



**Research article**

## **An improved residual U-Net with morphological-based loss function for automatic liver segmentation in computed tomography**

**Peiqing Lv<sup>1</sup>, Jinke Wang<sup>1,2,\*</sup>, Xiangyang Zhang<sup>1</sup>, Chunlei Ji<sup>2</sup>, Lubiao Zhou<sup>1</sup> and Haiying Wang<sup>1</sup>**

<sup>1</sup> School of Automation, Harbin University of Science and Technology, Harbin 150080, China

<sup>2</sup> Department of Software Engineering, Harbin University of Science and Technology, Rongcheng 264300, China

**\* Correspondence:** Email: [jkwang@hitwh.edu.cn](mailto:jkwang@hitwh.edu.cn); Tel: +86-13863132787; Fax: +86-06317595512.

**Abstract:** This paper proposes an improved ResU-Net framework for automatic liver CT segmentation. By employing a new loss function and data augmentation strategy, the accuracy of liver segmentation is improved, and the performance is verified on two public datasets LiTS17 and SLiver07. Firstly, to speed up the convergence of the model, the residual module is used to replace the original convolution module of U-Net. Secondly, to suppress the problem of pixel imbalance, the opposite number of Dice is proposed to replace the cross-entropy loss function, and the morphological method is introduced to weigh the pixels. Finally, to improve the generalization ability of the model, random affine transformation and random elastic deformation are employed for data augmentation. From 20 training datasets of Sliver07, 16 sets were selected as the training set, two sets were used for verification, and two sets were used for the test; meanwhile, from 131 training datasets of LiTS2017, eight sets were selected as the test set. In the experiment, four evaluation metrics, including DICE global, DICE per case, VOE, and RVD, were calculated, with the accuracies of 94.28,  $94.24 \pm 2.07$ ,  $10.83 \pm 3.70$ , and  $-0.25 \pm 2.74$ , respectively. Compared with U-Net and ResU-Net, the performance of the proposed method is significantly improved. The experimental results show that, although the method's complexity is high, it has a faster convergence speed and stronger generalization ability. The segmentation effect on the 2D image is significantly improved, and the scalability on 3D data is also robust. In addition, the proposed method performs well in the case of low-contrast neighboring organs, which proves the robustness of the proposed method.

**Keywords:** liver segmentation; CT; U-Net; residual; morphology

---

## 1. Introduction

Liver cancer is the third cause of cancer death. Therefore, in liver CT diagnosis and surgical planning, segmentation is an essential prerequisite. However, when faced with hundreds of CT slices, clinicians have to manually depict the liver contour and lesions slice by slice, which is time-consuming, labor-intensive, and highly dependent on clinicians' subjective experiences. Therefore, in clinical practice, there is an urgent need for high-precision automatic liver segmentation methods.

For the automatic segmentation of liver CT, many algorithms, including thresholding [1], deformation model [2,3], statistical model [4,5], active contour model [6] and machine learning-based methods, have been proposed in the past decades [7–9].

The above methods can effectively achieve liver segmentation, most of them require manual intervention, and the degree of automation is not high. Moreover, due to the partial volume effect of CT imaging and the inter-differences between annotators [8], the conventional statistical information model-based methods are often not feasible. At the same time, the shape complexity and variability of liver tissue also make it difficult for deformation model-based methods to adapt to complex liver segmentation tasks [10].

Nevertheless, with the rapid development of hardware in recent years, deep learning-based technology has shown outstanding advantages, particularly in image classification and segmentation [11–13].

Currently, there are mainly two kinds of methods for automatic liver segmentation based on deep learning. One is to transform the segmentation problem into a pixel classification problem, and the other is to perform segmentation through a self-coding network directly.

The first type of method, such as Li et al. [14], divides the image into several small blocks for classification to obtain the segmentation results of liver and pathological region. However, this kind of method has an inherent defect; that is, the simple sliding window method would produce a large number of image blocks, and the same pixel in the image will be repeatedly calculated multiple times, which would cause a considerable system cost.

The second type of method is to segment the entire image and output the target mask directly. FCN [15] is a typical representative of this method, which directly obtains the segmentation results by changing the full connection layer to the convolution layer. Based on FCN, Ronneberger et al. [16] proposed a symmetrical and elegant U-shaped structure (U-Net). This network structure first proposed the combination of contraction path and expansion path feature graph and verified the excellent performance of the method in cell edge segmentation experiments. Ben Cohen et al. [17] used the improved U-Net for liver segmentation and liver metastasis detection. At the same time, it removed the cross-connection in the network and achieved significantly better results than the image block-based method.

The above-mentioned deep learning-based are all based on 2D image strategy. In recent years, more and more methods directly segment 3D data. Cicek et al. [18] proposed a 3D U-Net that replaces the original 2D convolution with 3D convolution and performs segmentation tasks directly on the 3D data. Dou et al. [19] combine the 3D deeply supervised network (3D DSN) with the conditional random field and introduce a deep-level supervision mechanism in the learning process to overcome potential optimization problems, to obtain faster convergence speed and more accurate recognition ability. Zhao et al. [20] proposed a method for automatically segmenting lung parenchyma, which uses three-dimensional V-Net for end-to-end lung extraction, and uses a deformation module to limit the V-Net

output based on prior shape knowledge. Experimental results show that the salient features measured in their segmentation results are consistent with the features in manual annotations. Besides, Zhao et al. [21] also extracted coronary arteries through deep learning and detecting arterial stenosis from ICAs. This model combines the feature pyramid with the U-Net++ model to automatically segment the coronary arteries in ICAs. Experimental results show excellent clinical application prospects and can provide auxiliary suggestions for CAD diagnosis and treatment.

Hoogi et al. [22] used a convolutional neural network to promote the level set segmentation method and used the output CNN probability map to adaptively calculate the parameters of the active contour function. Hu et al. [23] proposed an automatic segmentation framework based on a 3D convolutional neural network and global optimization surface evolution. The framework first trains a deep 3D CNN to learn the probability map of the liver, then adaptively integrates the global and local appearance information obtained from segmentation into the model and globally optimizes it through surface evolution. Lu et al. [24] developed a deep learning algorithm with image segmentation and refinement to automatically segment liver CT. Firstly, 3D CNN is used for liver detection and probabilistic segmentation. Then, the initial segmentation is refined using the graph cut result and the previously learned probabilistic map.

