Journal of Advanced Research 75 (2025) 421–433

Contents lists available at ScienceDirect

Journal of Advanced Research

journal homepage: www.elsevier.com/locate/jare

Full dimensional dynamic 3D convolution and point cloud in pulmonary nodule detection

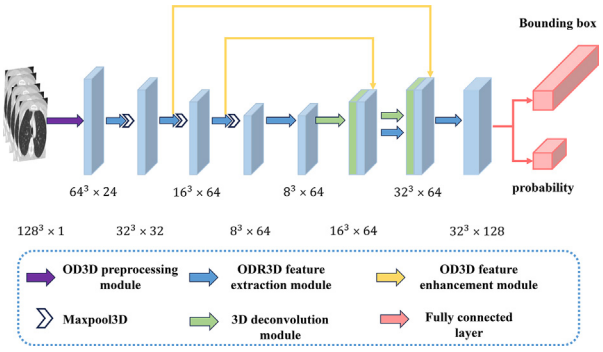
Yun Tie [a], Ying Wang [a,*], Dalong Zhang [a,*], Zepeng Zhang [a], Fenghui Liu [b], Lin Qi [a]

[a] School of Electrical and Information Engineering, Zhengzhou University, Zhengzhou, China
[b] The First Affiliated Hospital of Zhengzhou University, Zhengzhou University, Zhengzhou, China

HIGHLIGHTS

- We address the limitations of 3D CNNs on lacking adaptability.
- We propose a novel approach Omni-dimension Dynamic Residual 3D Net.
- Our algorithm achieves a high CPM score of 0.885.
- We explored the possibility of pulmonary CT point cloud format.
- We found that point clouds can also be used to detect lung nodules.

GRAPHICAL ABSTRACT



ARTICLE INFO

Article history:
Received 5 September 2023
Revised 6 October 2024
Accepted 28 November 2024
Available online 29 November 2024

Keywords:
Pulmonary nodule detection
Dynamic convolution
3D point cloud
CT images

ABSTRACT

Lung cancer is a leading cause of death worldwide, making early and accurate diagnosis essential for improving patient outcomes. Recently, deep learning (DL) has proven to be a powerful tool, significantly enhancing the accuracy of computer-aided pulmonary nodule detection (PND). In this study, we introduce a novel approach called the Omni-dimension Dynamic Residual 3D Net (ODR3DNet) for PND, which utilizes full-dimensional dynamic 3D convolution, along with a specialized machine learning algorithm for detecting lung nodules in 3D point clouds. The primary goal of ODR3DNet is to overcome the limitations of conventional 3D Convolutional Neural Networks (CNNs), which often struggle with adaptability and have limited feature extraction capabilities. Our ODR3DNet algorithm achieves a high CPM (Competition Performance Metric) score of 0.885, outperforming existing mainstream PND algorithms and demonstrating its effectiveness. Through detailed ablation experiments, we confirm that the OD3D module plays a crucial role in this performance boost and identify the optimal configuration for the algorithm. Moreover, we developed a dedicated machine learning detection algorithm tailored for lung 3D point cloud data. We outline the key steps for reconstructing the lungs in 3D and establish a comprehensive process for building a lung point cloud dataset, including data preprocessing, 3D point cloud conversion, and 3D volumetric box annotation. Experimental results validate the feasibility and effectiveness of our proposed approach.

Introduction

Lung cancer poses a serious health threat to the entire human population. Once afflicted with lung cancer, patients may face

☆ Peer review under responsibility of Cairo University.
* Corresponding authors at: Zhengzhou University, Science Avenue, Zhengzhou High-tech Zone, No. 100, China.
E-mail addresses: wy_2022_wy@gs.zzu.edu.cn (Y. Wang), iedlzhang@zzu.edu.cn (D. Zhang).

https://doi.org/10.1016/j.jare.2024.11.033
2090-1232/

immense suffering. According to statistics[5], the incidence and mortality rates of lung cancer in our country are very close, and over the past decade, the five-year survival rate for lung cancer has only improved by approximately 3.6%. This highlights the poor prognosis of lung cancer patients in our country, primarily due to the lack of early diagnosis and treatment. Study has shown a close correlation between the survival rate of lung cancer patients and disease staging[10], with earlier intervention leading to better outcomes. Stage I patients have a five-year survival rate of over 85%. However, the proportion of patients diagnosed at Stage I in our country is even less than 20%, indicating the critical value of reinforcing lung cancer screening and early diagnosis as the most effective approach to reducing the mortality rate among lung cancer patients and improving the current severe situation of lung cancer prevention and treatment.

Among various diagnostic approaches, imaging examinations are considered the most appropriate for early lung cancer screening. Clinically, they are recognized as the best non-invasive method for this purpose. Among the available imaging modalities, chest CT scans are regarded as the most effective tool for diagnosing lung cancer. In its early stages, lung cancer often appears as lung nodules on CT scans, making chest CT an essential method for detecting malignant nodules with a potential risk of cancer.

However, detecting pulmonary nodules in CT scans presents several significant challenges due to the specific characteristics of these nodules. Firstly, nodule similarity: Micronodules often exhibit considerable variation in shape and are discontinuously distributed, which can blur the distinction between nodules and normal lung tissue. Secondly, limited feature information: In commonly used lung nodule datasets, the samples often have low resolution and small annotated regions, providing less distinctive feature information. This scarcity of clear features makes the detection process more susceptible to noise interference. Thirdly, imbalanced dataset distribution: A common issue in lung nodule detection datasets is the imbalance between positive and negative samples. Specifically, small nodules are more prevalent, while medium and large nodules are underrepresented, potentially leading to biased detection performance. Finally, three-dimensional complexity: Lung nodule detection is inherently more challenging than two-dimensional detection tasks due to the three-dimensional nature of CT scans, adding further complexity to accurately identifying nodules.

With the rapid development of emerging technologies such as artificial intelligence, big data, and cloud computing, innovative solutions have emerged to enhance the success rate of lung nodule screening and alleviate the workload pressure on physicians. The latest edition of the Chinese guidelines for the diagnosis of pulmonary nodules [16] explicitly recommends the use of computer-aided detection (CAD) software combined with radiologist interpretation to improve the detection rate of pulmonary nodules. CAD systems have provided significant assistance to physicians in detecting and diagnosing various complex conditions. While they cannot replace physician decision-making entirely, they have been widely acknowledged as an important "second opinion" for physicians. Currently, CAD systems used in medical imaging can be broadly categorized into two types: computer-aided detection (CADe) systems and computer-aided diagnosis (CADx) systems. CADe systems aim to detect and localize lesion locations, significantly improving the efficiency of image interpretation. On the other hand, CADx systems focus on classifying or grading suspected lesion areas in terms of benign or malignant characteristics or severity. The PN task addressed in this study is the core of CADe systems.

The contributions of this study can be summarized in the following two aspects:

To address the limitations of traditional 3D CNNs in terms of adaptability and limited feature extraction capabilities, as well as to fully extract the stereoscopic features of lung nodule CT sequences, we propose a PND algorithm called Omni-dimension Dynamic Residual 3D Net (ODR3DNet). This algorithm adopts an encoder-decoder architecture and introduces the innovative concept of Omni-dimension Dynamic 3D Convolution (OD3D) to design ODR3DNet. Thanks to OD3D, this algorithm not only maintains powerful feature extraction capabilities for stereo features but also possesses dynamic and adaptive capabilities in the comprehensive dimensional sense, significantly improving the model performance and detection robustness.

The PND model based on 3D point clouds presents a preliminary application approach for the next generation of PND algorithms. This approach proposes key steps for lung 3D reconstruction and establishes a complete process for creating a lung point cloud dataset, which includes data preprocessing, 3D point cloud conversion, and 3D volumetric box annotation. We have developed a machine learning detection algorithm specifically designed for lung 3D point clouds and verified the feasibility of this approach. This study contributes to the exploration of the next generation of PND algorithms.

Related Work

The Pulmonary Nodule Detection (PND) algorithm is a crucial component of lung diagnostic CADe (Computer-Aided Diagnosis) systems. Its design objective is to leverage the powerful computational capabilities of computers to swiftly search for suspected lung nodules in a vast number of image slices and provide annotated positions for reference by physicians. Due to the stringent requirements of clinical medicine, PND algorithms typically need to achieve high sensitivity. PND algorithms have a relatively early origin, and their development can be traced through two major stages: the "traditional era," where manually designed features played a central role, and the "deep learning era," where algorithms intelligently extract features.

Among traditional rea, researchers have gradually realized that relying on expert-designed rules or traditional machine learning (which is essentially mathematical algorithms) has encountered bottlenecks and limited generalization capabilities. The emergence of deep learning (DL) algorithms, particularly the advent of Convolutional Neural Network (CNN) architectures for image tasks, has shown significant performance advantages, offering a glimmer of hope for PND algorithms. Through an investigation of current DL-based PND algorithms, existing algorithms can generally be divided into two categories: 2D CNN-based two-dimensional detection algorithms and 3D CNN-based three-dimensional detection algorithms.2D CNN-based algorithms emerged slightly earlier. Ding et al. [1] proposed a lung nodule detection method based on the Faster R-CNN algorithm, incorporating a deconvolutional structure. Setio et al.[21] introduced a multi-view CNN algorithm that fused CT slices from different orientations to generate the final result. George et al.[4] presented a 2D CNN regression-based detection algorithm using the YOLO structure.

