

Article

nmPLS-Net: Segmenting Pulmonary Lobes Using nmODE

Peizhi Dong, Hao Niu, Zhang Yi  and Xiuyuan Xu * 

College of Computer Science, Sichuan University, Chengdu 610065, China; dongpeizhi@stu.scu.edu.cn (P.D.); 2018323045038@stu.scu.edu.cn (H.N.); zhangyi@scu.edu.cn (Z.Y.)

* Correspondence: xuxiuyuan@scu.edu.cn

Abstract: Pulmonary lobe segmentation is vital for clinical diagnosis and treatment. Deep neural network-based pulmonary lobe segmentation methods have seen rapid development. However, there are challenges that remain, e.g., pulmonary fissures are always not clear or incomplete, especially in the complex situation of the trilobed right pulmonary, which leads to relatively poor results. To address this issue, this study proposes a novel method, called nmPLS-Net, to segment pulmonary lobes effectively using nmODE. Benefiting from its nonlinear and memory capacity, we construct an encoding network based on nmODE to extract features of the entire lung and dependencies between features. Then, we build a decoding network based on edge segmentation, which segments pulmonary lobes and focuses on effectively detecting pulmonary fissures. The experimental results on two datasets demonstrate that the proposed method achieves accurate pulmonary lobe segmentation.

Keywords: pulmonary lobe segmentation; neural memory ordinary differential equation; multi-task learning

MSC: 68T07



Citation: Dong, P.; Niu, H.; Yi, Z.; Xu, X. nmPLS-Net: Segmenting Pulmonary Lobes Using nmODE. *Mathematics* **2023**, *11*, 4675. <https://doi.org/10.3390/math11224675>

Academic Editor: Debesh Jha

Received: 4 October 2023

Revised: 6 November 2023

Accepted: 14 November 2023

Published: 17 November 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Pulmonary lobe segmentation plays a crucial role in clinical practice, including assisting in diagnosing pulmonary diseases and planning lung surgeries. Relying on manual pulmonary lobe segmentation is time-consuming and labor-intensive. Furthermore, manual segmentation results can be influenced by subjective factors. Thus, an effective automatic method is vital to ensure high efficiency and repeatability.

Although automatic pulmonary lobe segmentation algorithms have made significant progress, there are still several challenges in their practical clinical application. Regarding right pulmonary lobe segmentation, it comprises three lobes and is more complex than the left ones. Specifically, the right middle lobe is sandwiched between two other lobes. Moreover, segmenting pulmonary fissures always exhibit another challenge for unclear or incomplete fissures. The existing methods rarely consider the crucial aspect of learning correlations between features and cannot effectively extract the overall information between samples, which is essential for resolving the abovementioned issue.

To address these issues, this study proposes a novel pulmonary lobe segmentation method using nmODE [1], called nmPLS-Net. Firstly, this method benefits from the robust nonlinear expressive and memory capabilities of nmODE, and the constructed model can extract more complex and rich pulmonary features to support the precise segmentation of pulmonary lobes. Then, due to the significant challenges in identifying pulmonary fissures, nmPLS-Net leverages a fissure-enhanced associative learning approach, focusing on learning detailed features of pulmonary fissures. Finally, this study utilizes two datasets to evaluate the proposed method's effectiveness. The experimental results demonstrate that the proposed method achieves competitive results compared to other state-of-the-art methods. The main contributions of this study can be summarized as follows:

1. The main contribution of this study is the combination of nmODE with convolutional neural networks in the task of pulmonary lobe segmentation. This combination leverages nmODE's high nonlinearity and memory capabilities better to identify global pulmonary features and the relationships between these features;
2. This study employs a novel fissure-enhanced associative learning approach to make the model focus on detail identification near pulmonary fissures;
3. This study conducted experiments on two datasets, and our network achieved competitive overall segmentation results, significantly outperforming previous works in segmenting the right middle pulmonary lobe.

2. Related Works

2.1. Medical Image Processing

Deep learning has found widespread application in medical image processing due to its simplicity and efficiency. Early deep learning image segmentation networks have used fully convolutional networks (FCNs) [2], demonstrating good performance due to their ability to extract hierarchical features and end-to-end processing of inputs of arbitrary sizes. However, an FCN's limitation lies in the fact that it under-utilizes features at different resolutions obtained from different layers, especially in large-resolution medical images. U-Net [3], with its multi-level encoder-decoder architecture and skip connections, addresses the FCN's limitations and has gained popularity in medical image processing. Shoaib et al. [4] used several different state-of-the-art convolutional neural networks to perform left ventricle segmentation, achieving commendable results. Zhao [5] and his team conducted a detailed investigation and analysis of deep learning methods for a neuroimaging diagnosis of Alzheimer's disease.

The introduction of the transformer [6] architecture had a profound impact on the field of deep learning. Based on the transformer, the vision transformer (ViT) has found widespread application in image processing. The swin-transformer [7] proposed by Liu et al. addressed the computational challenges associated with ViT in image processing by incorporating a sliding window and partial attention mechanism. The swin-transformer v2 [8] further improved upon the swin transformer. TransUNet [9], developed by Chen et al., combines the strengths of U-Net and the transformer to process medical images. Shen et al. [10] proposed a boundary-guided transformer that can simultaneously identify rectum and tumor regions in sagittal MR images of rectal cancer patients. For a comprehensive overview of related research, Liu et al. offered a survey [11]. Transformer-based approaches have shown superior performance in many aspects. However, due to its huge parameters, traditional 3D convolutional neural networks are currently more practical in this study.

Pulmonary lobe segmentation typically uses the entire CT scan as input. Three-dimensional U-Net structures are commonly employed for this task, such as V-Net [12]. Based on V-Net, Imran et al. [13] introduced a progressive dense V-Net, enhancing performance through feature fusion. Ferreira et al. [14] proposed FRV-Net, utilizing regularization techniques to achieve better segmentation results, even with limited data. Tang et al. [15] employed a mixed loss function to enhance the model's focus on challenging pixels in pulmonary lobes. While the previously mentioned methods have achieved good results, they often have a large number of parameters. To obtain a more efficient network, Lee et al. [16] introduced an efficient PLS-Net for pulmonary lobe segmentation. They leveraged 3D depth-wise separable convolution to effectively reduce parameters while employing dilated residual modules to enhance the network's receptive field without increasing the parameter count. Despite significantly reducing the model's parameter count and improving the receptive field, PLS-Net exhibits subpar performance at the pulmonary lobe edges and with incomplete pulmonary fissures. To address the issue of incomplete fissures, Xie et al. [17] harnessed relational modeling for pulmonary lobe segmentation, effectively capturing relationships and contextual information among pulmonary lobes. However, this approach comes at the cost of a substantially higher parameter count than PLS-Net, resulting in relatively slower computational speeds due to its intricate process. Fan et al. [18] intro-

duced a learnable interpolation and extrapolation network designed to handle incomplete pulmonary fissures and enhance segmentation performance. Nevertheless, this method necessitates five processing stages, including lung segmentation, pulmonary fissure segmentation, pulmonary fissure completion, pulmonary lobe segmentation, and post-processing, significantly diminishing its efficiency. Liu et al. [19] took the enhancement of PLS-Net a step further with RPLS-Net by incorporating pulmonary lobe boundary segmentation as an auxiliary task, thus focusing on challenging edge pixels. This approach achieved superior results compared to PLS-Net without significantly increasing the parameter count. However, it is worth noting that RPLS-Net cannot learn correlations between features and heavily relies on an expanded receptive field to gather global information. This limitation leads to relatively poorer performance on the right middle pulmonary lobe compared to other pulmonary lobes.

2.2. Neural Ordinary Differential Equations

In recent years, deep learning has made remarkable strides across various applications. One of the most renowned neural networks in the realm of deep learning is Res-Net [20], celebrated for its robustness due to the incorporation of residuals. Research conducted by Haber et al. [21] has revealed that Res-Net can be viewed as a discrete counterpart of certain differential equations. The concept of neural ordinary differential equations (NODEs) was introduced by Chen et al. [22], shifting the paradigm from discrete neural networks to continuous differential equations. The core concept in Res-Net, the residual, can be likened to a discrete differential equation form. Denote the output of the l -th layer in a neural network as x^l , conventionally. In that case, the output of the subsequent layer ($l + 1$) can be represented as $x^{l+1} = f(x^l)$, where f encompasses the computational processes of a single layer, including linear transformations, regularization, and nonlinear activation functions. In the case of Res-Net, the output of the $(l + 1)$ -th layer is defined as $x^{l+1} = x^l + f(x^l) \cdot 1$. If considering l as time step t and interpreting the output x^l as a function $y(t)$ evolving over time steps, with the step interval of 1 being analogous to a time interval Δt , the model's computations can be conceptualized as the equation $y(t + \Delta t) - y(t) = f(y(t)) \cdot \Delta t$. As the number of layers in the model approaches infinity, i.e., as Δt tends to zero, it arrives at the NODE expression, a differential equation illustrated in Equation (1). Obtaining the exact solution to this equation is nearly impossible. In practice, by providing an initial value $y(0)$ to the abovementioned differential equation, an ODE solver library can be leveraged to approximate the solution as $y(T)$.

$$\dot{y}(t) = \lim_{\Delta t \rightarrow 0} \frac{y(t + \Delta t) - y(t)}{\Delta t} = f(y(t)). \quad (1)$$

Compared to traditional neural networks, neural ordinary differential equations exhibit more nonlinearity and fewer parameters. Subsequently, many related studies have emerged. Dupont et al. [23] pointed out that the NODE, when using the data input as the initial value, preserves the spatial structure of the input, and the authors provided examples illustrating the NODE's limitations in this regard. However, this issue can be addressed by inputting the data through external input variables instead of using them as initial values for the differential equation [1]. For more related methods, please refer to Refs. [24,25].