The method based on deep CNN can solve the problems of the tedious extraction process of manual feature extraction and provide an end-to-end learning approach, which significantly simplifies the learning steps and can substantially improve the performance of segmentation.

This paper mainly improves the network structure and loss function based on U-Net, with the following main contributions.

- 1) Use the residual module to replace the original convolution module, and speed up the convergence through skip connection and batch normalization.
- 2) Use the opposite number of the Dice coefficient to replace cross-entropy and eliminate the disturbance of the class imbalance.
- 3) Propose a morphology-based weighting method to force the model to deal with liver edges properly.
- 4) Modify the network to a 3D network to verify the segmentation effect of the proposed method on the 3D image.

The following parts of this paper are arranged as follows: the third section gives a detailed description of the proposed network framework and the improved loss function, the fourth section provides the experimental results and analysis, and the final part is the conclusion.

## 2. Related works

### 2.1. Degradation problem of U-Net

In the task of medical image classification, detection, and segmentation, to improve the accuracy of the task, the usual approach is to build a deeper network framework by directly stacking convolution kernels. However, experiments show that with the increase of convolution layers, the effect of the model is not as good as that of the relatively shallow neural network; that is, the network degrades as the number of layers increases.

Although the decoder of the U-Net network can integrate the encoder's low, medium, and high-level features, it is widely used in various medical tasks. However, as the depth of the network deepens, the degradation problem is unavoidable. On the one hand, after each down-sampling of the U-Net

coder, the channel number of the convolution kernel decreases. On the other hand, when a specific layer in the middle of the network compresses the features in a large proportion in the spatial dimension (e.g. when pooling is used), a lot of information will be lost in the feature map [25]. One solution is to double the number of feature channels before pooling, which can be understood as adding redundant features and then pooling again to avoid reducing information.

On the other hand, the maximum pooling needs to record the location index of the pooling during each backpropagation process. Thus, it would increase the model's computing cost to a certain extent. Besides, as the number of network layers deepens, backpropagation becomes more and more complex; that is, the convergence speed of the model tends to decrease.

## 2.2. *Residual-based resolution*

Aiming at the resolution of network degradation, He et al. [26] proposed reconstructing the mapping of the network, in which the original convolution module is replaced with the residual module. Thus, unlike the previous training that relies on multi-layer stacking to learn the implicit mapping, the network expects to learn a residual mapping.

The residual structure has been widely used in many applications of medical image segmentation. For example, to accelerate the convergence speed of the network, Milletari et al. [27] added a residual module to the network to obtain a V-Net segmentation network. They received good results in segmentation tasks, especially in prostate segmentation. Furthermore, Jin et al. [28] proposed a 3D hybrid residual attention perception segmentation network that combines residual structure and U-Net to extract liver and tumors from CT. The model was verified on brats2018 and brats2017 datasets.

Although there are two ways of identity transformation and weight mapping in skip connection, identity transformation is significant for the above bottleneck structure because the weight mapping matrix can make the input and output of skip connection become into two high-dimensional matrices, which would double the computational complexity and model size. Therefore, the skip connection of identity transformation can make the residual network more efficient based on the bottleneck structure.

## 2.3. *Class-imbalance problem and resolution*

In liver segmentation, another challenge is the imbalance of different categories. That is, the segmented objects in different liver images vary significantly in size.

To solve the class imbalance problem, one solution is to improve the loss function. One of the commonly used loss functions in semantic segmentation is cross-entropy. However, cross-entropy does not consider the class imbalance problem, and thus it may be difficult to detect small liver regions or fuzzy boundaries. Christ et al. [29] assigned a weight to each category, where the weight belongs to the reciprocal of the pixel proportion of the category. Milletari et al. [27] proposed a dice loss function in the 3D image segmentation task, effectively reducing the segmentation deviation caused by the imbalance of the ROI area and the background. Sudre et al. [30] proposed a Tversky loss function based on Tversky similarity, taking the difference between false positives and false negatives into account. The Tversky loss function takes two parameters and makes a trade-off between false positive and false negative. Goceri [31,32] proposed a hybrid loss function using the cross-entropy with Tversky loss or mutual information for lesion segmentation from colored images.

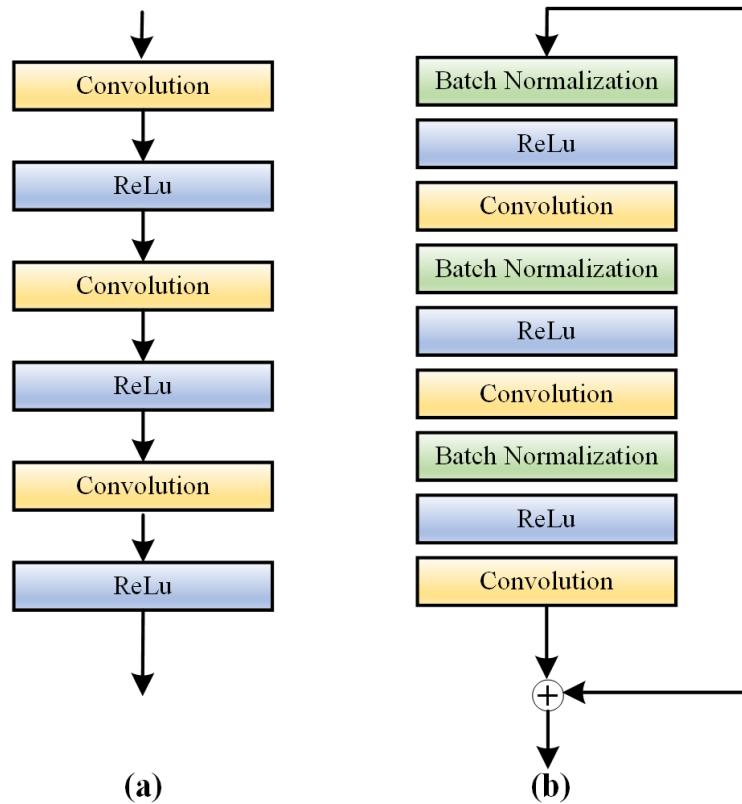
The second solution is the data augmentation strategy, improving the network's generalization

ability, enhancing the system's robustness, and reducing the overfitting problem. General data augmentation strategies mainly include rigid transformation and non-rigid transformation to improve the network's generalization ability.