The aforementioned works achieved significant performance improvements compared to traditional algorithms. However, CT images inherently consist of a series of slices, containing rich volumetric information. Multiple consecutive slices contain abundant 3D information, which is overlooked by 2D CNNs, making it suboptimal. Since the introduction of 3D CNNs, an increasing number of researchers have recognized their immense advantages. Currently, PND algorithms based on 3D CNNs remain predominant. Dou et al. [2] employed 3D CNNs to extract volumetric features and introduced a multi-level context extraction strategy to enhance model

robustness. Zhu et al.[28] presented a 3D Faster R-CNN detection algorithm that effectively extracted lung nodule features using a dual-path 3D extraction module. Huang et al.[9] introduced a parallel 3D CNN network with three branches, using CT sequences of different sizes as network inputs to enhance adaptability, and obtained the final detection result through fusion. Mei et al.[14] proposed the SGNL module, significantly enhancing the long-range dependency acquisition capability of CT images and achieving good performance.Then, there was a wave of attention, typical researches [27,26] introduce attention to PND and got good results. Zhu et al.[30]attention gate (AG) is introduced to reduce the false positive rate by employing critical feature dimensions at skip connections for feature propagation. Xu et al.[25]proposed a nodule detection method based on Slice Grouping Domain Attention (SGDA) to improve generalization performance, and the SGDA module worked in the axial, sagittal and coronal directions to explore the interdependencies of each set of feature maps in different directions.

ODR3DNet

A lung CT scan is essentially 3D data that contains rich stereoscopic features in the slice dimension, which is why PND algorithms based on 3D convolutional networks (3D CNNs) have shown better performance. However, due to the weight-sharing nature of CNNs, their lack of adaptability and dynamic feature extraction ability has become a key factor limiting the performance of CNN-based networks. To address this issue, some researchers, inspired by biological studies, have developed attention mechanisms that allow networks to adaptively focus on key features using specific algorithms, resulting in significant performance improvements. This led to the development of a new type of CNN known as dynamic CNNs, whose core idea is to transform the traditional static convolutional kernel into a dynamic linear combination of multiple kernels. Unlike attention mechanisms that operate on the convolution output layer, dynamic convolution directly acts on the convolutional kernel weights, enabling the kernels themselves to adapt to different data variations. The 3D CNN is still the most established architecture currently used for PND algorithms, known for its simple structure and high computational efficiency. It remains the most mature and cost-effective architecture for 3D feature extraction, and the significant advantages of dynamic convolution have provided us with further inspiration.

The overall structure of the proposed ODR3DNet is illustrated in Fig. 1. Different colored arrows represent different modules, and the numbers indicate the dimensions of the feature maps (width × height × slice number × channel number. Since all dimensions are cubes, they can be simplified as the cube edge length cubed × channel number.) The network follows an encoder-decoder architecture, where the feature map size progressively decreases in the left encoder part, allowing dynamic extraction of lung nodule features through the novel OD3D dynamic convolution. Subsequently, the feature map size is restored using a multi-level decoder, ensuring comprehensive recovery of detailed features. Additionally, by concatenating feature maps from the encoder and decoder with skip connections, richer feature fusion is achieved, further enhancing network performance. The entire algorithm is based on 3D CNN design, enabling the extraction of comprehensive stereoscopic features from CT images. Moreover, the use of the full-dimensional OD3D dynamic design endows the network with adaptability, leading to significant improvements in detection performance.

Encoder-decoder structure

The overall structure of this algorithm is inspired by the three-dimensional Region Proposal Network[17] and consists of an encoder, decoder, skip connection branch, and regression classification module. Leveraging the exceptional feature extraction capabilities and reliability of residual structures, this algorithm elevates the residual structure to the three-dimensional domain, forming the encoder architecture using three-dimensional residual modules. Furthermore, this algorithm introduces the concept of full-dimensional dynamic three-dimensional convolution (OD3D) and applies it to the residual structure, endowing the encoder with powerful adaptive feature extraction capabilities.

Since the encoder part extracts abstract features using max pooling, it leads to information loss and small-sized feature maps that are not conducive to subsequent candidate box regression. Therefore, based on the fundamental idea of U-Net[18], this algorithm designs the decoder structure. The decoder employs deconvolutional modules to upsample the feature maps, thereby increasing the size and resolution of the small-sized feature maps containing high-level abstract features. Importantly, the yellow feature-enhancement arrows shown in Fig. 1(a) represent the skip connections, allowing the fusion of lower-level, large-sized feature

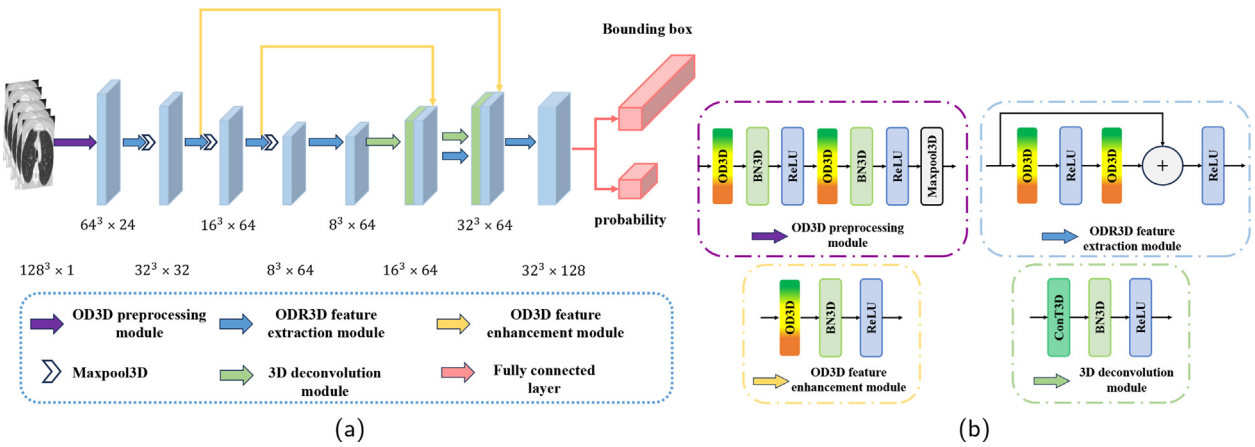


Fig. 1. Overall framework of ODR3DNet. (a) all modules are depicted, including the preprocessing module indicated by the purple arrow, the dynamic residual modules represented by the blue arrow, and the feature enhancement module represented by the yellow arrow. (b) The specific structures of each module. BN3D refers to three-dimensional batch normalization. Maxpool3D represents 3D max pooling. ConT3D denotes the three-dimensional deconvolution module.

maps with the decoder's images, enriching the features and significantly enhancing the overall network's detection performance.

OD3D-based feature extraction module

By replacing static convolutions with OD3D, this algorithm achieves comprehensive dynamic feature extraction capabilities, enabling adaptive feature extraction from different CT images using dynamic convolutional kernels. In Fig. 1(a), all modules containing OD3D are depicted, including the preprocessing module indicated by the purple arrow, the dynamic residual modules[8] represented by the blue arrow, and the feature enhancement module represented by the yellow arrow. The specific structures of each module are illustrated in Fig. 1(b). In Fig. 1(b), BN3D refers to three-dimensional batch normalization, which contributes to nonlinear transformations and enhances robustness, along with the ReLU activation function. Maxpool3D represents 3D max pooling, which extracts abstract features from three-dimensional input data, reduces dimensionality to decrease computation, mitigates overfitting, and improves training efficiency. ConT3D denotes the three-dimensional deconvolution module used for upsampling the feature maps in the decoder part. Except for the deconvolution module indicated by the green arrow, the other three modules are core components of feature extraction in the encoder.

Both OD3D and regular dynamic convolutions aim to transform traditional static convolutional kernels into adaptive kernels that vary with input data using SE attention. However, OD3D exhibits greater flexibility in terms of dynamics as it is fully dynamic in all dimensions, unlike DyConv, which is dynamic in a single dimension. To clarify the operational principle of OD3D, different convolutional kernels are directly represented by weights. Fig. 2(a) is an example of a common dynamic convolution DyConv.

By learning the linear combination of multiple convolutional kernels, static convolutions are transformed into a novel dynamic convolution with adaptive extraction capabilities. Dynamic convolutions enable the network to dynamically adapt without increasing its depth or width, thereby improving performance while maintaining efficiency. and the mathematical expression of Dynamic convolution is like (1):

$$\begin{aligned} Y &= (\alpha_1 W_1 + \alpha_1 W_1 + \ldots \alpha_N W_N) * X \\ &= \sum_{i=1}^{N} \alpha_i W_i * X. \end{aligned} \tag{1}$$

Fig. 2(a) and (1) provide a concise overview of the basic principles of single-dimensional dynamic convolution. In (1), α_i represents the weight of the convolutional kernel, which is entirely generated based on the input data. Through the harmonization of these weights, a regular convolutional kernel is upgraded to a linear combination of N kernels. This transformation turns the convolutional kernel from a static weight matrix into a dynamic weight matrix that adapts to different inputs.