3. Methods

3.1. Problem Formulation

The main task of this study is a multi-class segmentation problem. The input of the constructed model is a whole CT image, denoted by $X \in \mathbb{R}^{1 \times D \times H \times W}$, and the labels are denoted by $G \in \mathbb{R}^{1 \times D \times H \times W}$. D , H , and W denote the depth, height, and width in an input image, respectively. $\mathcal{F}$ denotes the constructed model. $\mathcal{P}$ and $\mathcal{P}_e$ denote the output of the main and fissure-enhanced associative learning task. $\mathcal{L}$ denotes the loss function to optimize the constructed model.

3.2. Model

The architecture of the proposed model in this study was inspired by PLS-Net [16] and RPLS-Net [19]. Figure 1 illustrates the workflow of the proposed method. The reason for choosing PLS-Net as the backbone network for this study is that, compared to commonly used backbone networks for pulmonary lobe segmentation, such as V-Net, PLS-Net has fewer parameters and yields better results. Additionally, using the same backbone network and dataset as in Ref. [19] ensures a fair and accurate comparison of the performance of the nmODE and FEAL modules.

The model inputs the original CT image X and then produces pulmonary lobe result P and fissure-enhanced associative learning result P_e . This process can be represented as $P, P_e = \mathcal{F}(X)$.

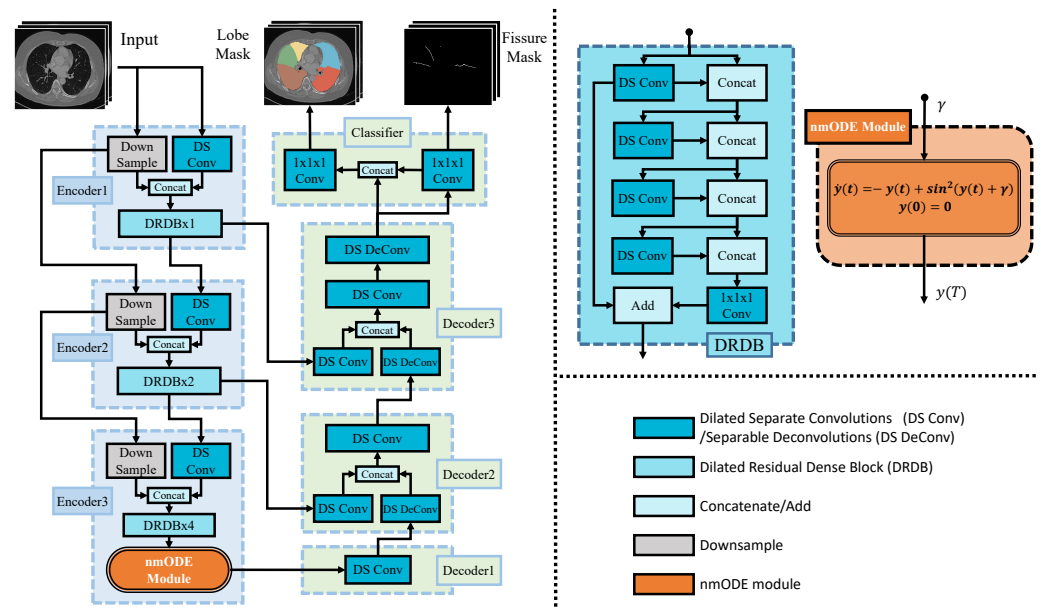


Figure 1. The detailed structure of nmPLS-Net, the modules of the backbone network encoder, and the detailed structure of the nmODE module are displayed on the right. The color coding for each module is as shown in the figure. Within the DRDB module, DRDB $\times 1$, DRDB $\times 2$, and DRDB $\times 4$ represent one, two, and four consecutive DRDB modules, respectively. Encoder1, Encoder2, and Encoder3 in the figure are denoted as $E1$, $E2$, and $E3$ in this paper. Similarly, Decoder1, Decoder2, and Decoder3 are denoted as $D1$, $D2$, and $D3$.

The model's encoder consists of three hierarchical stages, denoted as $E1$, $E2$, and $E3$. Similarly, the decoder comprises three corresponding stages, namely $D1$, $D2$, and $D3$, and a classifier for generating results. Skip connections connect the same stage.

The model's encoder comprises Dilated Separate Convolutions (DS Conv) and Dilated Residual Dense Block (DRDB). In the case of dilated separate convolution, its principle is similar to regular depth-wise separable convolution [26]. Additionally, dilated separate convolution employs dilated convolutions [27] to increase the receptive field, thus enlarging the receptive field without increasing the parameter count. The Dilated Residual Dense Block, written as DRDB for simplicity, used in the model consists of dilated separate convolutions and their structure, as shown in Figure 1. This module combines the results of convolutions with different receptive fields in a residual manner to achieve a wide range of receptive field combinations. In each encoder stage, the dilation rate for the dilated separate convolution is set to 1, indicating no dilation, and the convolution stride is set to 2 for down-sampling. In the DRDB module, the dilation rates for the four dilated separate convolutions are set from top to bottom as (1, 2, 3, 4), as recommended in [16]. The input for each encoder stage is concatenated with the original image down-sampled to the corresponding size to compensate for the information loss during down-sampling. In

stage *E3*, the nmODE module is introduced to process the features extracted by its DRDB module. The input to the nmODE module is the output feature of the DRDB module, and the output is the solution of the differential equation in the nmODE module. The following section will provide a detailed description of the nmODE module.

The model's decoder consists of Dilated Separate Convolutions (DS Conv) and Depth-wise Separable Deconvolutions (DS DeConv). The structure of dilated separate convolution is the same as that used in the encoder, with a dilation rate of 1. Depth-wise separable deconvolution employs deconvolution to perform depth-wise separable convolution for up-sampling. The outputs from the corresponding layers of the encoder, transmitted through skip connections, are used to supplement the missing feature information after up-sampling. The classifier comprises two $1 \times 1 \times 1$ convolution modules responsible for generating segmentation results from model outputs. Within the classifier, a fissure-reinforcement method is introduced, with further information available in Section 3.4.

3.3. Neural Memory Ordinary Differential Equation (nmODE)

Neural differential equations exhibit a higher degree of nonlinearity and better robustness, and the architecture of the NODEs is depicted in Figure 2a. However, regular NODE has some limitations, such as their inability to represent mappings such as $g(1) = -1, g(-1) = 1$. This limitation primarily arises since NODE treats data as initial values, which preserves the spatial structure of the input data [23]. Furthermore, differential equations can describe dynamical systems, and research suggests that attractors within dynamical systems are considered to be associated with memory capacity [28,29]. However, traditional neural differential equations often fail to harness the memory capacity offered by attractors.

Yi [1] proposed a new type of neural differential equation, called the neural memory differential equation (nmODE), which not only addresses the inherent limitations of traditional differential equations but also fully harnesses the memory capacity offered by dynamical systems. The nmODE treats the input data as external parameters rather than utilizing them as the initial conditions for the ordinary differential equation while employing a fixed initial value. By doing so, nmODE not only avoids the problem mentioned above but also separates the functionality of neurons into learning and memory components. Learning only occurs in the learning part, while the memory part is responsible for mapping the input to its global attractor, establishing a mapping from the input space to the memory space [1]. The structure of nmODE is illustrated in Figure 2b, where γ represents external input, and the learning part occurs within the transformation $g(x^l)$. In our model, γ represents the output of the DRDB module within the encoder stage, labeled as *E3*. The formula representation of the nmODE architecture is shown in Equation (2). Based on the nmODE architecture, Yi [1] also proposed a novel and efficient implementation, which is structured as shown in Equation (3). Furthermore, Yi demonstrated in Ref. [1] that a key property of this implementation is that it has a unique global attractor for each input, defining a mapping that transforms an external input x into an output $y(T)$ within the memory space.

$$\begin{cases} \gamma = g(x^l), \\ \dot{y}(t) = f(y(t), t, \gamma). \end{cases} \quad (2)$$

$$\dot{y}(t) = -y(t) + \sin^2(y(t) + \gamma). \quad (3)$$

In pulmonary lobe segmentation, the right lung with three lobes is more challenging than the left lung, and cases with unclear or incomplete boundaries between pulmonary lobes can make segmentation even more difficult. Human annotators rely on the overall pulmonary structure and dependencies between objects when performing manual segmentation. Similarly, learning better global features and capturing dependencies between features can help address these challenges for neural network models.