### 3. Method

#### 3.1. Proposed network framework

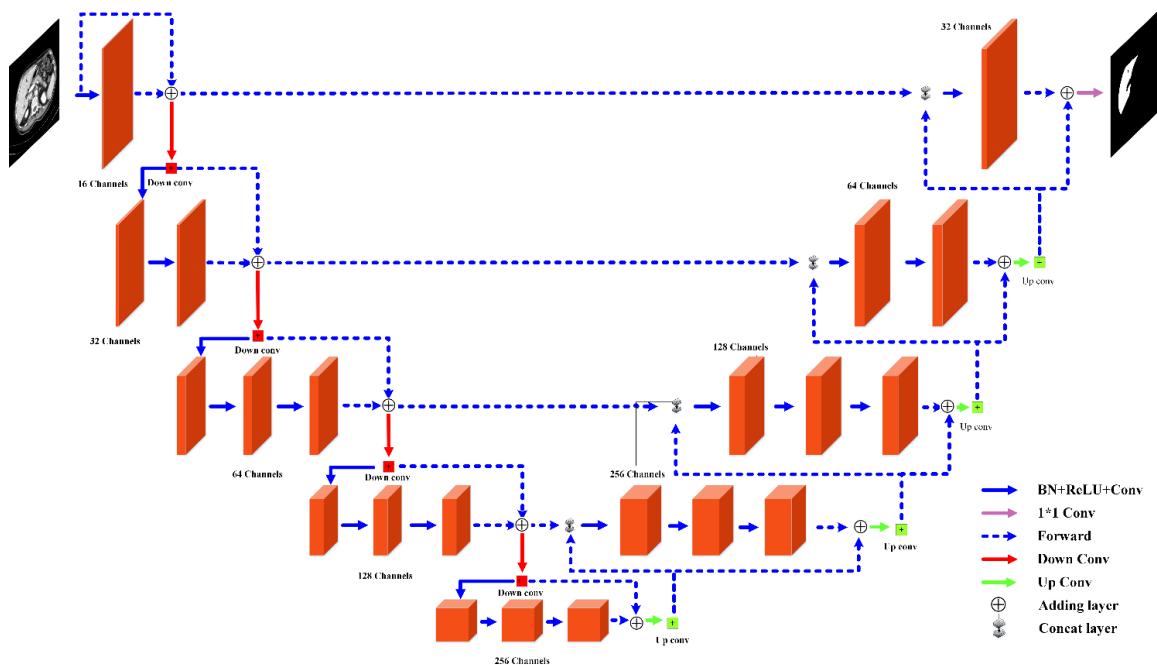
The improved network framework presented in this paper is based on the V-Net [27] proposed by Milletari et al. and the residual U-Net [31] proposed by Zhang et al. Their core modules are shown in Figure 1.



**Figure 1.** Convolutional and residual modules (a) convolution (b) residual module.

The improved network structure proposed in this paper is shown in Figure 2. The main modifications are as follows: First, before each down-sampling, the number of channels is changed to twice the number of the upper layer; at the same time, before each up-sampling (i.e., transpose convolution), the number of channels of convolution kernel is halved, and thereby effectively reducing the bottleneck of the model. Secondly, using convolution with a step size of 2 instead of pooling, on the one hand, it can alleviate the information loss caused by directly discarding features, on the other hand, it can reduce the memory consumption. Thirdly, for the problem of slow model convergence, it introduces residual structure to improve the model, to speed up the convergence. At the same time, the skip connections in the network can promote information propagation, reduce the number of

parameters, and thus improve the network performance.



**Figure 2.** Proposed iResU-Net network structure.

In Figure 2, each red cube represents the feature map obtained after convolution.

- 1) The solid blue arrow indicates the process of BN (Batch Normalization), ReLU activation function, and convolution (kernel size is  $3 \times 3$ , the step size is 1) on the input data in sequence.
- 2) The purple arrow represents the convolution process with a convolution kernel size of  $1 \times 1$ , a step size of 1, and an activation function of sigmoid.
- 3) The blue dashed arrow represents the forward propagation process of the information flow in the network without any operation. Note that the blue dashed arrow has two main functions: on the one hand, it is used for the skip connection of the ResNet module, and on the other hand, it is the skip connection that transmits high-resolution information to low-resolution in the original U-Net.
- 4) The red arrow is a convolution with a convolution kernel size of  $2 \times 2$  and a step size of 2. The number of convolution kernels is twice the number of channels of the input feature map, and its primary purpose is to down-sample the feature map.
- 5) The green arrow indicates a transposed convolution with a convolution kernel size of  $2 \times 2$  and a step size of 2. The number of convolution kernels is half of the number of channels of the input feature map, and its primary purpose is to up-sample the feature map.
- 6) The Adding layer only adds the corresponding positions of the two inputs, while the Concat layer concatenates the two inputs in the channel dimension.

### 3.2. Proposed loss function

In traditional U-Net, the cross-entropy loss function is calculated on the output result of Softmax as the loss function of the entire network. Pre-calculate the corresponding weight distribution map for each pixel in the image, as shown in Eq (1).

$$w(x) = w_c(x) + w_0 \cdot \exp\left(-\frac{(d_1(x)+d_2(x))^2}{2\sigma^2}\right) \quad (1)$$

In the first part,  $w_c(x)$  is used to compensate for the imbalance between different types of pixels in the dataset. In the second part, by calculating the square of the sum of the distance between the current pixel and the closest edge and the distance between the current pixel and the second nearest edge, the network pays more attention to learning the edge of cells in contact with each other. Thus, Eq (1) mainly solves two problems to strengthen the network's learning ability for the target task. One is the problem of sample imbalance; the other is the challenge of the task itself, which often leads to the low accuracy of edge segmentation in the liver segmentation task. Based on these two problems, combined with the specific situation of liver segmentation, this paper improves the loss function.

### 3.2.1. Redefinition of the loss function

Using cross-entropy as the loss function can only indirectly estimate the effect of the current model but cannot directly evaluate it. When the loss function reaches the minimum value, it cannot guarantee that the model obtains the best performance but only ensures a relatively good result. Since the DICE coefficient only cares about the gap between the segmentation result and the ground truth, the employment of the opposite of the Dice coefficient can directly avoid the problem of the uneven number of pixels in different categories without using weighting. Besides, cross-entropy as the loss function can be used to estimate the effect of the current model, but not directly; that is, when the loss function reaches the minimum, it can not guarantee the best performance but only a relatively good effect. Therefore, this paper uses the opposite of the Dice coefficient as the loss function and only focuses on the gap between the segmentation result and the ground truth, which can directly avoid the imbalance problem of the pixels in different categories without weighting. The Equation is shown as follows:

$$l = -\frac{2 \times \sum w_x Y_{pre\_x} Y_{true\_x} + \varepsilon}{\sum_x w_x Y_{pre\_x} + \sum_x w_x Y_{true\_x} + \varepsilon} \quad (2)$$

where  $Y_{pre\_x}$  indicates the prediction result of the  $x$ -th pixel in the prediction result,  $Y_{true\_x}$  represents the category label of the  $x$ -th pixel in the actual label,  $w_x$  is the weight of the  $x$ -th pixel, and  $\varepsilon$  is used to prevent the numerator or denominator from being zero.