The single-dimensional dynamic convolution, as shown in (1), consists of two components: the weight matrix W, representing the convolutional kernel, and the attention matrix α, used to harmonize the convolutional kernel itself. However, the design space of convolution algorithms is multidimensional, encompassing four dimensions: spatial dimension of the kernel, input channel dimension, output channel dimension, and overall dimension of the kernel. Yet, mainstream approaches such as DyConv only employ attention matrices in one dimension, specifically for each individual convolutional kernel. This dimension corresponds to the overall dimension of the kernel, while the other three dimensions are disregarded, severely limiting the flexibility of dynamic convolutions. Furthermore, DyConv's performance improvement primarily relies on having multiple convolutional kernels, typically with $N = 4$. This implies that applying DyConv extensively incurs a significant computational cost. If only one kernel is used ($N = 1$), the attention mechanism itself becomes one-dimensional, rendering it almost devoid of dynamics. Therefore, this dynamic design is undoubtedly coarse and has substantial room for improvement. Given the aforementioned theoretical foundation, the key to enhancing dynamic convolution lies in designing more refined dynamic strategies that fully embody its adaptability and superiority. This is precisely the core concept of OD3D. OD3D is a fully three-dimensional and full-dimensional dynamic convolution, as illustrated in Fig. 2(b).

Comparing Fig. 2(b) with Fig. 2(a), we can observe that their core principles are consistent, but OD3D is more complex. Both approaches employ attention mechanisms derived from the SE mechanism, but OD3D's attention matrix encompasses all four dimensions: spatial dimension α_s, input channel dimension α_c, output channel dimension α_{out}, and kernel dimension α_k. The activation functions used for the first three dimensions are Sigmoid (sig), while the kernel dimension activation function utilizes Softmax (sof). Thanks to the full-dimensional nature of OD3D, even with a single convolutional kernel ($N = 1$), significant performance improvements can be achieved. This represents a significant advantage over DyConv, which will be further discussed in the subsequent ablation experiment section. In OD3D, in the case of a single convolutional kernel, the kernel dimension is ignored, as indicated by the dashed box in Fig. 2(b). The formulation of OD3D is presented in (2).

$$Y = \left[\sum_{i=1}^{N} (\alpha_{si} \odot \alpha_{ci} \odot \alpha_{\text{outi}} \odot \alpha_{ki}) \odot W_i\right] * X. \tag{2}$$

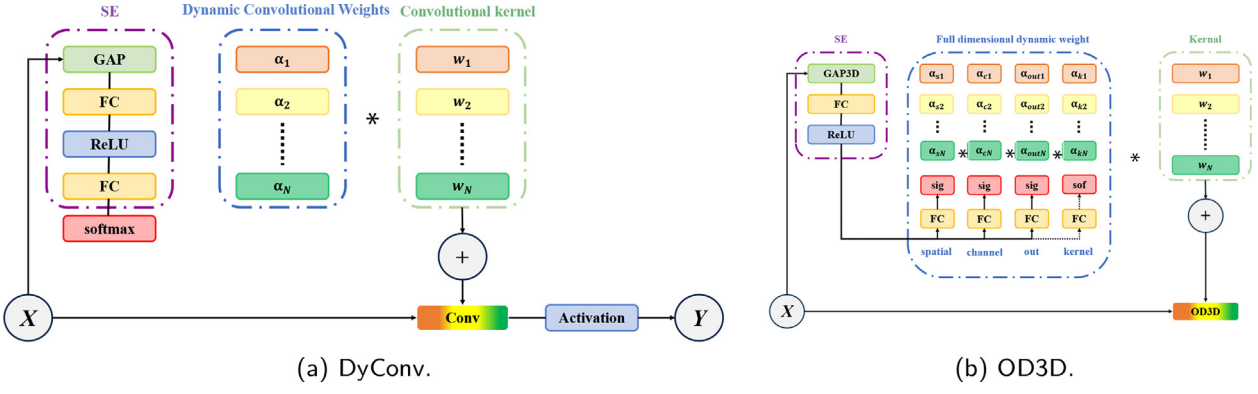


Fig. 2. Compare OD3D with DyConv.Both approaches employ attention mechanisms derived from the SE mechanism, but OD3D's attention matrix encompasses all four dimensions.

In (2) above, the spatial dimension attention $\alpha_{si} \in \mathbb{R}^{k\times k\times k}$, the input channel dimension attention $\alpha_{ci} \in \mathbb{R}^{1\times 1\times 1\times C^{in}}$, the output channel dimension attention $\alpha_{outi} \in \mathbb{R}^{1\times 1\times 1\times C^{out}}$, and the kernel dimension attention $\alpha_{ki} \in \mathbb{R}^{1\times 1\times 1\times C^{n}}$. Fig. 3 shows a concrete example of OD3D using two convolution cores($N = 2$), kernal size $K = 2(2 \times 2 \times 2)$.Then the schematic diagrams of spatial dimension, input channel dimension, output channel dimension and convolution kernel dimension are shown in Fig. 3 A, B, C and D respectively.

Classification module

For the PND algorithm, after the input data undergoes feature extraction through the encoder-decoder network, the generation of lung nodule detection boxes and their classification probabilities is handled by this module, typically located in the final part of the network. This module consists of two parallel and independent fully connected (FC) layers, one for bounding box regression and the other for classification probabilities. The bounding boxes follow the anchor theory, and this algorithm defines three anchor boxes with sizes [5,10,20] to adapt to different nodule sizes. The bounding box FC layer regresses the three-dimensional coordinates and diameter of the lung nodules, while the classification FC layer outputs the probability of the detected object being a lung nodule.

Loss Functions

The loss function of this algorithm consists of two parts: the classification loss L_c and the bounding box regression loss L_r. To address the issue of imbalanced classification samples and prevent the network from paying excessive attention to incorrect samples, this algorithm incorporates the focal loss[13] into L_c. The classification FC layer of this algorithm outputs the probability of the detected object being a lung nodule, denoted as P. The formulation for L_c is as follows:

$$L_c = -\alpha(1 - P_t)^{\gamma} \log(P_t). \tag{3}$$

In (3), if the target region is indeed a lung nodule, then $P_t = P$; otherwise, it indicates a classification error and $P_t = 1 - P$. α and γ are harmonic hyperparameters, and in this algorithm, they are set as $\alpha = 0.5$ and $\gamma = 2$.

The bounding box regression loss, L_r, is defined as follows:

$$L_r = \sum_k L1(G_k, Pk). \tag{4}$$

In (4), $L1(.)$ denotes the L1 regularization function, and G_k and P_k represent the ground truth relative coordinates and predicted relative coordinates, respectively. They are defined as follows:

$$\begin{aligned} G_k &= \left(\tfrac{x_g - x_\alpha}{r_\alpha}, \tfrac{y_g - y_\alpha}{r_\alpha}, \tfrac{z_g - z_\alpha}{r_\alpha}, \log\tfrac{r_g}{r_\alpha}\right), \\ P_k &= \left(\tfrac{x - x_\alpha}{r_\alpha}, \tfrac{y - y_\alpha}{r_\alpha}, \tfrac{z - z_\alpha}{r_\alpha}, \log\tfrac{r}{r_\alpha}\right). \end{aligned} \tag{5}$$

(x_g, y_g, z_g, r_g) represents the ground truth coordinates of the label, while (x, y, z, r) represents the predicted coordinates by this algorithm. $(x_\alpha, y_\alpha, z_\alpha, r_\alpha)$ denotes the anchor coordinates, and k represents the batch size during training. Thus, combining Eqs. 4,5, the model continuously adjusts the predicted coordinates based on the ground truth values.

The final loss function of this algorithm, denoted as L, combines L_c and L_r and is defined as $L = L_c + \delta L_r$. Only when the predicted result is a positive sample (lung nodule), $\delta = 1$, indicating further regression of the bounding box. Otherwise, when $\delta = 0$, the bounding box regression operation is skipped directly.