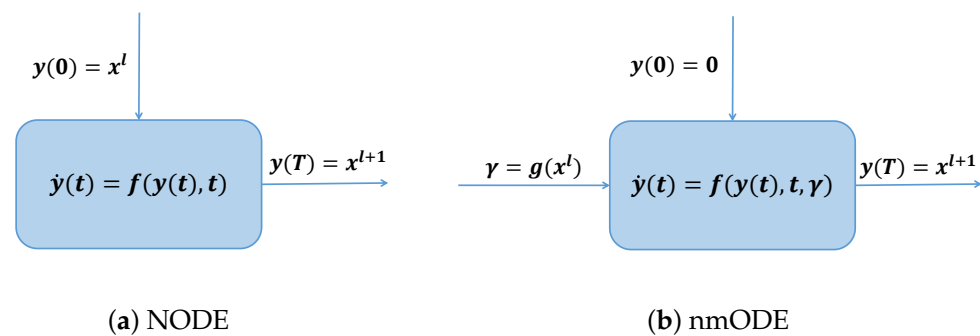


Figure 2. Architecture of NODE and nmODE. (a) represents the typical NODE architecture, where the initial value of the differential equation, denoted as $y(0)$, is derived from the data itself or the output of the previous layer, denoted as x^l . The output is the numerical solution of the differential equation, denoted as x^{l+1} . (b) represents the nmODE Architecture, where the initial value of the differential equation, denoted as $y(0)$, is a fixed value of 0. Data or the output of the previous layer serves as an external input γ into the module. The output is the same as in typical NODE, which is the numerical solution of the differential equation.

In this study, an nmODE module was embedded in the encoder stage $E3$ to further refine the features extracted by the encoder. The powerful nonlinear expressive capability of nmODE enables effective learning of feature representations in complex conditions. Embedding the nmODE module in PLS-Net leverages nmODE's memory capacity and strong nonlinear capabilities to obtain better global features and learn the relationships between features. Across different samples, similar features are mapped to nearby attractors in the nmODE memory space, allowing nmODE to memorize similar information between samples and enabling the model to capture overall pulmonary characteristics. The nmODE module can simultaneously learn and remember dependencies between features during the mapping process. For individual samples, nmODE's high nonlinearity and memory capabilities refine and fuse information from different channels of features more effectively, resulting in more global feature information. The primary reasons for placing the nmODE module in the last stage of the encoder are two-fold. Firstly, the lowest-level stage boasts the largest channel dimension and the most expansive receptive field. A higher channel dimension facilitates a more comprehensive exploration of profound connections among features. Moreover, a larger receptive field diminishes interference from superfluous fine-grained details and steers the input features toward a more globally contextualized state within the nmODE module. Second, it is for performance considerations. The lowest-level output has the lowest resolution, and placing the nmODE module here does not significantly increase memory usage or reduce model computation speed during training.

3.4. Fissure-Enhanced Associative Learning

Previous work [19] proposed an auxiliary task involving segmenting the entire pulmonary lobe edge. Complete pulmonary segmentation is relatively straightforward, and our observations during training have shown that the network tends to distinguish the outer pulmonary contour first naturally. So, emphasizing the model's attention on external boundaries is unnecessary. Instead, this study aims to direct the model's focus toward the information within the pulmonary fissures, which are more challenging areas.

This study introduces a novel approach called Fissure-Enhanced Associative Learning (FEAL) to encourage the model to focus on voxels near pulmonary fissures and integrate pulmonary fissure information during classification. The associated loss is computed using the pulmonary fissure segmentation results as a secondary model output. This integration enables the model to identify voxels corresponding to pulmonary fissure locations accurately. Within the classifier, the FEAL module merges the pulmonary fissure segmentation output with the decoder's output, employing a convolutional layer to combine the fissure information with the decoder's output features and make corresponding predictions. Pre-

viously, in multitask learning approaches for pulmonary lobe segmentation, the model primarily relied on auxiliary tasks to encourage the decoder to prioritize pulmonary fissures. However, during the final classification step, the classifier independently handled pulmonary fissure segmentation. In contrast, our fissure-enhanced associative learning approach incorporates classification information during the classification process, further associating decoder features with pulmonary fissure information.

A simple and efficient algorithm can be used to automatically generate pulmonary fissure labels from the label mask of CT images without any manual intervention. Thanks to the fact that the values in CT label masks are limited to a few categories (e.g., a pulmonary lobe segmentation mask may have only 6 classes, with values ranging from 0 to 5 as integers), we determine whether a point belongs to the pulmonary fissure by sliding a window and verifying whether the values inside the window belong to two neighbor target categories simultaneously. The process is outlined in pseudocode in Algorithm 1. The pulmonary fissure segmentation results obtained by the algorithm are shown in Figure 3. This algorithm only needs to scan the image once and has a time complexity of $O(n)$, assuming the input size is n . Compared to the methods used in previous work, which involved image processing with the scikit-image library, Gaussian filtering, and thresholding, this algorithm is simpler and more efficient. Additionally, it can mitigate the adverse effects of partial mislabeling. For example, in Figure 3, a small portion of the right upper lobe is incorrectly labeled as the left lower lobe, which is ignored in the generated pulmonary fissure mask.

Algorithm 1 Pulmonary Fissure Label Generation Algorithm

```

1: neighbours  $\leftarrow [(1,2), (1,3), (2,3), (4,5)]$ 
2: img_array  $\leftarrow$  CT image mask of shape  $[D, W, H]$ 
3: result  $\leftarrow$  metric with all False values of shape  $[D, W, H]$ 
4: for  $i \leftarrow 1$  to  $D$ ,  $j \leftarrow 2$  to  $W - 1$ ,  $k \leftarrow 2$  to  $H - 1$  do
5:   window  $\leftarrow$  img_array  $[i, j - 1 : j + 1, k - 1 : k + 1]$ 
6:   for pair  $\in$  neighbors do
7:     if pair[1]  $\in$  window and pair[2]  $\in$  window then
8:       result  $[i, j, k] \leftarrow$  True
9:     end if
10:  end for
11: end for
12: Output: Pulmonary fissure labels.

```

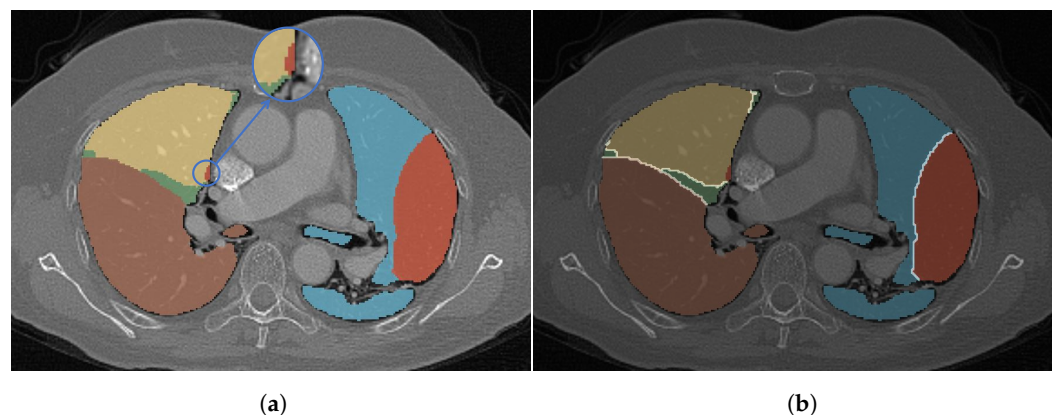


Figure 3. (a) Lung CT image and its mask, with an annotation error circled in blue. (b) Pulmonary fissures separated by Algorithm 1. In this visual representation, distinct lung regions are color-coded: yellow for the right upper lobe, green for the right middle lobe, brown for the right lower lobe, blue for the left upper lobe, red for the left lower lobe, and white for pulmonary fissures.

3.5. Loss Functions

The loss function used for training is a mixed loss function consisting of two components: $\mathcal{L}_{main}$ for calculating pulmonary lobe segmentation loss and $\mathcal{L}_e$ for calculating the FEAL method loss. A hybrid loss function consisting of dice loss $\mathcal{L}_{Dice}$ and focal loss $\mathcal{L}_{Focal}$ is used as the primary loss $\mathcal{L}_{main}$. The calculation formula for the $\mathcal{L}_{Dice}$ in the hybrid loss is as in Equation (4). In this equation, C represents the number of pulmonary lobe categories, and N represents the number of points in a single sample. g_c^i and p_c^i , respectively, denote the ground truth and the probability of point i being predicted as category c . ϵ is the smoothing coefficient; in this study, the value is set to 1. The dice loss is calculated separately for each category. However, there is a significant difference in the number of foreground and background pixels for each category. Therefore, the focal loss is introduced to balance the positive and negative samples. The formula for the $\mathcal{L}_{Focal}$ used in the hybrid loss is as in Equation (5). In this equation, C , N , g_c^i , p_c^i are the same as in Equation (4). α_c represents the weight for each category, and β is a hyperparameter. $\mathcal{L}_{main}$ is as shown in Equation (6), where λ_1 and λ_2 are the weight coefficients for the two types of losses.

