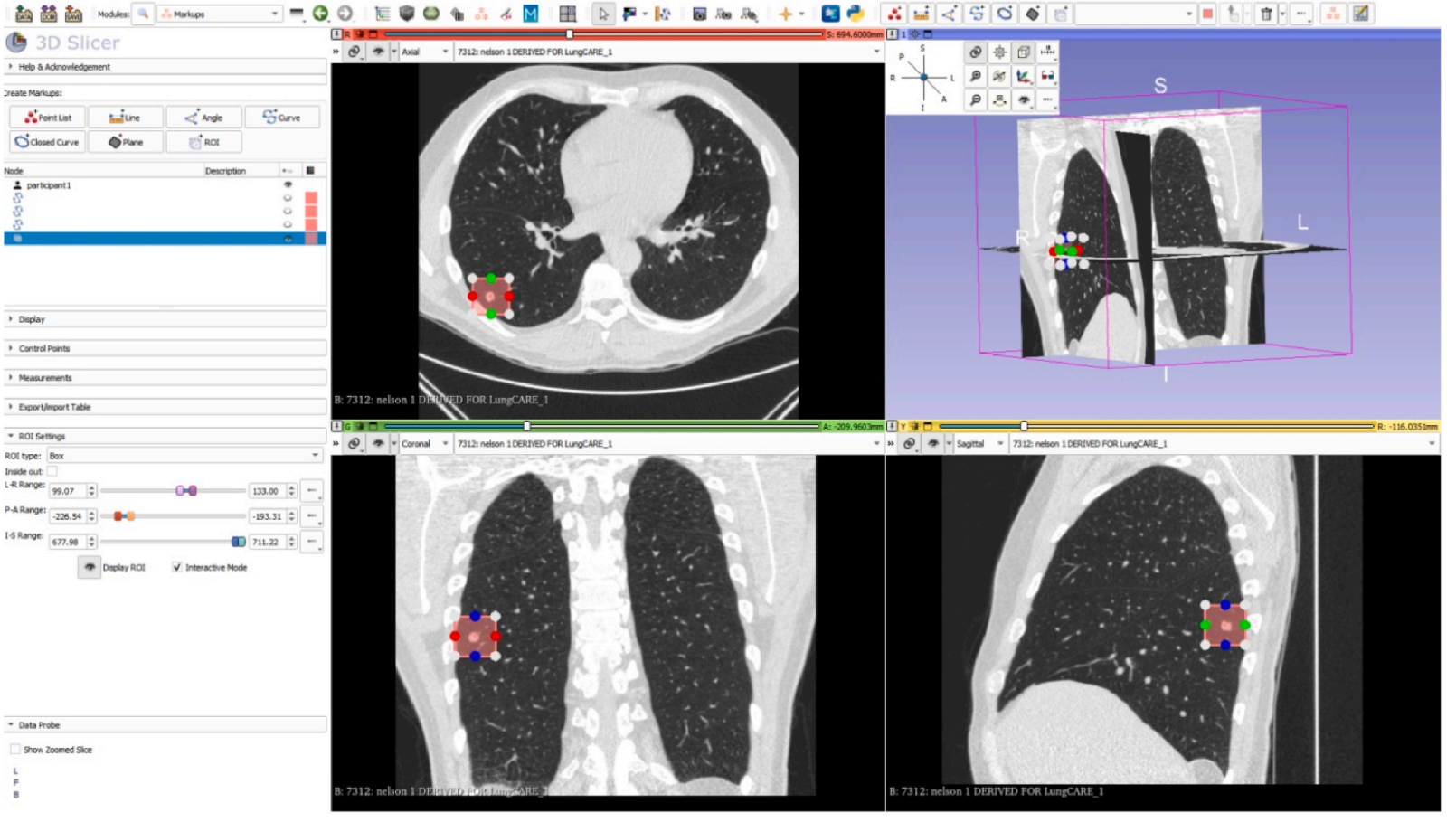

Fig.2. The rectangle bounding box around the lung nodule on three axes

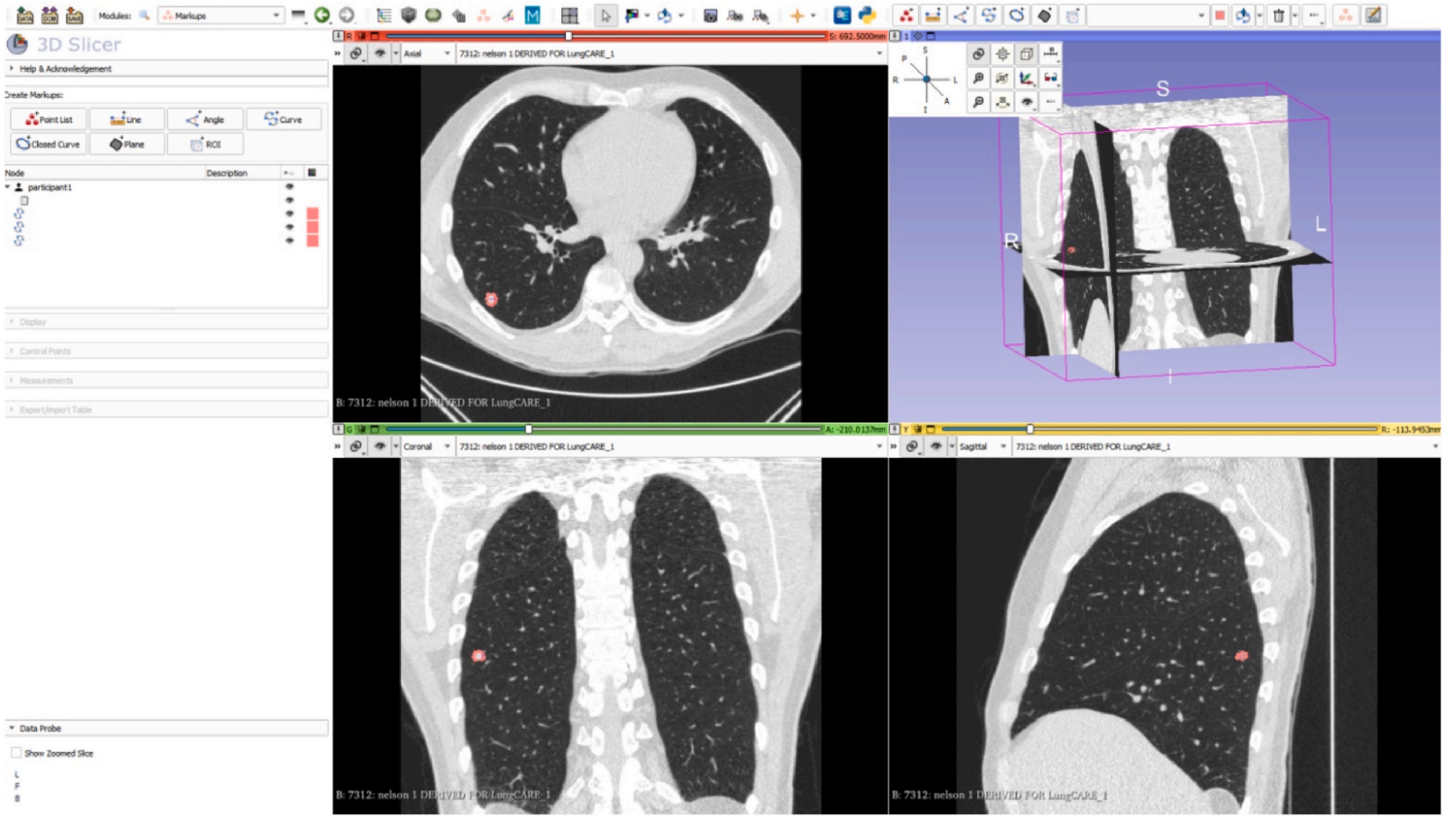

Fig.3. The contour of the lung nodule

In order to name the nodule, the annotator used the following tags: screenee id, nodule id, and slice id. The screenee id and nodule id uniquely identify participants and nodules. The slice id refers to the unique id of the axial plane where the nodule is located.