Experiments

Dataset

The Pulmonary Nodule Detection (PND) dataset in this experiment is LUNA16[22]. The LUNA16 dataset provides the CT source files and their corresponding format information for each patient. Additionally, the dataset includes annotation information for all lung nodules in the form of a.csv file. To facilitate the use of the dataset, the LUNA16 official also provides lung parenchyma mask files, which are essential for extracting the lung parenchyma from the original CT scans. Extracting the lung parenchyma is crucial for algorithm training, as our algorithm utilizes a three-dimensional CNN for feature extraction, and the network input consists of three-dimensional CT data, which has a high computational

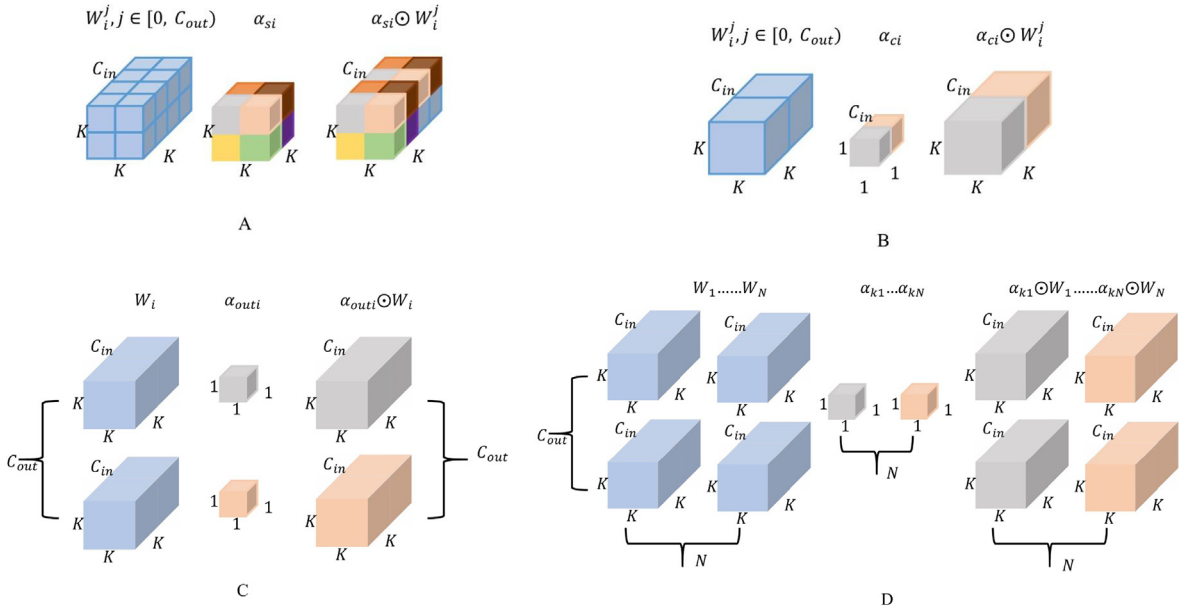


Fig. 3. OD3D full-dimensional dynamic convolution schematic. (A) spatial dimension, (B) input channel dimension, (C) output channel dimension and (D) convolution kernel dimension.

complexity. By removing irrelevant tissues such as the chest wall and bones that are not related to PND, we not only reduce computational load but also prevent potential performance degradation. Therefore, the effective extraction of the lung parenchyma is a crucial step in the data preprocessing pipeline.

In our algorithm, we utilize the provided mask images from the dataset and employ the Pydicom and Numpy libraries to read the original CT data. By performing a logical "AND" operation between the CT data and the mask files, we successfully extract the lung parenchyma. Furthermore, we perform interpolation to resample the extracted lung parenchyma for efficient algorithm processing. Subsequently, we apply a further reduction of interference by preserving only the pixel values within the range [-1200, 600] Hounsfield Units (HU). We then normalize the pixel values within this range to the range [0, 1] for algorithm training.

Due to the computational complexity of 3D CNN, it is not practical nor meaningful to feed the processed CT files into the network for training in their entirety. Therefore, our algorithm only extracts local CT data with dimensions of 128 × 128 × 128 (height × width × number of slices) from each patient for network training. During network testing, larger image sizes are extracted to approximate the clinical reality (208 × 208 × 208). Due to the limited size of the dataset, we apply necessary data augmentation strategies to improve training performance. Specifically, these strategies include random cropping, random flipping, and random scaling (with a scaling range of [0.75, 1.25]).

Experimental setting

The algorithm employs the stochastic gradient descent (SGD) optimizer for training, with a dynamic learning rate. The initial learning rate is set to 0.01 and the training is conducted for a total of 150 epochs. After 100 epochs, the learning rate is reduced to 0.001, and further reduced to 0.0001 after 120 epochs. The batch size is set to 8, and the code is developed using Python 3.7 and PyTorch 1.6.0. The algorithm training and experiments are conducted on a large-scale server equipped with four NVIDIA TITAN GPUs (12 GB each). The system operates on Ubuntu 20.04, and CUDA 10.0 is utilized for GPU acceleration.

To adhere to the requirements of LUNA16, the algorithm follows a ten-fold cross-validation during training and validation to ensure stable experimental results and enhance model robustness. The algorithm employs the free-response receiver operating characteristic (FROC) curve and the competitive performance measure (CPM) as evaluation metrics, where sensitivity is a necessary parameter. It is calculated using the following equation:

$$Sensitivity = \frac{TP}{TP + FN}. \tag{6}$$

In (6), TP represents the number of correctly predicted lung nodules, while FN represents the number of true lung nodules falsely classified as false positives.

$$CPM = \frac{\sum_{7}^{i=1} S_i}{7}. \tag{7}$$

Following the official evaluation criteria, the free-response receiver operating characteristic (FROC) curve is obtained by calculating the sensitivity at different false positive rates (FPs/scan) ranging from 0.125 to 8 per CT image. The competitive performance measure (CPM), shown in (7) is an average sensitivity across seven predefined average false positive rates: 1/8, 1/4, 1/2, 1, 2, 4, and 8.

Comparative experiment and analysis

In this section, we provide a detailed comparative analysis of the performance of current state-of-the-art PND algorithms. All algorithms were trained on the LUNA16 dataset, and the experimental results are presented in Table 1. When FPs/scan = 0.5,1,2, it acquires best performance. For other conditions, although ODR3DNet did not achieve the best performance, it also achieved the second best performance and was very close to the best figure. In addition, the experimental results clearly demonstrate the superiority of the proposed ODR3DNet, a full-dimensional residual dynamic convolutional network, which achieves the highest CPM performance among the compared algorithms.

To visually demonstrate the performance of our algorithm, Fig. 4 presents the visualization results of four cases. As shown in Fig. 4, the predicted probabilities and bounding boxes reflect the accuracy of our algorithm. Our algorithm is capable of accurately detecting lung nodules of different shapes and sizes. Even for challenging cases like the one depicted in Fig. 4(D), where the nodules are small and difficult to notice, our algorithm can still achieve accurate detection. Although the predicted probability may be relatively low, it still showcases the powerful detection capability of ODR3DNet.

Ablation experiment and analysis

In this section, we perform an ablation analysis on key parameters and core components. For ODR3DNet, the most critical hyperparameter is the number of convolutional kernels, denoted as "n" in each dynamic convolution module. The core component is the OD3D full-dimensional dynamic convolution. The experimental results are presented in Table 2, Table 3.In particular, DyConv in Table 3 is from reference [11].

Table 2 illustrates the impact of different numbers of convolutional kernels in the OD3D dynamic convolution module. To clearly demonstrate the superiority of dynamic convolution, we intentionally replaced all OD3D instances with regular 3D CNN for comparison. The results show that the performance is highest when n = 4, but it comes at the cost of significant memory consumption, with only marginal improvement compared to n = 2. The comparison with 3D CNN confirms that even with a single convolutional kernel, the full-dimensional dynamic mechanism still leads to

Table 1
Performance comparison of mainstream PND algorithms. The experimental results clearly demonstrate the superiority of the proposed ODR3DNet, a full-dimensional residual dynamic convolutional network, which achieves the highest CPM performance among the compared algorithms.

FPs/scan	0.125	0.25	0.5	1	2	4	8	CPM
Zhu et al.[29]	0.692	0.769	0.824	0.865	0.893	0.917	0.933	0.842
Tang et al.[23]	0.652	0.768	0.839	0.875	0.911	0.929	0.934	0.844
Li et al.[12]	**0.739**	**0.803**	0.858	0.888	0.907	0.916	0.920	0.862
Mei et al. [15]	0.712	0.802	0.865	0.901	0.937	0.946	0.955	0.874
Han et al. [7]	0.679	0.789	0.843	0.891	0.920	0.940	0.947	0.861
Xu et al.[25]	0.591	0.651	0.775	0.853	0.931	**0.963**	**0.967**	0.819
Wu et al.[24]	0.709	0.772	0.835	0.883	0.917	0.938	0.956	0.859
ODR3DNet	0.722	0.795	**0.887**	**0.928**	**0.941**	0.958	0.961	**0.885**

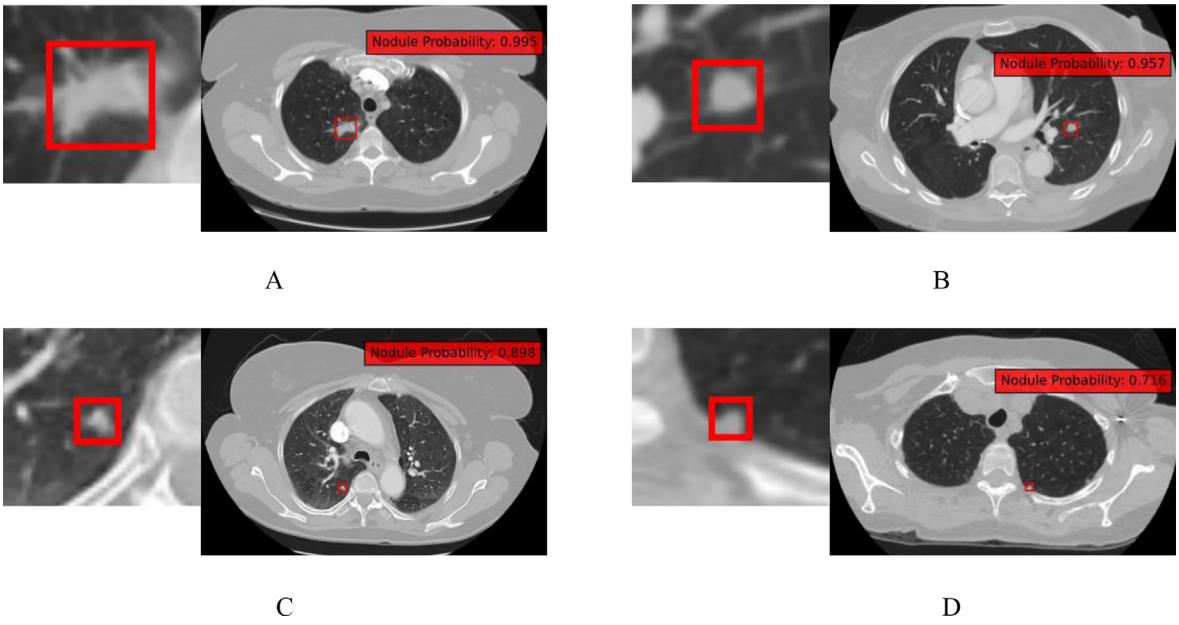


Fig. 4. Visualization of ODR3DNet detection results.