$$\begin{cases} I_c = \sum_{i=1}^N g_c^i * p_c^i, \\ U_c = \sum_{i=1}^N (g_c^i + p_c^i), \\ \mathcal{L}_{Dice}(\mathcal{P}, G) = \sum_{c=1}^C 1 - \frac{2I_c + \epsilon}{U_c + \epsilon}. \end{cases} \quad (4)$$

$$\begin{aligned} \mathcal{L}_{Focal}(\mathcal{P}, G) = & - \sum_{c=1}^C \alpha_c \sum_{i=1}^N (g_c^i (1 - p_c^i)^\beta \log p_c^i \\ & + (1 - g_c^i) (p_c^i)^\beta \log (1 - p_c^i)). \end{aligned} \quad (5)$$

$$\mathcal{L}_{main}(\mathcal{P}, G) = \lambda_1 \mathcal{L}_{Dice}(\mathcal{P}, G) + \lambda_2 \mathcal{L}_{Focal}(\mathcal{P}, G). \quad (6)$$

Since positive samples account for only a small portion of the pulmonary fissure labels, leading to severe class imbalance, the focal loss is used as the loss function for the fissure-enhanced associative learning output $\mathcal{L}_e$ to address this issue. The formula is as in Equation (7). In this context, $\mathcal{P}_e$ and G_e represent the predictions and ground truth for pulmonary fissure, while p_e^i and g_e^i represent the predictions and ground truth for each pixel. α_e and β_e are hyperparameters.

$$\begin{aligned} \mathcal{L}_e(\mathcal{P}_e, G_e) = & \sum_{i=1}^N \alpha_e (g_e^i (1 - p_e^i)^{\beta_e} \log p_e^i \\ & + (1 - g_e^i) (p_e^i)^{\beta_e} \log (1 - p_e^i)). \end{aligned} \quad (7)$$

The overall loss function is as in Equation (8), where λ represents the weight of the fissure-enhanced associative learning loss function.

$$\mathcal{L}(\mathcal{P}, G, \mathcal{P}_e, G_e) = \mathcal{L}_{main}(\mathcal{P}, G) + \lambda \mathcal{L}_e(\mathcal{P}_e, G_e). \quad (8)$$

4. Experiments

4.1. Experimental Settings

Dataset: A base dataset consistent with Ref. [19] was used in this study, which comprises 32 chest CT scan images with a maximum slice thickness of 1 mm. The pulmonary fissure labels used for fissure-enhanced associative learning were generated from pulmonary lobe labels using the fissure label generation algorithm. Experiments were conducted on another publicly available LUNA16 dataset, as presented in Ref. [15], which

contains 50 cases with unclear or incomplete pulmonary fissures and lesions to validate the effectiveness of our method. All CT data were resized to a (1 mm, 1 mm, 1 mm) spacing and filtered to include voxels with Hounsfield Unit (HU) values between 400 and 1000. The data were then normalized. A 20-voxel-wide region of uninformative background near each CT image's boundary was cropped to improve training efficiency. For example, a CT of size (300, 512, 512) was cropped to (300, 472, 472) after this process.

Model and Training: Our model and training code were implemented using the PyTorch library [30]. The code of the model can be found at <https://github.com/EdewagaPoe/nmPLS-Net-Segmenting-Pulmonary-Lobes-using-nmODE.git> accessed on 4 November 2023. We employed PyTorch's built-in automatic mixed precision training (AMP) to save GPU memory during training. Due to the non-uniform input sizes, the batch size was fixed at 1. The optimizer used was the Adam optimizer [31] with a learning rate of 0.001 and a weight decay of 5×10^{-4} . The dilated residual convolution module of the model used dilation rates and growth rates of (1, 2, 3, 4) and 12, respectively, following the recommendations in Ref. [16]. The hyperparameters in the overall loss function, namely λ_1 , λ_2 , and λ , were set to 0.6, 0.4, and 0.2, respectively. α_c and α_e were both set to 1, and β and β_e were set to 2. The training was performed on an NVIDIA TITAN RTX GPU with 24 GB of VRAM. We trained the model for 100 epochs and selected the best-performing result on the validation set for testing.

Evaluation Metrics: This study used the dice coefficient (DC) as an evaluation metric for segmentation performance, as shown in Equation (9). Here, $\mathcal{P}_{val}$ represents the output processed through the argmax function, $\|\cdot\|_1$ denotes the L1 Norm, N is the total number of voxels in the output, p_{val}^i is the predicted value for voxel i , and g^i is the corresponding ground truth. The Jaccard index is also used as an evaluation metric for segmentation performance, as shown in Equation (10).

$$\begin{cases} \mathcal{P}_{val} = \operatorname{argmax}(\mathcal{P}), \\ \|\mathcal{P}_{val} \cap G\|_1 = \sum_{i=1}^N |p_{val}^i * g^i|, \\ DC(\mathcal{P}_{val}, G) = 2 \frac{\|\mathcal{P}_{val} \cap G\|_1}{\|\mathcal{P}_{val}\|_1 + \|G\|_1}. \end{cases} \quad (9)$$

$$\begin{cases} \mathcal{P}_{val} = \operatorname{argmax}(\mathcal{P}), \\ \|\mathcal{P}_{val} \cap G\|_1 = \sum_{i=1}^N |p_{val}^i * g^i|, \\ \|\mathcal{P}_{val} \cup G\|_1 = \sum_{i=1}^N |p_{val}^i + g^i|, \\ Jaccard(\mathcal{P}_{val}, G) = \frac{\|\mathcal{P}_{val} \cap G\|_1}{\|\mathcal{P}_{val} \cup G\|_1}. \end{cases} \quad (10)$$

The average symmetric surface distance (ASSD) is used as another evaluation metric to measure the segmentation accuracy of pulmonary lobe boundaries. The formula for ASSD is expressed as in Equation (11). In this equation, v represents any voxel, S represents the set of surface voxels, $\|\cdot\|_2$ denotes the L2 Norm, $S(\mathcal{P}_{val})$ and $S(G)$ represent the sets of surface voxels for $\mathcal{P}_{val}$ and G , respectively, and $|S|$ indicates the number of elements in a set.

$$\begin{cases} D(v, S) = \min_{s \in S} \|v - s\|_2, \\ ASSD = \frac{\sum_{p \in S(\mathcal{P}_{val})} D(p, S(G)) + \sum_{g \in S(G)} D(g, S(\mathcal{P}_{val}))}{|S(\mathcal{P}_{val})| + |S(G)|}. \end{cases} \quad (11)$$

The Precision and Recall metrics are also used to measure the model's performance. The formula for precision and recall is shown in Equations (12) and (13), where TP represents the number of true positives, FP represents the number of false positives, and FN represents the number of false negatives.

$$Precision = \frac{TP}{TP + FP} \quad (12)$$

$$Recall = \frac{TP}{TP + FN} \quad (13)$$

4.2. Quantitative Experiment Results

This study conducted experiments on the base dataset using three networks: PLS-Net, RPLS-Net, and nmPLS-Net. The specific experimental results are shown in Table 1a. In this table and the subsequent tables, UR, MR, LR, UL, and LL represent the metric of upper right lobe, middle right lobe, lower right lobe, upper left lobe, and lower left lobe, respectively. The average dice coefficient achieved by nmPLS-Net is 0.9570, the average ASSD is 1.1133, the average precision is 0.9578, the average recall is 0.9615, and the average Jaccard index is 0.9185, surpassing the compared networks. From the experimental results, it can be seen that nmPLS-Net exhibits a significant improvement in overall segmentation performance, especially with a substantial enhancement in the segmentation of the right middle pulmonary lobe. At the same time, nmPLS-Net has also achieved the optimal ASSD value, which means that the network's segmentation results along the pulmonary lobe edges are more precise. This performance improvement can be attributed to two key factors: Firstly, the nmODE module's increased memory capacity and high nonlinearity enhance the network's capability to capture global features and learn dependencies between features. This enables the model to predict results from a more comprehensive and interrelated perspective. Secondly, the fissure-enhanced associative learning approach concentrates the model's training on voxels near pulmonary fissures and associates them with the fissure information, resulting in an enhanced segmentation performance for challenging fissure voxels. We also trained and validated on another publicly available dataset in [15]. As shown in Table 1b, the results are consistent with those on the base dataset. Our network still achieves the best performance in both overall segmentation and the segmentation of the right middle lobe, demonstrating the effectiveness and reliability of our approach.

Furthermore, this study conducted training on the training set of the base dataset and subsequently validated the model using 20 randomly selected CT scans from the LUNA16 dataset, as presented in Table 2. Notably, nmPLS-Net continues to exhibit superior performance in the validation results, particularly in the right lung. While the overall performance decreases compared to directly training on the LUNA16 dataset, several factors may contribute to this phenomenon. Firstly, our training dataset comprises only 20 CT scans, in contrast to the 40 CT scans in the training set of the LUNA16 dataset, potentially impacting the model's generalization ability. Secondly, inconsistencies in data sources and the subjective nature of manual annotations have led to disparities in data distribution between the training and validation sets, contributing to the observed performance drop. However, it is worth highlighting that our model demonstrates resilient performance under these circumstances, especially in segmenting the right middle lobe. This result underscores the enhancing effect of nmPLS-Net's memory capabilities and the memorized features on the model's generalization and robustness.