References

[1] R.L. Siegel, K.D. Miller, H.E. Fuchs, et al., Cancer statistics, 2022, CA A Cancer J. Clin. 72 (1) (2022) 7–33, https://doi.org/10.3322/caac.21708.
[2] Y. Gu, J. Chi, J. Liu, et al., A survey of computer-aided diagnosis of lung nodules from CT scans using deep learning, Comput. Biol. Med. 137 (2021), 104806, https://doi.org/10.1016/j.compbiomed.2021.104806.
[3] S. Zheng, L.J. Cornelissen, X. Cui, et al., Deep convolutional neural networks for multiplanar lung nodule detection: improvement in small nodule identification, Med. Phys. 48 (2) (2021) 733–744, https://doi.org/10.1002/mp.14648.
[4] K.L. Hua, C.H. Hsu, S.C. Hidayati, et al., Computer-aided classification of lung nodules on computed tomography images via deep learning technique, OncoTargets Ther. 8 (2015) 2015–2022, https://doi.org/10.2147/OTT.S80733. Published 2015 Aug 4.
[5] H.R. Tizhoosh, J. Fratesi, COVID-19, AI enthusiasts, and toy datasets: radiology without radiologists, Eur. Radiol. 31 (5) (2021) 3553–3554, https://doi.org/10.1007/s00330-020-07453-w.
[6] M.J. Willemink, W.A. Koszek, C. Hardell, et al., Preparing medical imaging data for machine learning, Radiology 295 (1) (2020) 4–15, https://doi.org/10.1148/radiol.2020192224.
[7] A.E. Flanders, L.M. Prevedello, G. Shih, et al., Construction of a machine learning dataset through collaboration: the RSNA 2019 brain CT hemorrhage challenge [published correction appears in Radiol Artif Intell. 2020 Jul 29;2(4): e209002], Radiol. Artif. Intell. 2 (3) (2020), e190211, https://doi.org/10.1148/ryai.2020190211. Published 2020 Apr 29.
[8] S.G. Armato 3rd, G. McLennan, L. Bidaut, et al., The lung image database Consortium (LIDC) and image database resource initiative (IDRI): a completed reference database of lung nodules on CT scans, Med. Phys. 38 (2) (2011) 915–931, https://doi.org/10.1118/1.3528204.
[9] A.A.A. Setio, A. Traverso, T. de Bel, et al., Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the LUNA16 challenge, Med. Image Anal. 42 (2017) 1–13, https://doi.org/10.1016/j.media.2017.06.015.
[10] National Lung Screening Trial Research Team, D.R. Aberle, A.M. Adams, et al., Reduced lung-cancer mortality with low-dose computed tomographic screening, N. Engl. J. Med. 365 (5) (2011) 395–409, https://doi.org/10.1056/NEJMoa1102873.
[11] Y. Zhao, X. Xie, H.J. de Koning, et al., NELSON lung cancer screening study, Spec No A, Cancer Imag. 11 (1A) (2011) S79–S84, https://doi.org/10.1102/1470-7330.2011.9020. Published 2011 Oct 3.
[12] DICOM Standards Committee, Data dictionary, DICOM (June 05, 2023). https://dicom.nema.org/medical/dicom/current/output/html/part06.html.
[13] C. Jacobs, A.A.A. Setio, A. Traverso, et al., Lung Nodule Analysis 2016. Grand Challenge, Update January, 2018. June 05, 2023, https://luna16.grand-challenge.org/Download/.
[14] B. Vendt, B. Camp, Data from the lung image database Consortium (lidc) and image database resource initiative (idri): a completed reference database of lung nodules on CT scans (LIDC-IDRI), Cancer Imag. Arch. (November 01, 2022). Updated, https://wiki.cancerimagingarchive.net/pages/viewpage.action?pageId=1966254. (Accessed 5 June 2023).
[15] Lung Screening Study group (LSS), the American College of Radiology Imaging Network (ACRIN), Begin a new NLST project, Nat. Cancer Inst. Cancer Data Access Syst. (May 12, 2022). Update, https://cdas.cancer.gov/nlst/. (Accessed 5 June 2023).