### 3.2.2. Definition of the weight of loss function

The second improvement is to modify the weight of each pixel. The situation of liver segmentation is different from that of cell segmentation: most livers have only one piece, and the gap between a few pieces is significantly larger than that between cells. Thus, the weight definition of the classic U-Net is not suitable for liver segmentation. Besides, in the process of liver segmentation, the following situations often occur:

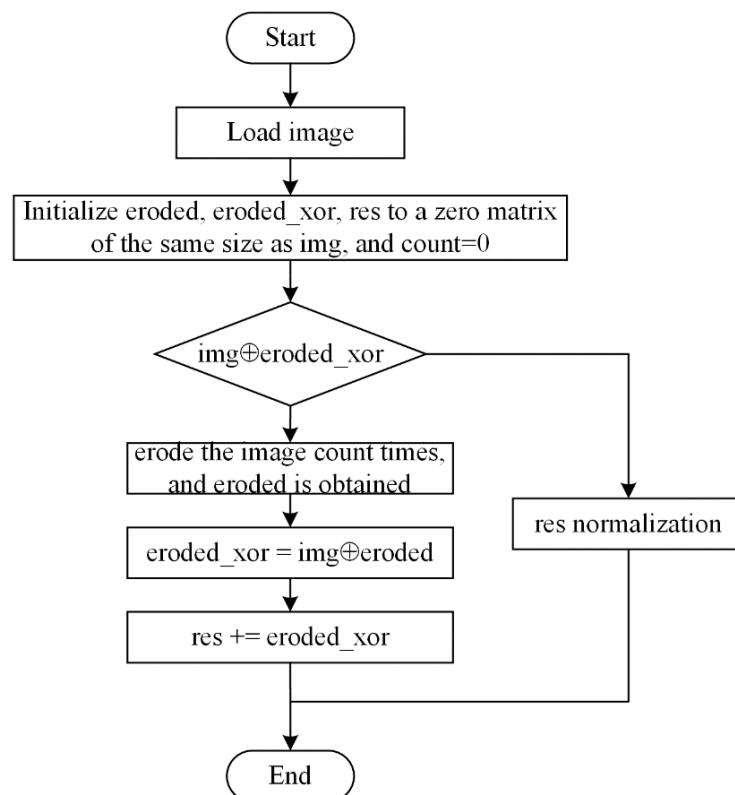
- 1) For smaller liver parts, it is prone to over-segment or under-segmentation;
- 2) The accuracy of liver edge segmentation is not high, which makes the size of segmented liver slightly smaller than that of real one.

The reasons for the above problems can be summarized into the two difficulties mentioned previously, namely the sample imbalance problem and the tricky boundary segmentation problem. The

imbalance problem of pixel categories can be solved by using the inverse number of the Dice coefficient as the loss function.

In addition, this paper also improves the weighting method of the image through the morphological approach, that is, the number of erosions obtains the weight of different pixels and the weight map  $\omega_{morphology}(x)$  of the image is formed, and the value of each point in the weight map is the value of  $w_x$  in the loss function.

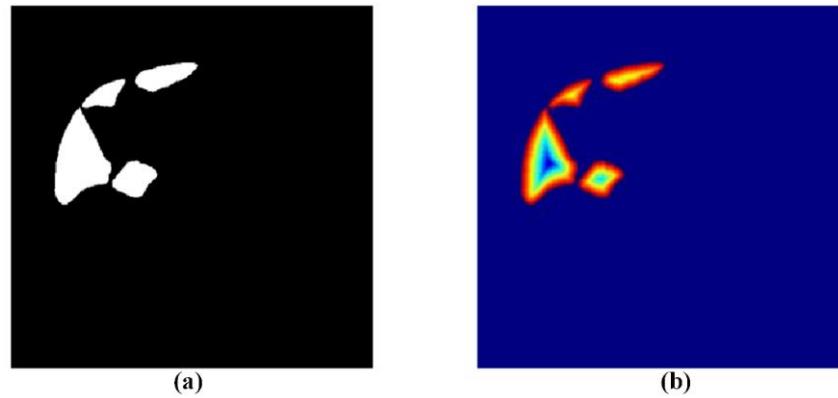
The primary purpose of calculating  $\omega_{morphology}(x)$  is to enhance the model's learning ability on edge and improve the small liver's learning ability. The specific calculation flow of the weight map  $\omega_{morphology}(x)$  is shown in Figure 3.



**Figure 3.** Flow chart of weight calculation algorithm based on morphology.

Through the algorithm flow shown in Figure 3, we can get the weight of each pixel in the image. The calculated weight map can be displayed in the form of a heat map, as shown in Figure 4.

Figure 4(a) is the original label of the liver CT, and Figure 4(b) is the weight map obtained through morphology. It can be seen from Figure 4 that the more difficult-to-segment contours of the liver get a relatively large weight. In contrast, the easier-to-segment interiors get a relatively small weight, forcing the network to pay more attention to learning the liver edges.



**Figure 4.** Weighted heat map of liver (a) label of CT (b) morphological-based weight heat map.

### 3.3. Evaluation metrics

In this paper, each patient's VOE (Volume Overlap Error), RVD (Relative Volume Difference), ASD (Average Symmetric Surface Distance), and MSD (Maximum Symmetric Surface Distance) are used to evaluate the proposed method.