Table 1. Comparison of results on different datasets. UR, MR, LR, UL, and LL represent the metric of upper right lobe, middle right lobe, lower right lobe, upper left lobe, and lower left lobe, respectively. The best results in subtables are highlighted with red, and the second best results are highlighted with blue.

(a) Comparison between the target method and previous state-of-art work training on the base dataset.							
Model	Metric	RU	RM	RL	LU	LL	AVG
PLS-Net	DC	0.9309 ± 0.0709	0.8690 ± 0.1182	0.9511 ± 0.0348	0.9442 ± 0.0528	0.9441 ± 0.0425	0.9279 ± 0.5470
	ASSD	3.1171 ± 0.9880	1.7678 ± 0.6075	1.5129 ± 0.5662	2.0548 ± 2.070	1.7204 ± 1.2939	2.0346 ± 0.7194
	Precision	0.9388 ± 0.0410	0.8704 ± 0.0896	0.9412 ± 0.0502	0.9738 ± 0.0171	0.9208 ± 0.0912	0.9290 ± 0.0221
	Recall	0.9555 ± 0.0440	0.9078 ± 0.0347	0.9173 ± 0.0912	0.9745 ± 0.0136	0.9404 ± 0.0625	0.9391 ± 0.0341
	Jaccard Index	0.8717 ± 0.1200	0.7683 ± 0.1703	0.9058 ± 0.0772	0.8943 ± 0.1133	0.8943 ± 0.0907	0.8668 ± 0.5675
RPLS-Net	DC	0.9562 ± 0.0401	0.8731 ± 0.1590	0.9664 ± 0.0110	0.9601 ± 0.0429	0.9546 ± 0.0356	0.9421 ± 0.0850
	ASSD	1.1829 ± 0.8243	2.2384 ± 2.1269	0.8308 ± 0.2972	1.0739 ± 1.1420	1.4468 ± 1.6806	1.3546 ± 1.4554
	Precision	0.9161 ± 0.0250	0.8904 ± 0.0563	0.9508 ± 0.0316	0.9551 ± 0.0207	0.9558 ± 0.0277	0.9417 ± 0.0146
	Recall	0.9698 ± 0.0133	0.9342 ± 0.0249	0.9665 ± 0.0130	0.9502 ± 0.0836	0.9515 ± 0.0197	0.9544 ± 0.0152
	Jaccard Index	0.9161 ± 0.0423	0.7748 ± 0.1876	0.9350 ± 0.0131	0.9233 ± 0.0542	0.9131 ± 0.0411	0.8925 ± 0.1246
nmPLS-Net	DC	0.9640 ± 0.0127	0.9336 ± 0.0217	0.9620 ± 0.0198	0.9660 ± 0.0200	0.9596 ± 0.0223	0.9570 ± 0.0100
	ASSD	1.0263 ± 0.3399	1.4077 ± 0.3747	0.9050 ± 0.1429	0.9642 ± 0.6733	1.2633 ± 1.0183	1.1133 ± 0.3381
	Precision	0.9636 ± 0.0363	0.9342 ± 0.0443	0.9705 ± 0.0248	0.9632 ± 0.0334	0.9572 ± 0.0495	0.9578 ± 0.0181
	Recall	0.9640 ± 0.0134	0.9583 ± 0.0188	0.9601 ± 0.0142	0.9634 ± 0.0435	0.9618 ± 0.0183	0.9615 ± 0.0095
	Jaccard Index	0.9300 ± 0.0319	0.8781 ± 0.0337	0.9330 ± 0.029	0.9292 ± 0.0523	0.9223 ± 0.0517	0.9185 ± 0.0236
(b) Comparison between the target method and previous state-of-art work training on the LUNA16 dataset.							
Model	Metric	RU	RM	RL	LU	LL	AVG
PLS-Net	DC	0.9165 ± 0.0662	0.7960 ± 0.1501	0.9266 ± 0.0553	0.9442 ± 0.0390	0.9366 ± 0.0392	0.9040 ± 0.0538
	ASSD	3.2066 ± 1.3894	5.4324 ± 2.5495	2.5325 ± 1.2278	3.0186 ± 1.858	2.7501 ± 0.9402	3.3880 ± 1.0641
	Precision	0.9326 ± 0.0642	0.7760 ± 0.1971	0.9455 ± 0.0219	0.9498 ± 0.0385	0.9367 ± 0.0455	0.9081 ± 0.0538
	Recall	0.9185 ± 0.0512	0.7847 ± 0.0800	0.9500 ± 0.0364	0.9540 ± 0.0161	0.9434 ± 0.0051	0.9101 ± 0.0152
	Jaccard Index	0.8454 ± 0.1120	0.6616 ± 0.1799	0.8632 ± 0.0694	0.8943 ± 0.0703	0.8809 ± 0.0753	0.8291 ± 0.0985
RPLS-Net	DC	0.9314 ± 0.0582	0.8291 ± 0.1703	0.9407 ± 0.0517	0.9548 ± 0.0239	0.9465 ± 0.0248	0.9205 ± 0.0595
	ASSD	2.5011 ± 1.4062	3.5855 ± 1.5051	1.5885 ± 0.4354	1.6182 ± 1.3881	1.5146 ± 1.0935	2.1616 ± 0.8422
	Precision	0.9130 ± 0.0734	0.8324 ± 0.1594	0.9538 ± 0.0212	0.9564 ± 0.0441	0.9519 ± 0.0231	0.9215 ± 0.0458
	Recall	0.9386 ± 0.0223	0.8378 ± 0.1479	0.9470 ± 0.0355	0.9651 ± 0.0130	0.9601 ± 0.0171	0.9297 ± 0.0358
	Jaccard Index	0.8716 ± 0.1100	0.7081 ± 0.1944	0.8880 ± 0.0899	0.9135 ± 0.0694	0.8984 ± 0.0657	0.8559 ± 0.0899
nmPLS-Net	DC	0.9307 ± 0.0550	0.8499 ± 0.1049	0.9523 ± 0.0200	0.9605 ± 0.0144	0.9558 ± 0.0128	0.9298 ± 0.0370
	ASSD	1.7161 ± 0.5950	2.9336 ± 1.5176	1.4748 ± 0.8940	1.2606 ± 1.0253	1.1010 ± 0.3341	1.6972 ± 0.5353
	Precision	0.9011 ± 0.0961	0.8817 ± 0.0992	0.9557 ± 0.0248	0.9570 ± 0.0411	0.9528 ± 0.0416	0.9296 ± 0.0335
	Recall	0.9366 ± 0.0277	0.8876 ± 0.0642	0.9489 ± 0.0476	0.9478 ± 0.0162	0.9739 ± 0.0057	0.9390 ± 0.0341
	Jaccard Index	0.8705 ± 0.1017	0.7390 ± 0.1403	0.9089 ± 0.0716	0.9241 ± 0.0553	0.9151 ± 0.0527	0.8715 ± 0.0775

4.3. Ablation Experiment

Ablation experiments were conducted to analyze the contributions of various components in the methodology for performance improvement. The results of the ablation experiments are shown in Table 3. In cases where the nmODE module and fissure-enhanced associative learning were not used, the model reverted to PLS-Net, so separate experiments for this scenario were not conducted again. It can be observed that both the nmODE module and fissure-enhanced associative learning contribute to the segmentation performance of the model. The nmODE module, in particular, exhibits the most significant improvement in segmentation performance, with the performance boost primarily evident in the right middle lobe.

Although adding fissure-enhanced associative learning on top of the nmODE module does not substantially improve overall segmentation performance, the enhancement in the segmentation performance of the right middle lobe is notable, indicating that fissure-enhanced associative learning does have an effect, aligning with the purpose of their incorporation into our methodology.

Table 2. Comparison between the target method and previous work training on the base dataset and testing on 20 LUNA16 dataset’s CT volumes. The best results in the table are highlighted with red, and the second best results are highlighted with blue.