Table 2
Number of convolutional kernal in OD3D module(*is best). The results show that the performance is highest when n = 4, but it comes at the cost of significant memory consumption, with only marginal improvement compared to n = 2.

FPs/scan	0.125	0.25	0.5	1	2	4	8	CPM	GPU Memory
3D CNN	0.682	0.775	0.821	0.865	0.886	0.908	0.934	0.839	11.768
1	0.709	0.787	0.868	0.896	0.921	0.932	0.947	0.866	13.463
2*	0.722	0.795	0.887	0.928	0.941	0.958	0.961	0.885	22.587
4	**0.735**	**0.801**	**0.897**	**0.934**	**0.946**	**0.965**	**0.971**	**0.893**	41.268

Table 3
Contribution of OD3D full-dimensional dynamic three-dimensional convolution(*is best). By replacing the 3D CNN structure with DyConv, the performance improvement brought by dynamic convolution is reflected. OD3D demonstrates greater flexibility, adaptability, and dynamic nature compared to the single-dimensional DyConv structure, resulting in better performance.

FPs/scan	0.125	0.25	0.5	1	2	4	8	CPM
3D CNN	0.682	0.775	0.821	0.865	0.886	0.908	0.934	0.839
DyConv	0.714	0.781	0.835	0.872	0.904	0.918	0.936	0.851
OD3D*	**0.722**	**0.795**	**0.887**	**0.928**	**0.941**	**0.958**	**0.961**	**0.885**

significant performance gains. Setting n = 2 achieves true ”full-dimensionality” while maintaining a balanced performance, making it the optimal choice.

Table 3 illustrates the limitations of regular static convolution compared to dynamic convolution. By replacing the 3D CNN structure with DyConv, the performance improvement brought by dynamic convolution is reflected. However, OD3D demonstrates greater flexibility, adaptability, and dynamic nature compared to the single-dimensional DyConv structure, resulting in better performance.

This algorithm is entirely based on a three-dimensional convolutional neural network, which can effectively extract the stereoscopic features of lung nodules. The algorithm employs an encoder-decoder architecture and incorporates skip connections to merge features from different levels. The key component of this algorithm is the OD3D module, a fully dimensional dynamic three-dimensional convolution module, which replaces traditional three-dimensional convolutions entirely. OD3D can leverage all dimensions of the convolutional kernel design space, enabling full-dimensional dynamic feature extraction and empowering the algorithm with strong adaptive feature extraction capabilities.

By comparing it with mainstream PND algorithms, our algorithm achieves a remarkable CPM score of 0.885, demonstrating significant superiority. The ablation experiments have firmly confirmed the contribution of OD3D to performance improvement and the advantages of the final configuration.

Point cloud-based pulmonary nodule detection applications

In the field of PND algorithm research, there is a contradiction between the inherent three-dimensional nature of CT images and the predominant focus on two-dimensional images. This contradiction prompted us to explore new possibilities for the next generation of PND algorithms. The advancements in modern medical three-dimensional reconstruction techniques have provided great inspiration for our study. Many frontline clinicians have been actively utilizing the computational power of computers to reconstruct complex lesions into three-dimensional models, enabling

more accurate diagnoses. The modern concepts of diagnosis and treatment, along with cutting-edge medical imaging technologies, have served as key sources of inspiration for our research.

After further investigation, we found that the field of autonomous driving and other domains have witnessed the emergence of point cloud-based three-dimensional detection algorithms. This demonstrates the feasibility of transitioning from traditional object detection algorithms based on two-dimensional images to the use of "full three-dimensional" point cloud data. In this chapter, we attempt to combine these two domains and propose an approach based on lung point cloud models. We explore a "true three-dimensional" PND algorithm based on lung point cloud data, presenting our thoughts on the future direction of PND algorithms.

Firstly, we introduce the methods for three-dimensional reconstruction of the lungs, followed by an explanation of the steps involved in creating a lung point cloud dataset. We then utilize machine learning and point cloud clustering techniques to design the detection algorithm. Objective analysis of the experimental results is provided to demonstrate the feasibility of our proposed approach.

Lung 3D point cloud dataset

High-quality datasets are prerequisites and essential for advancing research. Currently, almost all datasets consist of two-dimensional CT sequences. Therefore, the first step in this approach is to create a lung point cloud dataset. Dataset creation is the most time-consuming and labor-intensive step, and due to the limited availability of relevant references, the technical roadmap for generating lung three-dimensional datasets is not yet clear, making the process challenging. To address this, we have deepened our collaboration with affiliated hospitals, sought input from frontline clinical physicians specializing in radiology and thoracic surgery, and particularly valued the opinions of young physicians who are skilled in using modern three-dimensional reconstruction techniques for diagnostic assistance. Additionally, drawing inspiration from KITTI[3], this section proposes the technical route of generating lung point cloud data set.

Medical 3D reconstruction software and principles

Unlike two-dimensional CT sequences that can be directly obtained, the most crucial step in creating a lung point cloud dataset is how to reconstruct the two-dimensional CT sequences into three-dimensional models, which forms the foundation of this research. Unlike general object recognition, open-source 3D reconstruction algorithms cannot be used for creating lung point cloud datasets due to their insufficient reconstruction accuracy that does not meet the stringent requirements of clinical medicine. Therefore, the creation of lung three-dimensional datasets must be implemented using professional medical 3D reconstruction software.

Currently, medical 3D reconstruction software can be categorized into the following types: 1) Software development kits such as ITK and VTK[6], which are developed in the C++ programming language. These development kits provide platforms for medical algorithm programmers to develop 3D reconstruction systems. 2) Professional medical 3D reconstruction software represented by Mimics, OsiriX, DeepInsight, CTTRY, and IQQA. These clients have powerful interactive interfaces that facilitate physician operations to the maximum extent. 3) PACS professional reconstruction software built into modern CT equipment workstations in hospitals.

These three types of medical reconstruction software each have their advantages and disadvantages. The first type of software development kit can be essentially understood as a C++ standard library function that provides a large number of classes and functions for developers to use. It offers high flexibility, cross-platform compatibility, and portability. However, it is challenging to operate and requires professional developers to use, and its reconstruction accuracy is inferior to the other two types.

The third type of PACS software is the software currently used for clinical diagnosis and is specific to hospitals. This type of software is usually developed by well-known CT equipment manufacturers and comes bundled with CT equipment workstations. These manufacturers encompass numerous top-tier medical equipment companies, such as GE Healthcare, SIEMENS Healthineers, PHILIPS, etc. These companies represent the pinnacle of CT equipment research and development worldwide. Therefore, the three-dimensional reconstruction software they develop undergoes rigorous medical equipment licensing and testing. These PACS software packages achieve precision levels that meet clinical standards. However, they are not suitable for medical research as they are often closed software. Due to medical ethics and regulatory constraints, they are difficult to use directly for research purposes and do not have data processing capabilities.

Through collaboration with thoracic surgeons who have long been engaged in medical 3D reconstruction research at our affiliated hospital, this section has determined the use of Mimics as the medical 3D reconstruction software. Mimics belongs to the second type of software mentioned earlier, providing a comprehensive user interface and data export capabilities, making it highly suitable for medical research, preoperative planning, and medical 3D printing design work. Mimics is developed by Materialise, a Belgian company, and has obtained medical device certifications in multiple countries worldwide, including the United States, the European Union, and Japan. Its reconstruction accuracy meets clinical requirements. The software features a user-friendly graphical interface and does not require tedious development, making it easy to use. Moreover, it has robust data conversion and export functions, which align perfectly with the objectives of this research. Mimics offers both a Medical version tailored for clinical applications and a Research version targeted at researchers. The Research version is selected to access more research-oriented features.