Therefore, to properly analyze the overlapping degree between the ground truth and the segmentation results of the proposed model, this paper uses the VOE to evaluate the effect of the model, which reflects the degree of error between the segmentation results of the model and the ground truth, with the calculation formula shown in Eq (4).

$$VOE = 1 - \frac{|Y_{x_i\_pred} \cap Y_{x_i\_GT}|}{|Y_{x_i\_pred} \cup Y_{x_i\_GT}|} \quad (4)$$

Meanwhile, this paper also introduces the RVD, an asymmetric measurement method used to express the volume difference between the ground truth and the predicted results, with the calculation formula shown in Eq (5).

$$RVD = \frac{|Y_{x_i\_pred}| - |Y_{x_i\_GT}|}{|Y_{x_i\_GT}|} \quad (5)$$

We introduced the average surface distance (ASD), which represents the average distance between the segmentation result and the gold standard surface, where  $d(v, S(X))$  represents the shortest Euler distance from voxel  $v$  to the surface voxel of the segmentation result.

$$ASD = \frac{1}{|Y_{x_i\_pred}| + |Y_{x_i\_GT}|} \left( \sum_{p \in S(Y_{x_i\_GT})} d(p, S(Y_{x_i\_GT})) + \sum_{q \in S(Y_{x_i\_pred})} d(q, S(Y_{x_i\_pred})) \right) \quad (6)$$

Finally, we used the maximum surface distance (MSD), representing the maximum distance between the segmentation result and the gold standard surface.

$$MSD = \max \left\{ \max_{p \in S(Y_{x_i\_pred})} d(p, S(Y_{x_i\_GT})), \max_{q \in S(Y_{x_i\_GT})} d(q, S(Y_{x_i\_pred})) \right\} \quad (7)$$

## 4. Experiment

### 4.1. Dataset and implementation

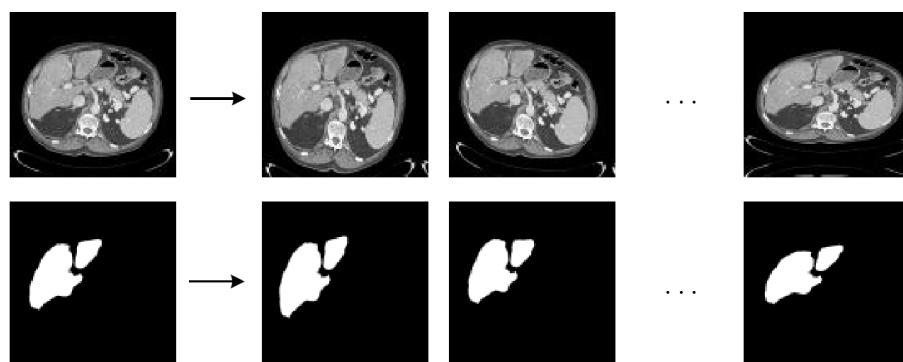
The first dataset of the experiment is from the public dataset LiTS17<sup>1</sup>. The LiTS17 dataset consists of 131 and 70 sets of abdominal CT scans, which can be used for training and testing, respectively. All the training sets provide the ground truth for liver and liver tumors, while the test set does not provide the gold standard, but the performance on each metric can be obtained by participating in the competition. Most of the scans come from patients with various liver tumor diseases and different acquisition protocols. The resolution range of cross-section is [0.56, 1.0] mm, and the resolution of the z-axis image is [0.45, 6.0] mm. The number of slices in each CT is between [42, 1026]. There are 201 sets of CT volumes in this dataset, including 131 sets for training and 70 sets for the test.

The second dataset is from SLiver 07<sup>2</sup>, with different acquisition protocols. Its resolution in cross-section is [0.56, 0.8] mm, and that in z-axis is [1, 3] mm. Most of the images are pathological, including tumors, metastases, and cysts of different sizes. All datasets used contrast-enhanced agents in the central venous phase. The dataset provides 20 sets for training and ten sets for the test.

Considering the huge gap of different data in the z-axis direction in LiTS17 and limited by the capacity and computing capability of the training platform, this paper randomly takes out 16 volumes from the 20 sets as the training set, two volumes as to the validation set. Meanwhile, the test set has eight volumes with a z-axis resolution of [1, 3] mm in the LiTS17 and the remaining two volumes in Sliver07. Therefore, this paper collected a total of 28 sets of liver images, of which 16 sets were used for training, two sets were used for validation, and ten sets were used for the test.

All networks in this paper are built with Keras 2.4.0<sup>3</sup>. Adam is used as the optimization algorithm to find the optimal solution in the training process. Since the BN layer is used in the network, the initial learning rate is set to 30 times the learning rate in the original method, that is,  $lr = 0.03$ . The specific configuration of the platform used for training is Ubuntu 18.04, graphics card RTX2080Ti, memory 64 G, single CPU Intel Xeon Silver 4110.

### 4.2. Image preprocessing



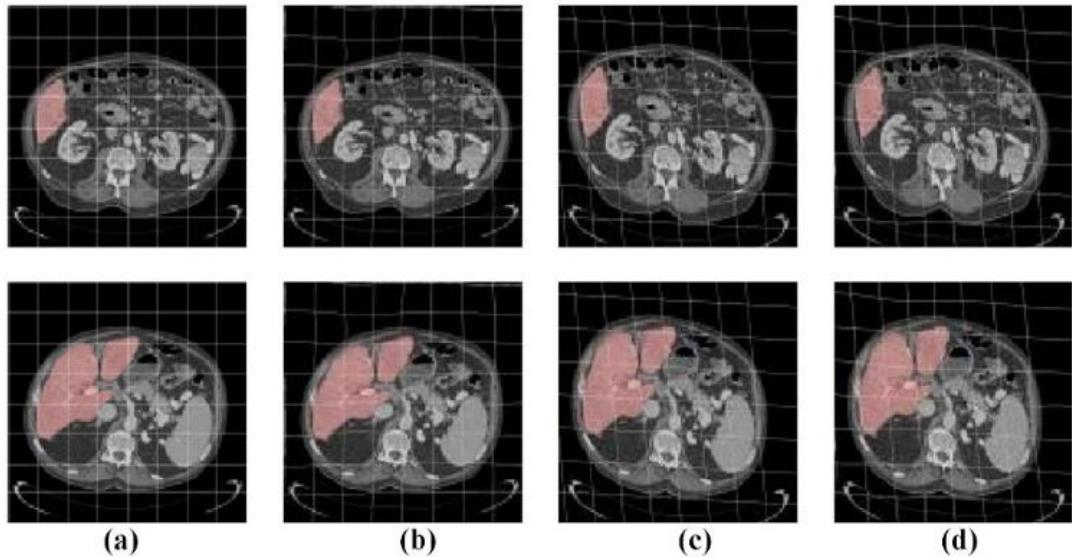
**Figure 5.** Illustration of 2D data augmentation.