Model	Metric	RU	RM	RL	LU	LL	AVG
PLS-Net	DC	0.8656 ± 0.1198	0.7832 ± 0.1688	0.9001 ± 0.0731	0.9032 ± 0.0808	0.8807 ± 0.1097	0.8666 ± 0.0833
	ASSD	5.1146 ± 3.3692	5.7794 ± 2.4364	2.8784 ± 0.7087	5.0594 ± 4.5588	4.4321 ± 3.3269	4.6528 ± 1.7337
	Precision	0.8693 ± 0.0211	0.7258 ± 0.0826	0.9464 ± 0.0154	0.9313 ± 0.0438	0.9059 ± 0.0650	0.8757 ± 0.0287
	Recall	0.9023 ± 0.0668	0.7387 ± 0.0642	0.9211 ± 0.0310	0.9414 ± 0.0205	0.9066 ± 0.0339	0.8820 ± 0.0483
	Jaccard Index	0.7640 ± 0.1623	0.6437 ± 0.1671	0.8183 ± 0.1156	0.8235 ± 0.1333	0.7858 ± 0.1346	0.7671 ± 0.1671
RPLS-Net	DC	0.8956 ± 0.0532	0.8188 ± 0.0679	0.9236 ± 0.0286	0.9552 ± 0.0254	0.9351 ± 0.0304	0.9057 ± 0.0358
	ASSD	3.6705 ± 1.9807	4.1332 ± 1.4911	2.8075 ± 1.7102	1.3967 ± 0.4519	1.9061 ± 0.5417	2.7828 ± 1.7246
	Precision	0.8788 ± 0.1316	0.8198 ± 0.1325	0.9519 ± 0.0312	0.9546 ± 0.0531	0.9462 ± 0.0270	0.9102 ± 0.0443
	Recall	0.9021 ± 0.0268	0.8207 ± 0.1052	0.9546 ± 0.0398	0.9462 ± 0.0298	0.9510 ± 0.0379	0.9149 ± 0.0328
	Jaccard Index	0.8109 ± 0.1042	0.6932 ± 0.1038	0.8681 ± 0.0556	0.9131 ± 0.0622	0.8782 ± 0.0867	0.8309 ± 0.1002
nmPLS-Net	DC	0.9197 ± 0.0438	0.8508 ± 0.0539	0.9381 ± 0.0227	0.9299 ± 0.0382	0.9238 ± 0.0281	0.9125 ± 0.0303
	ASSD	2.2433 ± 1.7509	3.5638 ± 2.2124	1.4560 ± 0.4983	1.5622 ± 1.3761	1.3036 ± 0.3838	2.0258 ± 0.7938
	Precision	0.8853 ± 0.0627	0.8711 ± 0.0895	0.9532 ± 0.0155	0.9447 ± 0.0247	0.9592 ± 0.0275	0.9227 ± 0.0184
	Recall	0.9150 ± 0.0583	0.8831 ± 0.0656	0.9484 ± 0.0463	0.9368 ± 0.0223	0.9696 ± 0.0108	0.9306 ± 0.0157
	Jaccard Index	0.8513 ± 0.0908	0.7403 ± 0.1045	0.8834 ± 0.0512	0.8690 ± 0.0499	0.8584 ± 0.0697	0.8405 ± 0.0817

Table 3. Ablation experiments, trained and tested on the base dataset. The right symbols at the first two columns represent the use of the corresponding module. FEAL represents fissure-enhanced associative learning. The best results in the table are highlighted with red, and the second best results are highlighted with blue.

nmODE	FEAL	Metric	RU	RM	RL	LU	LL	AVG
		DC	0.9309 ± 0.0709	0.8690 ± 0.1182	0.9511 ± 0.0348	0.9442 ± 0.0528	0.9441 ± 0.0425	0.9279 ± 0.5470
		ASSD	1.7678 ± 0.6075	3.1171 ± 0.988	1.5129 ± 0.5662	2.0548 ± 2.0700	1.7204 ± 1.2939	2.0346 ± 0.7194
		Precision	0.9388 ± 0.0410	0.8704 ± 0.0896	0.9412 ± 0.0502	0.9738 ± 0.0171	0.9208 ± 0.0912	0.9290 ± 0.0221
		Recall	0.9555 ± 0.0440	0.9078 ± 0.0347	0.9173 ± 0.0912	0.9745 ± 0.0136	0.9404 ± 0.0625	0.9391 ± 0.0341
		Jaccard Index	0.8717 ± 0.1200	0.7683 ± 0.1703	0.9058 ± 0.0772	0.8943 ± 0.1133	0.8943 ± 0.0907	0.8668 ± 0.5675
✓		DC	0.9648 ± 0.0066	0.9187 ± 0.0457	0.9655 ± 0.0087	0.9553 ± 0.0188	0.9446 ± 0.0458	0.9498 ± 0.0190
		ASSD	1.2216 ± 0.5127	1.8847 ± 0.6801	1.0419 ± 0.1517	0.9042 ± 0.3784	1.0596 ± 0.4025	1.2224 ± 0.2559
		Precision	0.9600 ± 0.0362	0.9200 ± 0.034	0.9653 ± 0.0105	0.9507 ± 0.0367	0.9504 ± 0.0486	0.9493 ± 0.0183
		Recall	0.9590 ± 0.0061	0.9372 ± 0.0219	0.9597 ± 0.0116	0.9449 ± 0.0921	0.9509 ± 0.0130	0.9523 ± 0.0180
		Jaccard Index	0.9320 ± 0.0185	0.8496 ± 0.1012	0.9333 ± 0.0636	0.9144 ± 0.0692	0.8950 ± 0.1024	0.9049 ± 0.0675
✓		DC	0.9692 ± 0.0075	0.9250 ± 0.0331	0.9660 ± 0.0062	0.9676 ± 0.0234	0.9556 ± 0.0237	0.9567 ± 0.0110
		ASSD	0.9574 ± 0.2561	1.5769 ± 0.4282	0.9529 ± 0.1772	1.0195 ± 0.7127	1.2403 ± 0.8388	1.1494 ± 0.3035
		Precision	0.9635 ± 0.0420	0.9276 ± 0.0843	0.9679 ± 0.0152	0.9705 ± 0.0178	0.9457 ± 0.0206	0.9550 ± 0.0200
		Recall	0.9659 ± 0.0068	0.9488 ± 0.0230	0.9628 ± 0.0149	0.9633 ± 0.0316	0.9576 ± 0.0182	0.9597 ± 0.0077
		Jaccard Index	0.9338 ± 0.0122	0.8603 ± 0.0425	0.9397 ± 0.0149	0.9385 ± 0.0410	0.9182 ± 0.0472	0.9181 ± 0.0166
✓	✓	DC	0.9640 ± 0.0127	0.9336 ± 0.0217	0.9620 ± 0.0198	0.9660 ± 0.0200	0.9596 ± 0.0223	0.9570 ± 0.0100
		ASSD	1.0263 ± 0.3399	1.4077 ± 0.3747	0.9050 ± 0.1429	0.9642 ± 0.6733	1.2633 ± 1.0183	1.1133 ± 0.3381
		Precision	0.9636 ± 0.0363	0.9342 ± 0.0443	0.9705 ± 0.0248	0.9632 ± 0.0334	0.9572 ± 0.0495	0.9578 ± 0.0181
		Recall	0.9641 ± 0.0134	0.9583 ± 0.0188	0.9601 ± 0.0142	0.9634 ± 0.0435	0.9618 ± 0.0183	0.9615 ± 0.0095
		Jaccard Index	0.9300 ± 0.0319	0.8781 ± 0.0337	0.9330 ± 0.029	0.9292 ± 0.0523	0.9223 ± 0.0517	0.9185 ± 0.0236

The results of the ablation experiments on the loss functions are presented in Table 4. Only the mean values of each metric are listed to simplify the table. The experimental results indicate that using a mixed loss function for pulmonary lobe segmentation tasks can lead to some improvement, although it has minimal overall impact. It can also be observed that dice or cross-entropy, as the loss function for the FEAL module, is less effective than not using the FEAL module. This is primarily due to the extreme class imbalance in the labels used by FEAL, where both dice and cross-entropy struggle to effectively calculate the loss contributions of the sparsely represented positive samples.

Table 4. Ablation experiments for the loss function, trained and tested on the base dataset. mDC, mASSD, mPrecision, mRecall and mJaccard-Index are the mean dice coefficient (DC), mean average symmetric surface distance (ASSD), mean precision, mean recall, and mean Jaccard index, respectively. The content within the brackets in the table header represents $(\mathcal{L}_{main}, \mathcal{L}_e)$. The best results in the table are highlighted with red, and the second best results are highlighted with blue.

Metric	(CE, Focal)	(Dice, Focal)	(CE+Dice, Focal)	(Dice+Focal, Dice)	(Dice+Focal, CE)	(CE+Focal, Focal)	(Dice+Focal, Focal)
mDC	0.9519 $\pm$ 0.0192	0.9552 $\pm$ 0.0238	0.9563 $\pm$ 0.0145	0.9531 $\pm$ 0.0178	0.9540 $\pm$ 0.0178	0.9558 $\pm$ 0.0126	0.9570 $\pm$ 0.0100
mASSD	1.2538 $\pm$ 0.4015	1.5171 $\pm$ 0.5332	1.1427 $\pm$ 0.3955	1.1411 $\pm$ 0.4172	1.2511 $\pm$ 0.4318	1.1503 $\pm$ 0.3524	1.1133 $\pm$ 0.3381
mPrecision	0.9528 $\pm$ 0.0190	0.9517 $\pm$ 0.0336	0.9556 $\pm$ 0.0202	0.9509 $\pm$ 0.0259	0.9538 $\pm$ 0.0271	0.9566 $\pm$ 0.0252	0.9578 $\pm$ 0.0181
mRecall	0.9581 $\pm$ 0.0099	0.9545 $\pm$ 0.0162	0.9597 $\pm$ 0.0131	0.9540 $\pm$ 0.0209	0.9578 $\pm$ 0.0125	0.9606 $\pm$ 0.0113	0.9615 $\pm$ 0.0095
mJaccard-Index	0.9092 $\pm$ 0.0343	0.9141 $\pm$ 0.0544	0.9159 $\pm$ 0.0485	0.9120 $\pm$ 0.0415	0.9131 $\pm$ 0.0373	0.9153 $\pm$ 0.0447	0.9186 $\pm$ 0.0236

4.4. Qualitative Analysis

Quantitative evaluation validates the effectiveness of nmPLS-Net in segmenting right pulmonary lobes and in case of unclear pulmonary fissures. Figure 4 showcases the segmentation outcomes achieved by our network and the comparative networks across five distinct CT scans. In the five CT cases, the first CT represents a normal lung with intact and clear pulmonary fissures. In the remaining four CT cases, partial or complete unclear pulmonary fissures are present. It can be observed that in the CTs of rows 2, 3, and 5, nmPLS-Net consistently demonstrates superior performance. In contrast, the other networks exhibit poor performance, with some results displaying noticeable segmentation errors. While the segmentation results in the CT scans of the third row are generally suboptimal, nmPLS-Net still provides the closest approximation to the ground truth. This case underscores nmPLS-Net's adeptness at effectively handling scenarios with unclear or incomplete pulmonary fissures.