[16] D.S. Marcus, T.R. Olsen, M. Ramaratnam, et al., The Extensible Neuroimaging Archive Toolkit: an informatics platform for managing, exploring, and sharing neuroimaging data, Neuroinformatics 5 (1) (2007) 11–34, https://doi.org/10.1385/ni:5:1:11.
[17] The NRG Lab at Washington University, Understanding the XNAT data model, XNAT (June 05, 2023). https://wiki.xnat.org/documentation/how-to-use-xnat/understanding-the-xnat-data-model.
[18] J. Wang, XNAT-for-downloading-DICOM-data, GitHub (June 05, 2023). https://github.com/wxlearncoding/XNAT-for-downloading-DICOM-data.
[19] P.A. Yushkevich, J. Piven, H.C. Hazlett, et al., User-guided 3D active contour segmentation of anatomical structures: significantly improved efficiency and reliability, Neuroimage 31 (3) (2006) 1116–1128, https://doi.org/10.1016/j.neuroimage.2006.01.015.
[20] A. Fedorov, R. Beichel, J. Kalpathy-Cramer, et al., 3D slicer as an image computing platform for the quantitative imaging network, Magn. Reson. Imag. 30 (9) (2012 Nov 1) 1323–1341.
[21] O. Diaz, K. Kushibar, R. Osuala, et al., Data preparation for artificial intelligence in medical imaging: a comprehensive guide to open-access platforms and tools, Phys. Med. 83 (2021) 25–37, https://doi.org/10.1016/j.ejmp.2021.02.007.
[22] J. Wang, Storage-structure-reorganization, GitHub (June 05, 2023). https://github.com/wxlearncoding/Storage-structure-reorganization.
[23] J. Van den Broeck, S.A. Cunningham, R. Eeckels, et al., Data cleaning: detecting, diagnosing, and editing data abnormalities, PLoS Med. 2 (10) (2005) e267, https://doi.org/10.1371/journal.pmed.0020267.
[24] Packt, Python-data-cleaning-cookbook, GitHub (June 05, 2023). https://github.com/PacktPublishing/Python-Data-Cleaning-Cookbook.
[25] G. Ranganathan, A study to find facts behind preprocessing on deep learning algorithms, J. Innovat. Image Process 3 (1) (2021) 66–74, https://doi.org/10.36548/jiip.2021.1.006.
[26] E.J. Stern, M.S. Frank, J.D. Godwin, Chest computed tomography display preferences. Survey of thoracic radiologists, Invest. Radiol. 30 (9) (1995) 517–521, https://doi.org/10.1097/00004424-199509000-00002.
[27] H. Takahashi, M. Kiyoshima, T. Kaburagi, et al., Influence of radiologic expertise in detecting lung tumors on chest radiographs, Diagn. Interv. Imaging 100 (2) (2019) 95–107, https://doi.org/10.1016/j.diii.2018.08.015.
[28] M. Shafiq-Ul-Hassan, K. Latifi, G. Zhang, et al., Voxel size and gray level normalization of CT radiomic features in lung cancer, Sci. Rep. 8 (1) (2018), 10545, https://doi.org/10.1038/s41598-018-28895-9. Published 2018 Jul 12.
[29] J.M. Ford, S.J. Decker, Computed tomography slice thickness and its effects on three-dimensional reconstruction of anatomical structures, J. Forensic. Radiol. Imaging 4 (2016) 43–46, https://doi.org/10.1016/j.jofri.2015.10.004.
[30] D. Ardila, A.P. Kiraly, S. Bharadwaj, et al., End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography, Nat. Med. 25 (6) (2019 Jun) 954–961.
[31] S.J. Devaraj, Chapter 2: emerging paradigms in transform-based medical image compression for telemedicine environment, in: D. Jude H, V.E. Balas (Eds.), Telemedicine Technologies, Academic Press, 2019, pp. 15–29, https://doi.org/10.1016/B978-0-12-816948-3.00002-7.