1 <https://competitions.codalab.org/competitions/17094#results>

2 <http://www.sliver07.org/>

3 <https://github.com/keras-team/keras>

In the preprocessing stage, the Hounsfield intensity value is unified between [-200, 250] to remove irrelevant organs and details. At the same time, the intensity value of the image is normalized to [0, 1]. After that, random affine transformation and elastic deformation are performed on the training dataset, and then the amount of data is expanded ten times to prevent the model from overfitting. Figure 5 shows an illustration of the data augmentation process for a 2D image.



**Figure. 6** Illustration of 3D dataset augmentation (a) Original CT (b) Elastic deformation (c) Affine transformation (d) Elastic deformation with affine transformation.

For 2D CT images, data can be enhanced directly one by one, but for 3D data, certain modifications are required. The same random affine transformation matrix and elastic deformation matrix are used for data augmentation for all cross-sectional slices in a liver scan. The specific process is shown in Figure 6.

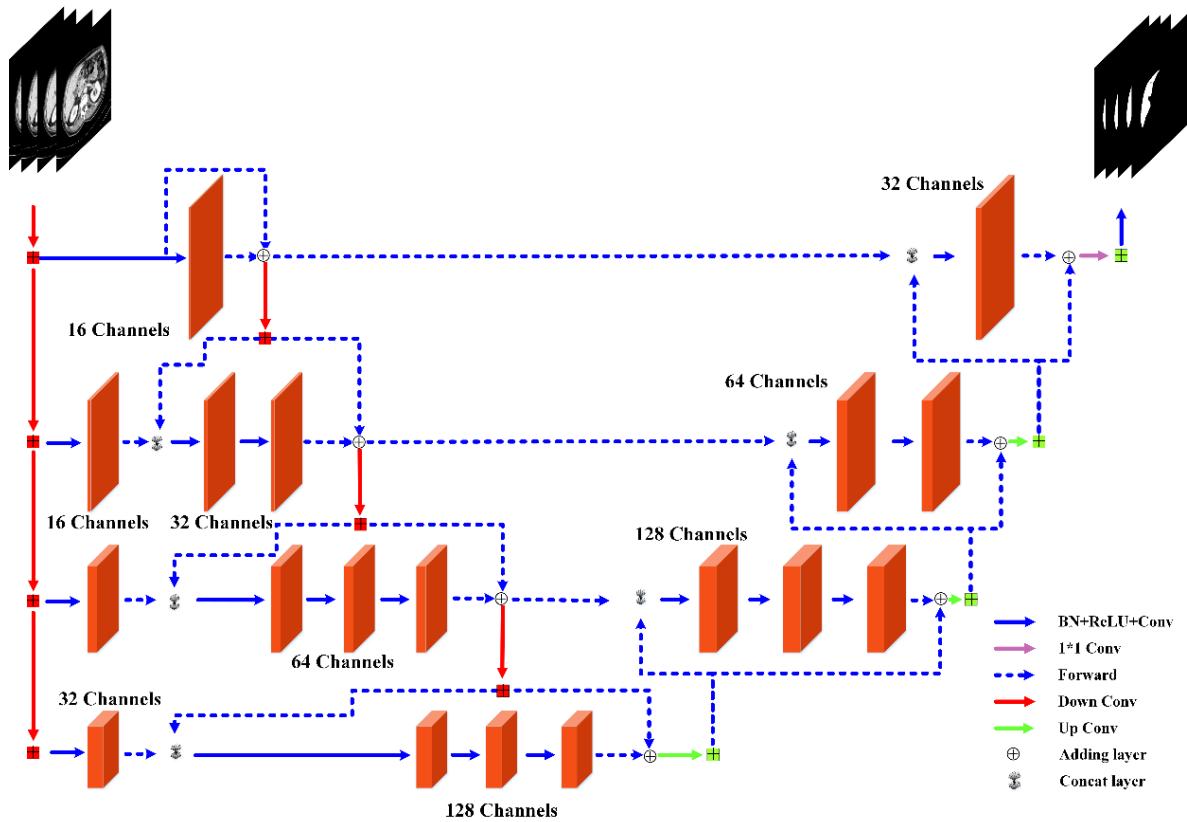
Figure 6(a) shows two slices from the same liver. To show the results of data augmentation more clearly, the liver is marked in red with a white grid; Figure 6(b) is the result using random elastic deformation. Figure 6(c) results from data augmentation using random affine transformation matrix; Figure 6(d) is the result of the combined effect of the elastic deformation and the affine transformation. By observing the deformation of the grid in Figure 6, it can be seen that the method can perform data augmentation on all slices in a liver scan and ensure the continuity of the 3D space. Therefore, this paper divides the data into  $512 \times 512 \times 64$  for training. The data is only divided into blocks on the z-axis but not on the x-axis and y-axis.

#### 4.3. Experimental results and discussion

In medical images, the original data is often three-dimensional. The segmentation algorithm for each slice in 2D space can only use the context information of the current slice, but not the spatial information between adjacent slices. Intuitively, using 3D iResU-Net to perform convolution would make good use of the 3D spatial information of the data and improve the model effect.

Although the method proposed in this paper is designed for 2D images, it can also be used to segment 3D images as long as 2D convolution is changed into 3D convolution. In order to enhance the

use of multi-scale information in the model and improve the segmentation effect of the model on 3D data, some improvements are made to the original network, as shown in Figure 7.