Similar to other CT-based 3D reconstruction algorithms, Mimics operates on the principle of utilizing the differences in CT values. CT images represent different tissues with varying grayscale intensities. In medical practice, these differences are quantified using CT values and expressed in Hounsfield Units(HU), which is a measure of the density of specific tissues or organs in the human body. Mimics employs advanced digital algorithms to perceive these differences and, after selecting the desired reconstruction region, applies a combination of various algorithms to overlay, fill, and render the CT sequence into a complete 3D model.

Reconstruction of lung 3D model

The first step is to determine the regions that need to be reconstructed. Given the lack of consensus in the academic community regarding the creation of lung 3D datasets, our team has discussed with the first-line physicians of the cooperative affiliated hospitals for many times. Taking into account the research objectives, we have ultimately decided to reconstruct the patient's trachea, pulmonary vessels (without distinguishing between arteries and veins, and excluding interfering tissues such as the heart), and the surrounding regions of pulmonary nodule lesions for this dataset.

To ensure model accuracy, the original CT data are all high-resolution (with an average of over 150 slices per patient). All data has been strictly desensitized to fully protect patient privacy. The lesion site of each patient was jointly determined by experienced imaging physicians and respiratory physicians. The whole process of generating 3D models was guided by physicians and could be stored in the database after being reviewed by physicians. All the researchers involved in this project have theoretical basis in med-

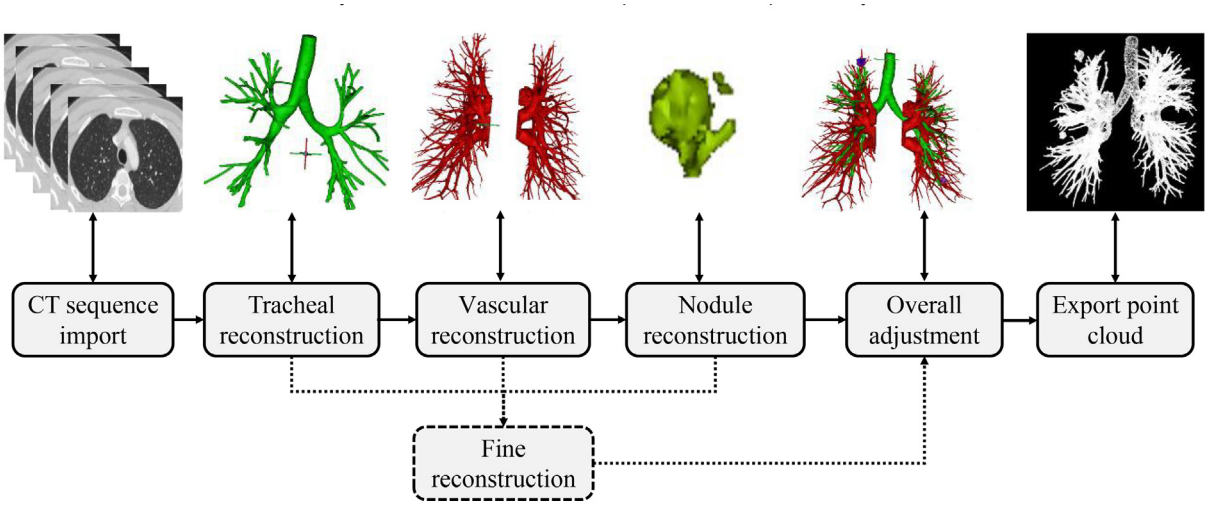


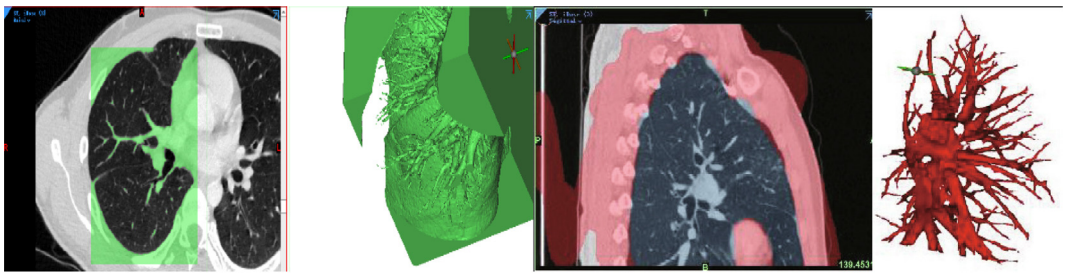
Fig. 5. Core steps of 3D model reconstruction. For trachea reconstruction, the automatic segmentation feature is utilized to quickly extract the tracheal structure from the CT images; for blood vessels and pulmonary nodule lesions, manual selection is performed accurately in the coronal, sagittal, and axial views of the CT, and the reconstruction is completed using the CT threshold segmentation feature.

ical anatomy, experience in CT interpretation, and proficiency in Mimics software. The specific steps for 3D reconstruction are shown in Fig. 5.

This reconstruction process makes full use of the powerful SEGMENT tissue segmentation functionality in Mimics. It involves techniques such as using rectangular boxes for localization and CT thresholding to extract the predefined target tissues. The methods for reconstructing the three types of tissues are different: for trachea reconstruction, the automatic segmentation feature is utilized to quickly extract the tracheal structure from the CT images; for blood vessels and pulmonary nodule lesions, manual selection is performed accurately in the coronal, sagittal, and axial views of the CT, and the reconstruction is completed using the CT threshold segmentation feature. However, achieving such high-quality models in the actual reconstruction process requires more than just conventional reconstruction methods. Based on anatomical principles and physiological theories, the lungs and the heart are both located within the thoracic cavity of the human body. Therefore, the CT images of the lungs inevitably contain structures of the heart. Additionally, the lungs have a dense network of blood vessels, and the heart is a vital organ responsible for pumping blood. It is not possible to differentiate them based on CT values alone. Furthermore, due to the limitations of using rectangular boxes for region selection during segmentation, the selected regions do not align perfectly with the lung parenchyma, resulting in significant interference from irrelevant tissues such as the chest wall. This issue severely affects the final reconstruction results, as shown in Fig. 6(a).

The left image shows the Mimics interface, where the green area represents the region for 3D reconstruction, and the right image shows the corresponding reconstruction result. Whether observed from the left CT images or the reconstructed model on the right, the lung structures that truly need to be preserved are interfered with by irrelevant tissues such as the chest wall and the heart. The reconstruction results on the right are completely inconsistent with the research expectations, significantly compromising the quality of the model. This is precisely the significance of "fine reconstruction" depicted in Fig. 5. Only through fine reconstruction and meticulous removal of irrelevant tissues can a high-quality 3D dataset be obtained. Under the guidance of medical professionals, this section employs the advanced segmentation feature of Split Mask to obtain a more refined pulmonary vascular model by annotating the irrelevant regions in the sagittal view, as illustrated in Fig. 6(b).

The left side of Fig. 6(b) shows the fine reconstruction window, where the red region represents the irrelevant area, and the blue region represents the reconstruction area. By specifying the red and blue regions in multiple slices, Mimics automatically filters the tissues that truly need to be reconstructed, resulting in a more refined 3D reconstruction model of the pulmonary vasculature, as shown on the right side of the image. Through the 3D reconstruction steps described in this section, the 2D CT images of the lungs



(a) Serious disturbances (b) Schematic diagram of fine reconstruction and results after reconstruction

Fig. 6. Results of reconstruction at different stages. (a) Significant interference from irrelevant tissues such as the chest wall. (b) A more refined pulmonary vascular model by annotating the irrelevant regions in the sagittal view.

are transformed into a 3D model, laying an important foundation for establishing the dataset. For data export, the original 3D point cloud is saved in the.txt format, where each point is represented by its spatial coordinates. In this study, the average number of points in the 3D model exceeds 160,000, providing a detailed depiction of the 3D structures.

Lung point cloud data preprocessing

A point cloud is a representation of three-dimensional structures using a large number of three-dimensional coordinate points. While there are other forms of three-dimensional representations, such as polygons and triangles, point clouds are capable of more accurately fitting three-dimensional structures and capturing finer details. However, a disadvantage of point clouds is their often substantial data volume. In the case of the dataset in this chapter, the average number of points in the 3D models is as high as 160,000, indicating that each patient's 3D lung model consists of over 100,000 three-dimensional coordinate points. The significant data volume is evident.

Since the point cloud files exported by Mimics are in the.txt format, which has slow reading speed and is not suitable for quick program reading, this section includes preprocessing code written in Python to convert the point cloud files into binary.bin format, significantly improving data reading speed. Additionally, the preprocessing program includes functionalities such as setting the coordinate origin and coordinate compression. To enhance dataset compatibility, the program also generates files in the.pcd (Point Cloud Data) format, which is widely used for point cloud data.

After the preprocessing steps described above, the newly generated point cloud data in both formats have undergone coordinate compression and normalization, satisfying the requirements for point cloud visualization. To visually inspect the effects of the preprocessing on the dataset, this section includes a point cloud visualization program developed using the Python Mayavi 3D visualization library. This program takes the point cloud in.bin format as input and directly displays its three-dimensional visualization structure. For point cloud data, each feature is represented by a large number of discrete and disordered three-dimensional points.