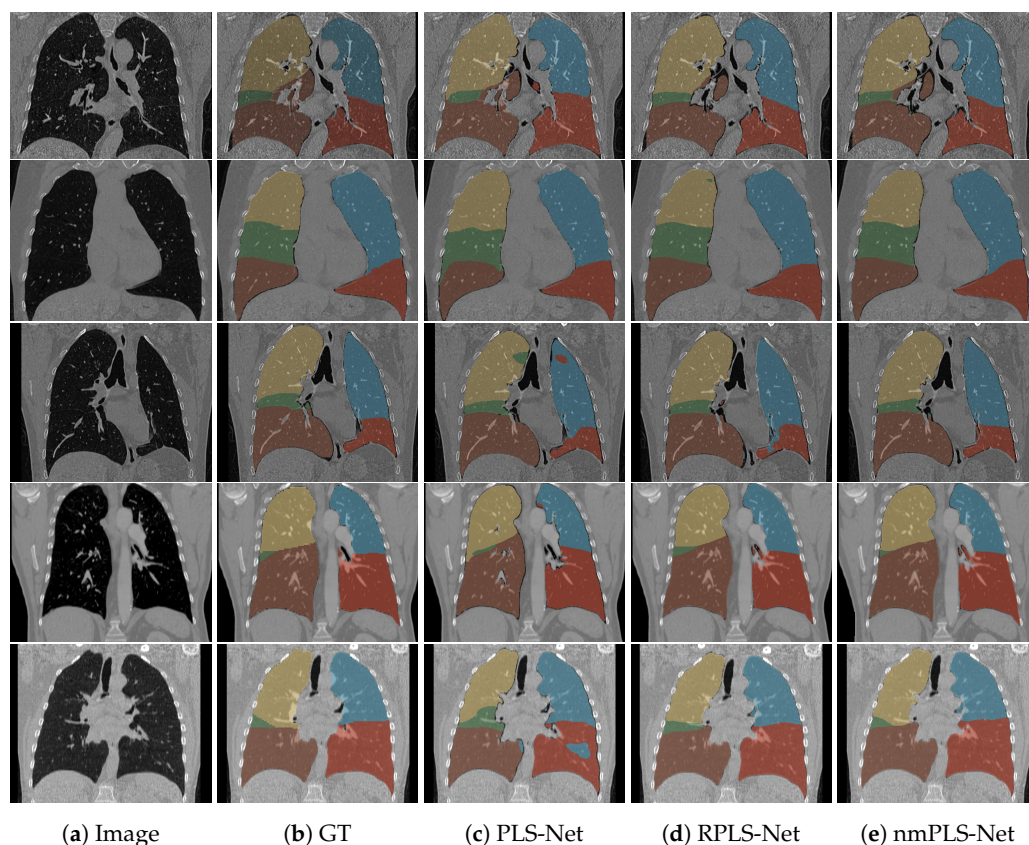


Figure 4. The segmentation results of three CT scans in nmPLS-Net and the comparative networks. In this visual representation, distinct lung regions are color-coded: yellow for the right upper lobe,

green for the right middle lobe, brown for the right lower lobe, blue for the left upper lobe, and red for the left lower lobe. The first column represents the original images, and the subsequent four columns represent the segmentation results for the ground truth, PLS-Net, RPLS-Net, and nmPLS-Net, respectively.

To further substantiate the efficacy of nmODE, an analysis was conducted on feature maps generated by the model using six CT scans. These feature maps included those with dimensions $(129, D/8, H/8, W/8)$ before entering the nmODE module, as well as those with the same dimensions after passing through the nmODE module. Principal component analysis (PCA) was applied to reduce the dimensionality of these feature maps from $(129, D/8, H/8, W/8)$ to a compact representation of $(129, 2)$. These reduced-dimensional feature representations were then visualized on a 2D plane, and the outcomes are presented in Figure 5. Figure 5a provides a visual representation of features before entering the nmODE module, while Figure 5b showcases the features after undergoing processing by the nmODE module. Each color in the visualizations corresponds to the same CT scan across both figures. It can be observed that the original features, after being processed by nmODE, exhibit more pronounced overall similarity in shape, indicating that nmODE's memory capacity can indeed extract common features among different samples. Furthermore, the distribution of features before processing by the nmODE module is more concentrated. In contrast, the feature distribution following nmODE processing exhibits a heightened level of uniformity and reveals stronger inter-feature correlations. These findings underscore the nmODE module's adeptness in learning intricate feature correlations, thus enhancing its capacity to capture meaningful relationships within the data.

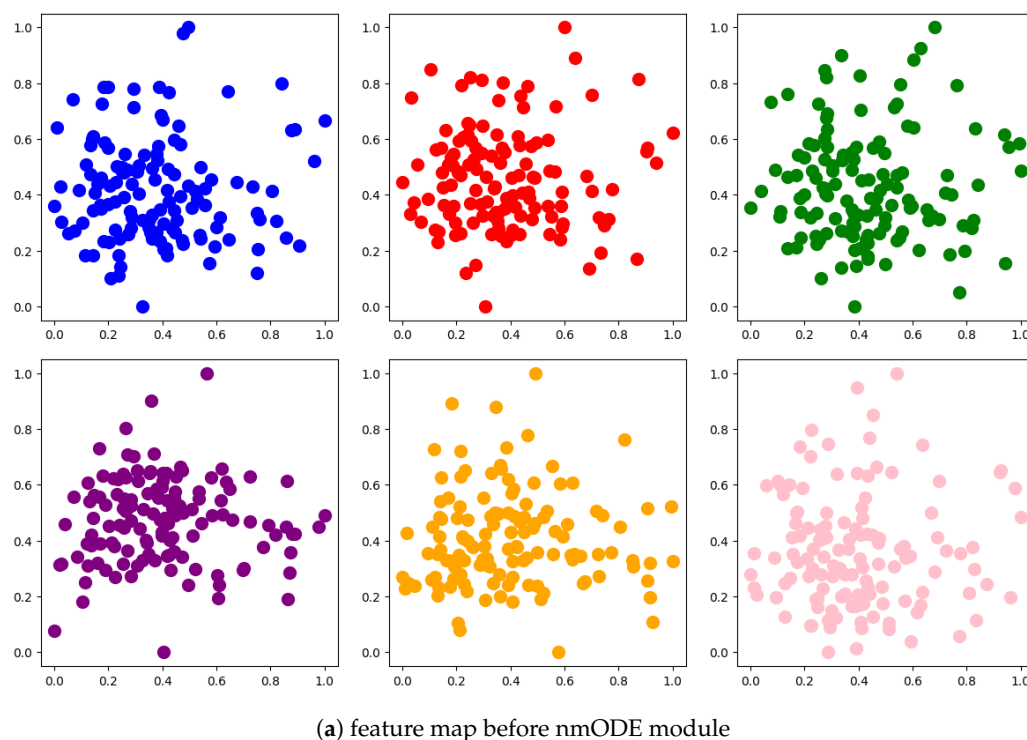
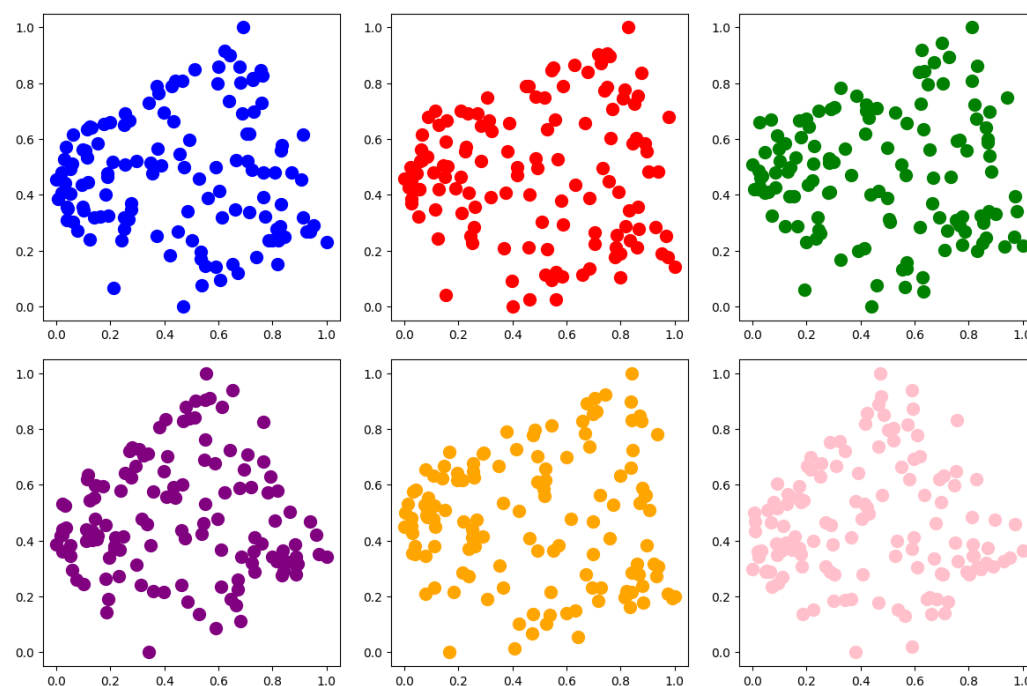


Figure 5. Cont.