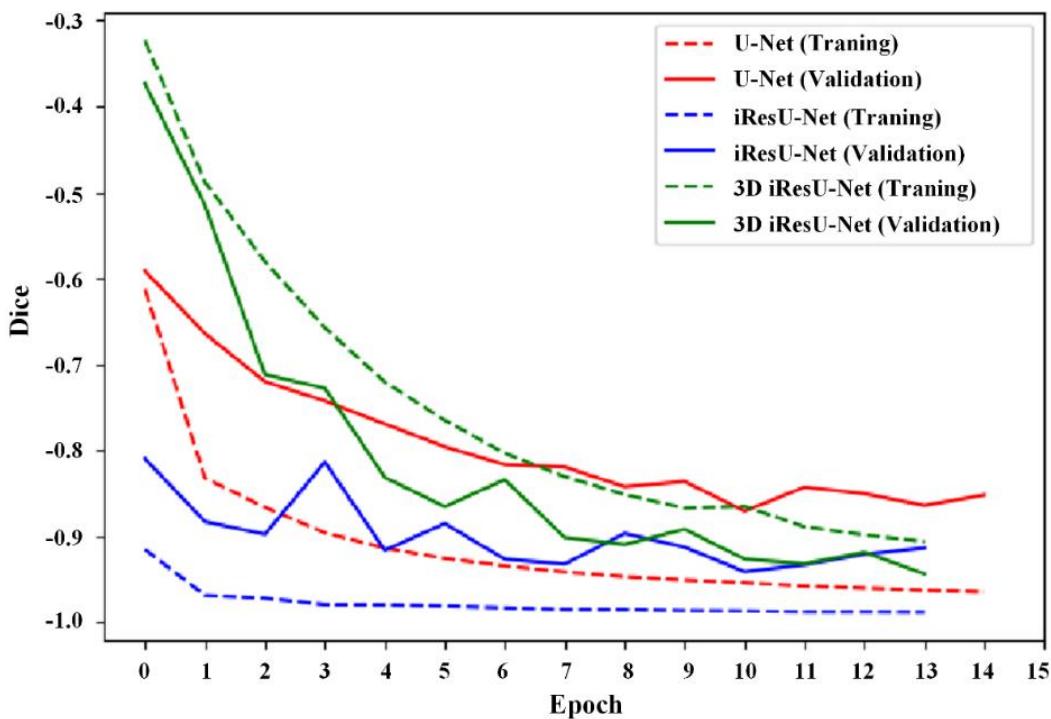


**Figure 7.** The framework of 3D iRes-U-Net.

This 3D network framework is different from the previous 2D network in four points. First of all, the input and output are all 3D data. Due to the limitation of the memory size, the data will be input in blocks. Secondly, all 2D convolutions are changed to 3D convolutions. Thirdly, to add multi-scale information to the network's input, the image is concatenated with the feature image of the previous layer after scaling and convolution. Then the convolution process of this layer is performed. Finally, to reduce the computational complexity, the input is down-sampled by a  $2 \times 2 \times 2$  maximum pooling and then input to the network. The output is up-sampled through a  $2 \times 2 \times 2$  bilinear as the final output. Therefore, the dimension of the input data of the network is  $512 \times 512 \times 64$ , and only the z-axis is partitioned, which largely preserves the spatial information and avoids partitioned processing.

#### 4.3.1. Comparison of loss function of different methods

The opposite number of the Dice coefficient is still used as the loss function in the loss function part. However, since it is challenging to implement opening and closing operations in 3D space, the weighted processing based on morphology is not used. Figure 8 shows the loss curves of different methods, including the ordinary U-Net, the proposed iResU-Net, and the 3D iResU-Net.



**Figure 8.** Loss curve of different methods on training data and validation data.

In Figure 8, the red dashed line represents the loss function curve of U-Net on the training set, and the solid red line represents the loss function curve of U-Net on the validation set. The blue dashed line represents iResU-Net on the training set, and the solid blue line represents the loss function curve of ResU-Net on the validation set. The green dashed line represents the loss function curve of 3D iResU-Net on the training set, and the solid green line represents the loss function curve 3D iResU-Net on the validation set.

In the training process, iResU-Net shows the fastest convergence speed, which is mainly thanks to two improvements: first, the BN layer in the network speeds up the convergence of the network, and second, the residual module is introduced into the network, which can not only be used to solve the problem of training difficulty of deeper network, but also enable an improvement on the shallow network, and accelerate the convergence of the network.

Through the loss function curve of three network tests, it can be seen clearly that compared with U-Net and 3D iResU-Net, iResU-Net performs best on the training set and shows superior generalization ability in the verification set.

In addition, the improved 3D iResU-Net for 3D data shows a good effect. To reduce the computational complexity of the model, the up-sampling and down-sampling operations on the input and output of the network make the model lose part of the information, resulting in slower convergence of the network. Besides, due to the shallower layers of the network, the model's generalization ability is greatly improved. Specifically, the performance of 3D iResU-Net on the verification set is higher than the training set and finally exceeds the segmentation effect of the U-Net.

Then, the segmentation effect of the model is further analyzed from the four metrics of VOE, RVD, ASD, and MSD, as shown in Table 1.

In Table 1, ResU-Net uses the cross-entropy loss function based on pixel category weighting.

iResU-Net uses the opposite number of Dice coefficients based on morphological weighting proposed in this paper.

**Table 1.** Summary of evaluations using four models on the test set.

Models	VOE (%)	RVD (%)	ASD (mm)	MSD (mm)
U-Net	15.39 $\pm$ 5.46	5.49 $\pm$ 4.24	5.48 $\pm$ 3.23	32.84 $\pm$ 5.98
ResU-Net	13.42 $\pm$ 3.68	-7.95 $\pm$ 3.04	3.58 $\pm$ 2.85	25.43 $\pm$ 4.21
iResU-Net	10.83 $\pm$ 3.70	-0.25 $\pm$ 2.74	2.12 $\pm$ 1.98	19.52 $\pm$ 4.01
3D iResU-Net	2.20 $\pm$ 12.46	15.73 $\pm$ 1.46	2.58 $\pm$ 2.12	13.76 $\pm$ 2.99

Comparing the experimental results of iResU-Net and U-Net, it can be seen that the improved model based on the residual module has brought a great improvement. iResU-Net outperforms U-Net in VOE, ASD, and MSD.

After the improvement, iResU-Net is significantly ahead of U-Net and ResU-Net in all metrics. Its Dice global value is more significant, and the mean value per case is larger with a most negligible variance, which indicates that iResU-Net achieves a superior segmentation effect both on the whole and on the individual. In addition, the mean and variance of VOE, RVD, ASD, and MSD are smaller, which means that it not only has a good segmentation effect, minor error, but also a more stable performance. Compared ResU-Net with iResU-Net, it can be found that the improved loss function proposed in this paper enabled a significant improvement on the basic U-Net model. In addition, compared with other networks, 3D iResU-Net showed the highest MSD score, which is because 3D convolution can make full use of the spatial information between slices.

**Table 2.** Summary of the parameters and reasoning time of the three models.

Models	Total number of parameters (M)	Trainable parameters (M)	Untrainable parameters (M)	Reasoning time (ms)
U-Net	1.851	0.563	0	46.73
iResU-Net	3.321	3.316	0.005	59.85
3D iResU-Net	0.564	0.563	0.001	8.46

Besides, this paper also compared the size and inference speed of the model, as shown in Table 2. There are two kinds of parameters in Table 2: trainable parameters and non-trainable parameters. There are mainly two types of non-trainable parameters in current neural networks.