[32] A. Mansoor, U. Bagci, B. Foster, et al., Segmentation and image analysis of abnormal lungs at CT: current approaches, challenges, and future trends, Radiographics 35 (4) (2015) 1056–1076, https://doi.org/10.1148/rg.2015140232.
[33] W. Tan, P. Huang, X. Li, et al., Analysis of segmentation of lung parenchyma based on deep learning methods, J. X Ray Sci. Technol. 29 (6) (2021) 945–959, https://doi.org/10.3233/XST-210956.
[34] L.Y. Tseng, L.C. Huang, An adaptive thresholding method for automatic lung segmentation in CT images, in: AFRICON Conference 2009, IEEE, 2009, pp. 1–5, https://doi.org/10.1109/AFRCON.2009.5308100.
[35] N. Mesanovic, M. Grgic, H. Huseinagic, et al., Automatic CT image segmentation of the lungs with region growing algorithm, in: 18th International Conference on Systems, Signals and Image Processing-IWSSIP, 2011, pp. 395–400.
[36] B.A. Skourt, A. El-Hassani, A. Majda, Lung CT image segmentation using deep neural networks, Procedia Comput. Sci. 127 (2018) 109–113, https://doi.org/10.1016/j.procs.2018.01.104.

[37] S. Ioffe, C. Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift, in: International Conference on Machine Learning, pmlr, 2015 Jun 1, pp. 448–456.
[38] A. Said, W.A. Pearlman, An image multiresolution representation for lossless and lossy compression, IEEE Trans. Image Process. 5 (9) (1996) 1303–1310, https://doi.org/10.1109/83.535842.
[39] M. Everingham, L. Van Gool, C.K.I. Williams, et al., The PASCAL visual object classes (VOC) challenge, Int. J. Comput. Vis. 88 (2010) 303–338, https://doi.org/10.1007/s11263-009-0275-4.
[40] T.Y. Lin, M. Maire, S. Belongie, et al., Microsoft coco: common objects in context, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), In European Conference on Computer Vision (ECCV) 2014, vol. 8693, Springer Cham, 2014, pp. 740–755, https://doi.org/10.1007/978-3-319-10602-1_48.
[41] Z. Reitermanova, Data splitting, WDS'10 Proc. Contributed Papers 10 (2010) 31–36. Part I.
[42] K.K. Dobbin, R.M. Simon, Optimally splitting cases for training and testing high dimensional classifiers, BMC Med. Genom. 4 (2011) 31, https://doi.org/10.1186/1755-8794-4-31. Published 2011 Apr 8.
[43] C. Shorten, T.M. Khoshgoftaar, A survey on image data augmentation for deep learning, J Big Data 6 (1) (2019) 1–48.
[44] P. Chlap, H. Min, N. Vandenberg, et al., A review of medical image data augmentation techniques for deep learning applications, J Med Imaging Radiat Oncol 65 (5) (2021) 545–563, https://doi.org/10.1111/1754-9485.13261.
[45] B.C. Russell, A. Torralba, K.P. Murphy, W.T. Freeman, LabelMe: a database and web-based tool for image annotation, Int. J. Comput. Vis. 77 (1–3) (2008) 157–173, https://doi.org/10.1007/s11263-007-0090-8.
[46] S. Chen, J. Guo, C. Wang, et al., DeepLNAnno: a web-based lung nodules annotating system for CT images, J. Med. Syst. 43 (7) (2019) 197, https://doi.org/10.1007/s10916-019-1258-9. Published 2019 May 22.
[47] NVIDIA, King's College London, Medical open network for artificial intelligence, MONAI (September 30, 2022). https://monai.io/index.html.
[48] F. Pérez-García, R. Sparks, Ourselin S. TorchIO, A Python library for efficient loading, preprocessing, augmentation, and patch-based sampling of medical images in deep learning, Comput. Methods Progr. Biomed. 208 (2021), 106236, https://doi.org/10.1016/j.cmpb.2021.106236.
[49] T. Hastie, R. Tibshirani, J.H. Friedman, et al., The Elements of Statistical Learning: Data Mining, Inference, and Prediction, springer, New York, 2009 Aug.