(b) feature map after nmODE module

Figure 5. Visualized feature-maps of 6 CT volumes. In (a,b), the same colors represent the output features of the same CT volumes and each scatterplot in the figure is obtained by projecting the various channels of the model's encoder output features onto a two-dimensional plane after PCA dimension reduction. (a) shows the scatterplot of the feature mapping obtained before processing with the nmODE module, while (b) shows the scatterplot of the feature mapping obtained after processing with the nmODE module.

5. Discussion and Conclusions

This study proposes a 3D fully convolutional neural network, nmPLS-Net, which combines nmODE (nonlinear memory ordinary differential equation) with multitask learning to perform the end-to-end segmentation of pulmonary lobes in chest CT scans. The memory capabilities and high nonlinearity of the nmODE module enhance the extraction of global features, improving the model's performance in segmenting the overall lung and complex scenarios in the right lung. Additionally, fissure-enhanced associative learning guides the model's attention toward challenging voxels within pulmonary fissures. The model's performance is validated on two distinct datasets, and the experimental results demonstrate that nmPLS-Net achieves notably superior results in right pulmonary lobe segmentation compared to previous methods. However, incorporating nmODE requires an ODE solver, which may slightly reduce computational speed. This problem can be addressed in future work through knowledge distillation, an area we plan to explore for further improvements.

Author Contributions: Conceptualization, Z.Y. and P.D.; methodology, P.D.; software, P.D. and H.N.; validation, P.D.; data curation, P.D. and H.N.; writing—original draft, P.D. and H.N.; writing—review and editing, Z.Y. and X.X.; supervision, Z.Y. and X.X.; All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Major Science and Technology Projects of China under Grant 2018AAA0100201, the National Natural Science Foundation of China under Grant 62106163, Major Science and Technology Project from the Science & Technology Department of Sichuan Province under Grant 2020YFG0473, the Natural Science Foundation Project of Sichuan Province under Grant 2023YFG0283 and the CAAI-Huawei MindSpore Open Fund under Grant 21H1235.

Data Availability Statement: The base dataset is public. The LUNA16 dataset presented in this study is openly available in https://github.com/deep-voxel/automatic_pulmonary_lobe_segmentation_using_deep_learning at <https://doi.org/10.1109/ISBI.2019.8759468>, reference number [15] (all accessed on 10 April 2019).

Conflicts of Interest: The authors declare no conflict of interest.

References

- Yi, Z. nmODE: Neural memory ordinary differential equation. *Artif. Intell. Rev.* **2023**, *56*, 14403–14438. [\[CrossRef\]](#)
- Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
- Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings, Part III 18; Proceedings of the Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015: 18th International Conference, Munich, Germany, 5–9 October 2015*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 234–241.
- Shoaib, M.A.; Lai, K.W.; Chuah, J.H.; Hum, Y.C.; Ali, R.; Dhanalakshmi, S.; Wang, H.; Wu, X. Comparative studies of deep learning segmentation models for left ventricle segmentation. *Front. Public Health* **2022**, *10*, 981019. [\[CrossRef\]](#) [\[PubMed\]](#)
- Zhao, Z.; Chuah, J.H.; Lai, K.W.; Chow, C.O.; Gochoo, M.; Dhanalakshmi, S.; Wang, N.; Bao, W.; Wu, X. Conventional machine learning and deep learning in Alzheimer’s disease diagnosis using neuroimaging: A review. *Front. Comput. Neurosci.* **2023**, *17*, 1038636. [\[CrossRef\]](#) [\[PubMed\]](#)
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; Volume 30.
- Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Virtual, 11–17 October 2021; pp. 10012–10022.
- Liu, Z.; Hu, H.; Lin, Y.; Yao, Z.; Xie, Z.; Wei, Y.; Ning, J.; Cao, Y.; Zhang, Z.; Dong, L.; et al. Swin transformer v2: Scaling up capacity and resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–24 June 2022; pp. 12009–12019.
- Chen, J.; Lu, Y.; Yu, Q.; Luo, X.; Adeli, E.; Wang, Y.; Lu, L.; Yuille, A.L.; Zhou, Y. Transunet: Transformers make strong encoders for medical image segmentation. *arXiv* **2021**, arXiv:2102.04306.
- Shen, J.; Lu, S.; Qu, R.; Zhao, H.; Zhang, L.; Chang, A.; Zhang, Y.; Fu, W.; Zhang, Z. A boundary-guided transformer for measuring distance from rectal tumor to anal verge on magnetic resonance images. *Patterns* **2023**, *4*, 100711. [\[CrossRef\]](#) [\[PubMed\]](#)
- Liu, Y.; Zhang, Y.; Wang, Y.; Hou, F.; Yuan, J.; Tian, J.; Zhang, Y.; Shi, Z.; Fan, J.; He, Z. A survey of visual transformers. *arXiv* **2023**, arXiv:2111.06091.
- Milletari, F.; Navab, N.; Ahmadi, S.A. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), Stanford, CA, USA, 25–28 October 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 565–571.
- Imran, A.A.Z.; Hatamizadeh, A.; Ananth, S.P.; Ding, X.; Terzopoulos, D.; Tajbakhsh, N. Automatic segmentation of pulmonary lobes using a progressive dense V-network. In Proceedings of the International Workshop on Deep Learning in Medical Image Analysis, Granada, Spain, 20 September 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 282–290.
- Ferreira, F.T.; Sousa, P.; Galdran, A.; Sousa, M.R.; Campilho, A. End-to-end supervised lung lobe segmentation. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–8.
- Tang, H.; Zhang, C.; Xie, X. Automatic pulmonary lobe segmentation using deep learning. In Proceedings of the 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), Venice, Italy, 8–11 April 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1225–1228.
- Lee, H.; Matin, T.; Gleeson, F.; Grau, V. Efficient 3D fully convolutional networks for pulmonary lobe segmentation in CT images. *arXiv* **2019**, arXiv:1909.07474.
- Xie, W.; Jacobs, C.; Charbonnier, J.P.; Van Ginneken, B. Relational modeling for robust and efficient pulmonary lobe segmentation in CT scans. *IEEE Trans. Med Imaging* **2020**, *39*, 2664–2675. [\[CrossRef\]](#) [\[PubMed\]](#)
- Fan, X.; Xu, X.; Feng, J.; Huang, H.; Zuo, X.; Xu, G.; Ma, G.; Chen, B.; Wu, J.; Huang, Y.; et al. Learnable interpolation and extrapolation network for fuzzy pulmonary lobe segmentation. *IET Image Process.* **2023**, *17*, 3258–3270. [\[CrossRef\]](#)
- Liu, J.; Wang, C.; Guo, J.; Shao, J.; Xu, X.; Liu, X.; Li, H.; Li, W.; Yi, Z. RPLS-Net: Pulmonary lobe segmentation based on 3D fully convolutional networks and multi-task learning. *Int. J. Comput. Assist. Radiol. Surg.* **2021**, *16*, 895–904. [\[CrossRef\]](#)
- He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NA, USA, 26 June–1 July 2016; pp. 770–778.
- Haber, E.; Ruthotto, L. Stable architectures for deep neural networks. *Inverse Probl.* **2017**, *34*, 014004. [\[CrossRef\]](#)
- Chen, R.T.Q.; Rubanova, Y.; Bettencourt, J.; Duvenaud, D.K. Neural Ordinary Differential Equations. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 3–8 December 2018; Volume 31.

23. Dupont, E.; Doucet, A.; Teh, Y.W. Augmented Neural ODEs. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Volume 32.
24. Chen, R.T.Q.; Amos, B.; Nickel, M. Learning Neural Event Functions for Ordinary Differential Equations. In Proceedings of the International Conference on Learning Representations, Virtual Event, 3–7 May 2021.
25. Zhang, T.; Yao, Z.; Gholami, A.; Gonzalez, J.E.; Keutzer, K.; Mahoney, M.W.; Biros, G. ANODEV2: A coupled neural ODE framework. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019; Volume 32.
26. Chollet, F. Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1251–1258.
27. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
28. Poucet, B.; Save, E. Attractors in memory. *Science* **2005**, *308*, 799–800. [[CrossRef](#)]
29. Wills, T.J.; Lever, C.; Cacucci, F.; Burgess, N.; O’Keefe, J. Attractor dynamics in the hippocampal representation of the local environment. *Science* **2005**, *308*, 873–876. [[CrossRef](#)] [[PubMed](#)]
30. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 8–14 December 2019, Volume 32.
31. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.