Point cloud data 3D annotation

Data annotation is a crucial step for the dataset. Given the absence of specific annotation standards for lung 3D point cloud datasets in the current academic community, this section refers to the annotation guidelines of the widely used KITTI 3D point cloud dataset to design the annotation specifications for the lung 3D dataset. However, since point cloud data is a three-dimensional distribution in space, it is fundamentally different from the "bounding box" annotation method used in 2D images. In this section, data annotation is performed using the LabelCloud point cloud annotation software, which was developed by Sager et al. [20] and released in 2021. This software supports popular point cloud formats and is capable of generating KITTI-format labels.

To enhance compatibility with the dataset, the label format for the lung point cloud dataset was also designed based on the KITTI

Table 4
Lung 3D point cloud data set annotation format.

digits	1	2 ∼ 8	9∼ 11	12∼ 14	15
meaning	Type	Zero setting	Dimensions	Location	Rotation

Table 5
KITTI label format.

digits	1	2	3	4	5∼ 8	9∼ 11	12∼ 14	15
meaning	Type	Truncated	Occluded	Alpha	Bbox	Dimensions	Location	Rotation

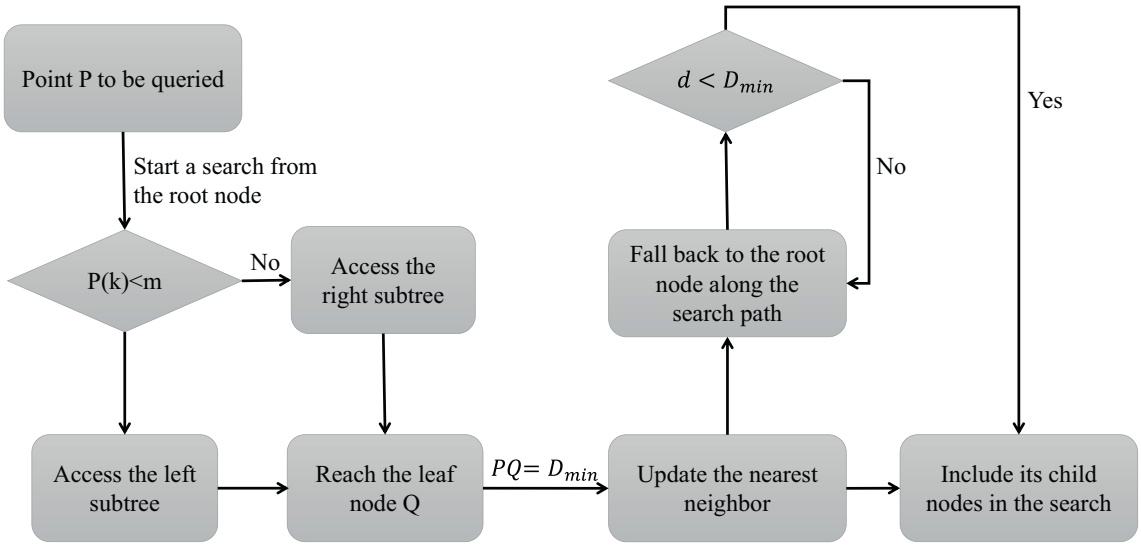


Fig. 7. KD-Tree nearest neighbor search algorithm.

label format,presented in Table 5. The label format for the lung point cloud dataset is presented in Table 4:

By comparing Table 5 and Table 4, it is evident that the formats of the two tables are essentially consistent. The proposed annotation format in this section is primarily based on the specific considerations of the lung dataset, wherein several metrics from the original KITTI dataset have been removed. Overall, the proposed format maintains consistency with the KITTI labels by zeroing out certain attributes.

Principle and design of 3D point cloud detection algorithm

Based on the principles of machine learning, this section develops a PND algorithm based on a three-dimensional lung point cloud dataset. The algorithm is developed using the PCL (Point Cloud Library) C++ runtime library. PCL [19] is a C++ programming library that integrates various advanced algorithms for point cloud processing. It provides built-in subclasses for point cloud reading, filtering, segmentation, and more, allowing developers to quickly develop point cloud processing algorithms. This algorithm is entirely implemented in C++ and compiled using Visual Studio Code. The program development, compilation, and deployment were performed on Ubuntu 20.04, using the 2020 version of the PCL library.

Machine learning algorithms are based on mathematical theories and use mathematical algorithms to process data, providing complete interpretability. The goal of this section is to automatically identify the positions of lung nodules in the generated three-dimensional lung point cloud dataset and annotate them with three-dimensional bounding boxes, which is a typical three-dimensional object recognition algorithm. Considering the specifics of this research, the section establishes a design approach centered around point cloud clustering algorithms. Clustering is one of the most widely used tasks in machine learning, aiming to partition data into different categories according to certain patterns. In the case of point cloud clustering, points that are close in three-dimensional space may belong to the same category. Therefore, by analyzing the differences in distances, lung nodule lesions can be accurately classified and their positions can be determined through bounding box selection.

This algorithm is designed based on the principle of Euclidean distance clustering. To improve the runtime efficiency, a KD-Tree nearest neighbor search [20] is used as a preliminary step. The algorithm flow is illustrated in Fig. 7, where $P(k)$ represents the value of the current query point, and m is the threshold for node partitioning. If the value is less than the threshold, the algorithm visits the left subtree; otherwise, it visits the right subtree, until reaching a leaf node Q. At this point, Q becomes the current nearest neighbor, and the shortest distance is the distance between P and Q. Then, the algorithm backtracks along the original search path to the root node. If a closer point is found during this process, the unvisited child nodes are included in the search range, and the nearest neighbor point is updated accordingly. This process continues until the search path is empty. Through the nearest neighbor search algorithm, all the neighboring points of a specific query point are discovered, and they are highly likely to be assigned to the same cluster. Subsequently, the point cloud clustering algorithm based on the KD-Tree can be executed. Euclidean clustering refers to clustering algorithms based on the Euclidean distance, which requires calculating the Euclidean distance between two points in three-dimensional space. Assuming point 1 is represented as (x_1, y_1, z_1) and point 2 as (x_2, y_2, z_2), the Euclidean distance is defined by (8):

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}. \tag{8}$$

Based on the nearest neighbor search algorithm illustrated in Fig. 7, the Euclidean point cloud clustering algorithm is straightforward to understand, and its core principle is the three-dimensional Euclidean distance shown in (8). For a point P in space, the algorithm first uses the nearest neighbor search algorithm to find k nearest neighbor points, and all points with a Euclidean distance smaller than a threshold are added to the point set Q. If there are no more additions to the set, the clustering process ends. Otherwise, another point from the set, excluding P, is selected, and the algorithm is rerun until the number of elements in the set no longer increases.

Table 6
C++ core class used.

steps	C++ core class name used
3D view initialization	pcl::visualization
Point cloud reading	pcl::io
Euclidean clustering	1) pcl::search 2) pcl::EuclideanClusterExtraction
Three-dimensional bounding box	pcl::getMinMax3D

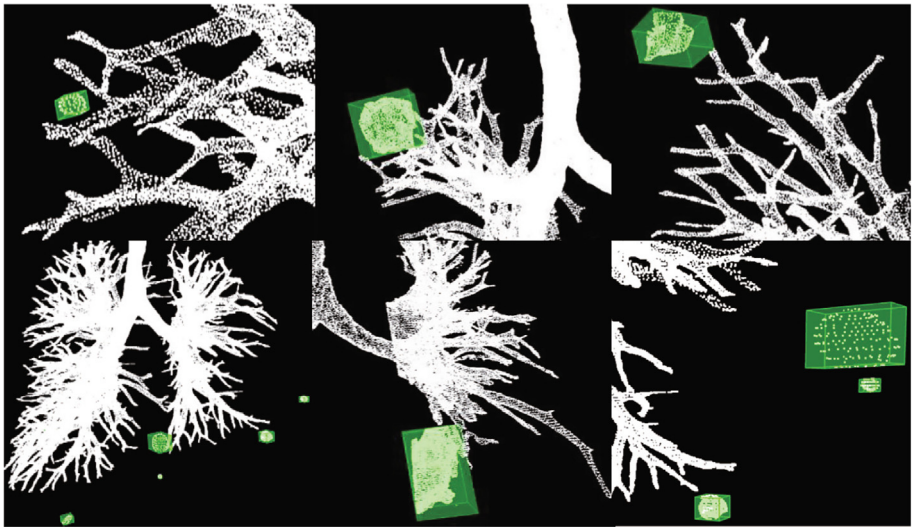


Fig. 8. Point cloud data set detection results.

Fig. 9. Limitations of 3D point cloud detection algorithm based on machine learning.

The algorithm designed in this section is presented in Table 6. The proposed algorithm is implemented using the pcl point cloud library. The initialization of the three-dimensional view aims to generate a visualization window for the detection results. Point cloud reading is performed using the io class, which retrieves data by pointing to the address of the point cloud dataset. The Euclidean clustering is the core of the algorithm, using the search class in the pcl library to build a KD-Tree, followed by executing the nearest neighbor search algorithm. Subsequently, the EuclideanClusterExtraction class performs the three-dimensional Euclidean clustering algorithm. After clustering is completed, the getMinMax3D class is used to calculate three-dimensional bounding boxes for detection.

Experimental results

Fig. 8 presents six typical detection results, where it is evident that the size, location, and relative distance to the trachea and blood vessels vary in each case. However, the algorithm accurately identifies the spatial position of the lung nodules through the use of green three-dimensional bounding boxes. These results demonstrate the effectiveness of the algorithm and confirm the applicability of the lung three-dimensional dataset proposed in this chapter for PND. The feasibility of lung nodule detection based on three-dimensional point clouds is validated.

Of course, due to the limitations of the machine learning algorithm, there are also examples of mistaken results, such as Fig. 9, where the tissue at the end of a normal blood vessel is recognized as a pulmonary nodule.

The proposed ideas for the next generation of PND algorithms, the steps and implementation methods for creating a three-dimensional lung dataset, have pioneering implications for future PND algorithm research. It is hoped that the work presented in this chapter will encourage further exploration beyond the mainstream direction and inspire new possibilities in PND algorithm research.

Conclusion

The adaptive PND algorithm ODR3DNet is designed entirely based on a three-dimensional convolutional neural network, which can effectively extract the stereoscopic features of lung nodules. The algorithm adopts an encoder-decoder architecture and incorporates skip connections to fully fuse features at different levels. In this algorithm, a full-dimensional dynamic three-dimensional convolution module, OD3D, is designed and used to replace traditional three-dimensional convolutions. OD3D can utilize all dimensions of the convolutional kernel to achieve dynamic feature extraction in all dimensions, thereby endowing the algorithm with powerful adaptive feature extraction capabilities. Compared with mainstream PND algorithms, this algorithm achieves a CPM score as high as 0.885, demonstrating significant superiority. The ablation experiments have confirmed the contribution of OD3D to performance improvement and the advantages of the final configuration.

We also explored the future direction of PND algorithms. Given the limited references in the field of pulmonary nodule detection based on 3D point clouds, we collaborated with a team of physicians to propose an application roadmap that covers 3D reconstruction, dataset creation and annotation, as well as algorithm design. The feasibility of this approach was validated through machine learning algorithms. And we hope to inspire further groundbreaking research in this area.

CREDIT Author Statement:

Yun Tie: Writing - original draft, Investigation, Formal analysis, Methodology, Software. **Ying Wang**: Conceptualization, Formal analysis, Validation, Supervision, Methodology, Visualization, Writing - review & editing. **Dalong Zhang**: Conceptualization, Formal analysis, Validation, Writing - review & editing, Project administration, Supervision. **Zepeng Zhang**: Conceptualization, Formal analysis, Investigation, Resources, Validation, Visualization, Writing - review & editing. **Fenghui Liu**: Conceptualization, Formal analysis, Investigation, Resources, Visualization, Writing - review & editing, Data Curation. **Lin Qi**: Conceptualization, Formal analysis, Validation, Supervision, Writing - review & editing.

Compliance with ethics requirements

This article does not contain any studies with human or animal subjects.

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.

References

[1] Ding J, Li A, Hu Z, Wang L. Accurate pulmonary nodule detection in computed tomography images using deep convolutional neural networks. In: Medical Image Computing and Computer Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11–13, 2017, Proceedings, Part III 20, Springer; 2017. p. 559–67.

[2] Dou Q, Chen H, Yu L, Qin J, Heng PA. Multilevel contextual 3-d cnns for false positive reduction in pulmonary nodule detection. IEEE Trans Biomed Eng 2016;64:1558–67.

[3] Geiger A, Lenzp UR. Arewereadyfor autonomousdriving; 2012.

[4] George J, Skaria S, Varun V, et al. Using yolo based deep learning network for real time detection and localization of lung nodules from low dose ct scans. In: Medical Imaging 2018: Computer-Aided Diagnosis, SPIE; 2018. p. 347–55.

[5] Group CLO, et al. China national lung cancer screening guideline with low-dose computed tomography (2023 version). Chin J Lung Cancer 26, 1; 2023.

[6] Guan Tianmin, Liu Guangmeng XL. Medical image segmentation and reconstruction based on itk and vtk. J Dalian Jiaotong Univ 40; 2019.

[7] Han Yu-fen, Honggang Qi LW, et al. Pulmonary nodules detection assistant platform: An effective computer aided system for early pulmonary nodules detection in physical examination. Comput Methods Prog Biomed 2022;217:106680. URL: https://api.semanticscholar.org/CorpusID:246732004.

[8] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition, in. In: Proceedings of the IEEE conference on computer vision and pattern recognition. p. 770–8.

[9] Huang X, Sun W, Tseng TLB, Li C, Qian W. Fast and fully-automated detection and segmentation of pulmonary nodules in thoracic ct scans using deep convolutional neural networks. Computer Med Imag Graph 2019;74:25–36.

[10] Shi Jufang, Le Wang NW, et al. Clinical characteristics and medical service utilization of lung cancer in china, 2005–2014: overall design and results from a multicenter retrospective epidemiologic survey. Lung cancer 2019;128:91–100.

[11] Li, C., Zhou, A., Yao, A., 2022. Omni-dimensional dynamic convolution. arXiv preprint arXiv:2209.07947.

[12] Li Y, Fan Y. Deepseed: 3d squeeze-and-excitation encoder-decoder convolutional neural networks for pulmonary nodule detection. In: 2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI). p. 1866–9. URL: https://api.semanticscholar.org/CorpusID:214176487.

[13] Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2017. Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, pp. 2980–2988.

[14] Mei J, Cheng MM, Xu G, Wan LR, Zhang H. Sanet: A slice-aware network for pulmonary nodule detection. IEEE Trans Pattern Anal Mach Intell 2021;44:4374–87.

[15] Mei J, Cheng MM, Xu G, et al. Sanet: A slice-aware network for pulmonary nodule detection. IEEE Trans Pattern Anal Mach Intell 2022;44:4374–87. doi: https://doi.org/10.1109/TPAMI.2021.3065086.

[16] Qinghua ZHOU, Yaguang FAN, Y.W., et al., 2016. China national guideline of classification, diagnosis and treatment for lung nodules (2016 version). Chinese Journal of Lung Cancer 19, 793.

[17] Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28.

[18] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation, in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18, Springer; 2015. p. 234–41.

[19] Rusu RB, Cousins S. 3d is here: Point cloud library (pcl). In: 2011 IEEE international conference on robotics and automation. IEEE; 2011. p. 1–4.

[20] Sager C, Zschech P, Kühl N. labelcloud: A lightweight domain-independent labeling tool for 3d object detection in point clouds. ArXiv abs/2103.04970 2021.

[21] Setio AAA, Ciompi F, Litjens G, Gerke P, Jacobs C, Van Riel SJ, Wille MMW, Naqibullah M, Sánchez CI, Van Ginneken B. Pulmonary nodule detection in ct images: false positive reduction using multi-view convolutional networks. IEEE Trans Med Imag 2016;35:1160–9.

[22] Setio AAA, Traverso A, De Bel T, Berens MS, Van Den Bogaard C, Cerello P, Chen H, Dou Q, Fantacci ME, Geurts B, et al. Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: the luna16 challenge. Medical image analysis 2017;42:1–13.

[23] Tang, H., Zhang, C., Xie, X., 2019. Nodulenet: Decoupled false positive reductionfor pulmonary nodule detection and segmentation. CoRR abs/1907.11320. URL: http://arxiv.org/abs/1907.11320, arXiv:1907.11320.

[24] Wu R, Liang C, et al. Multi-kernel driven 3d convolutional neural network for automated detection of lung nodules in chest ct scans. Biomed Opt Exp 2023;15:1195–218. URL: https://api.semanticscholar.org/CorpusID:266681962.

[25] Xu R, Liu Z, Luo Y, et al. Sgda: Towards 3-d universal pulmonary nodule detection via slice grouped domain attention. IEEE/ACM Transactions on Computational Biology and Bioinformatics 2023;21:1093–105. URL: https://api.semanticscholar.org/CorpusID:257378618.

[26] Huang Yao-Sian, Ping-Ru Chou HMC, et al. One-stage pulmonary nodule detection using 3-d dcnn with feature fusion and attention mechanism in ct image. Computer methods and programs in biomedicine 2022;220:106786. URL: https://api.semanticscholar.org/CorpusID:247964493.

[27] Zhang G, Zhang H, Yao Y, Shen Q. Attention-guided feature extraction and multiscale feature fusion 3d resnet for automated pulmonary nodule detection. IEEE Access 2022;10:61530–43. URL: https://api.semanticscholar.org/CorpusID:249590422.

[28] Zhu W, Liu C, Fan W, Xie X. Deeplung: Deep 3d dual path nets for automated pulmonary nodule detection and classification. In: 2018 IEEE winter conference on applications of computer vision (WACV). IEEE; 2018. p. 673–81.

[29] Zhu W, Liu C, Fan W, Xie X. Deeplung: Deep 3d dual path nets for automated pulmonary nodule detection and classification. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). p. 673–81. URL: https://api.semanticscholar.org/CorpusID:21120549.

[30] Zhu X, Wang X, Shi Y, Ren S, Wang W. Channel-wise attention mechanism in the 3d convolutional network for lung nodule detection. Electronics 2022;11. URL: https://www.mdpi.com/2079-9292/11/10/1600.