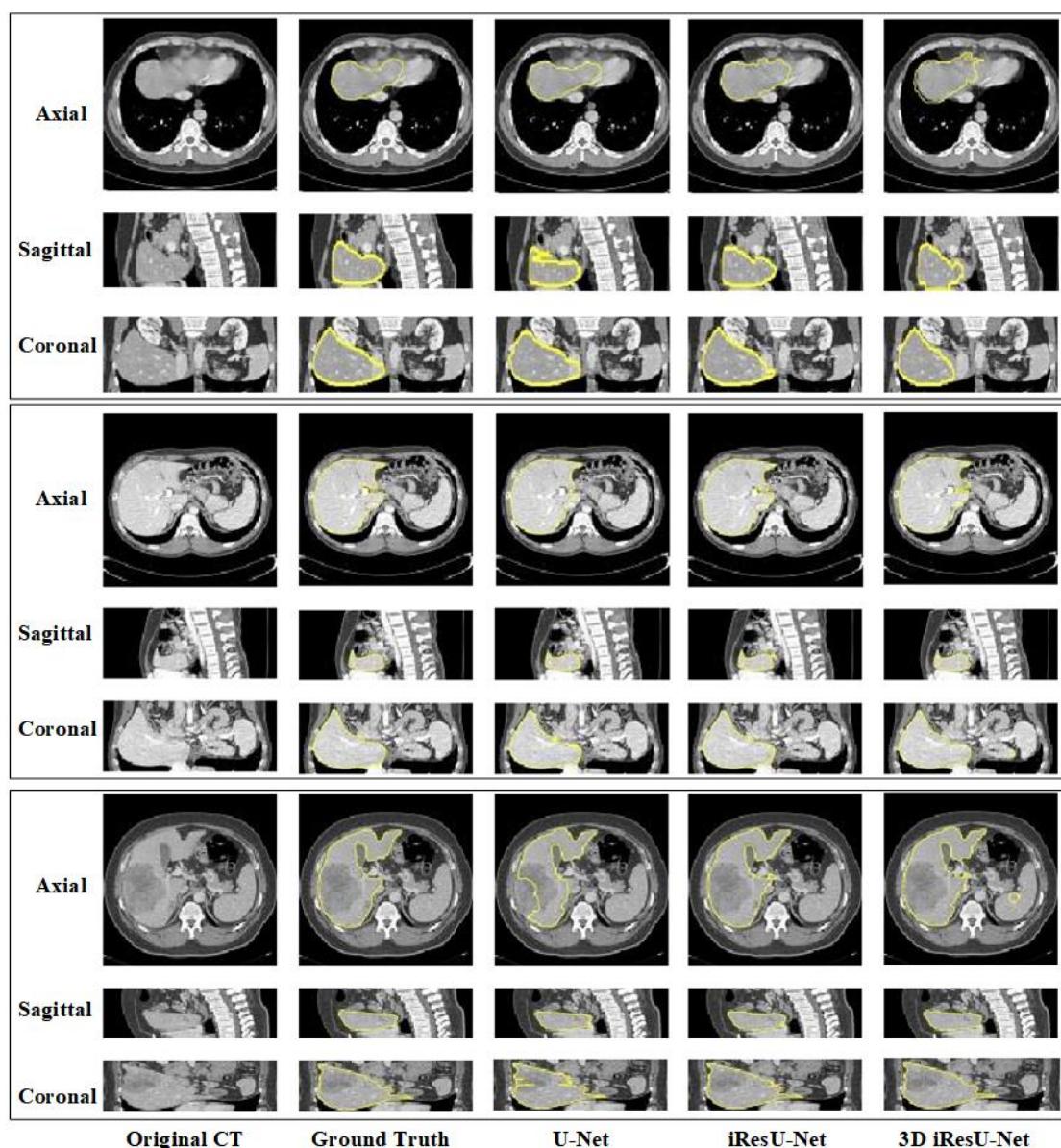
- 1) One is the parameters corresponding to the frozen layers in the network remain unchanged during the training process, which is often used in finetuning as a fixed feature extractor or unique image preprocessing.
- 2) The other is the statistical values in the BN layer, which would change with the mean and variance of the data, but are not directly affected by backpropagation.

It can be seen from Table 2 that, compared with U-Net, the total number of iResU-Net parameters is about 1.8 times that of U-Net, with higher complexity. Still, the convergence is faster, and the generalization ability is higher, indicating a significant improvement effect. However, the inference speed of iResU-Net is slower, about 1.28 times that of U-Net. Therefore, this improvement is more suitable for scenes that require fine segmentation but not high segmentation speed.

Specifically, it is necessary to analyze the experimental results of improved iResU-Net for 3D data. Only the test data in SLiver07 are used, mainly because the data in LiTS17 is quite different from SLiver07 in slice thickness, and the difference in data distribution between the two is noticeable. Due to the use of maximum pooling and up-sampling for input and output in 3D iResU-Net, the model's parameter amount and calculation amount are significantly reduced, and the model inference speed is faster. Still, it also limits the segmentation effect of 3D iResU-Net to a certain extent.

#### 4.3.2. Qualitative analysis on challenging cases

To verify the segmentation robustness of the network, tests were conducted on CT images in some difficult-to-segment cases, including liver CT with high noise, low contrast with neighboring organs, and pathological abnormalities. The segmentation results are shown in Figure 9.



**Figure 9.** Comparative results on challenging cases among U-Net, iResU-Net, and 3D iResU-Net.

The first three rows of images correspond to the low-contrast CT data of adjacent organs, the middle three rows correspond to the noisy CT data, and the last three rows correspond to the CT data with pathological abnormalities. For three rows of images from the same data, the first row is the axial plane that represents a cross-section of CT image, the second row is the sagittal plane that represents a sagittal plane of CT data, and the third row is the coronal plane that represents a coronal plane of CT data. In order to show the experimental results more clearly, we stretch the sagittal plane and coronal plane longitudinally, and the rest of the images are the same as the original size.

As shown in Figure 9, for the low-contrast CT image of adjacent organs, compared with the original U-Net, iResU-Net shows a better segmentation effect on the whole. Although some false positive segmentation results in the cross-plane, it provides superior results in the sagittal and coronal plane with a smooth edge. However, the 3D iResU-Net for the 3D dataset is not ideal on the three planes, but the segmentation effect in the sagittal plane is better than U-Net.

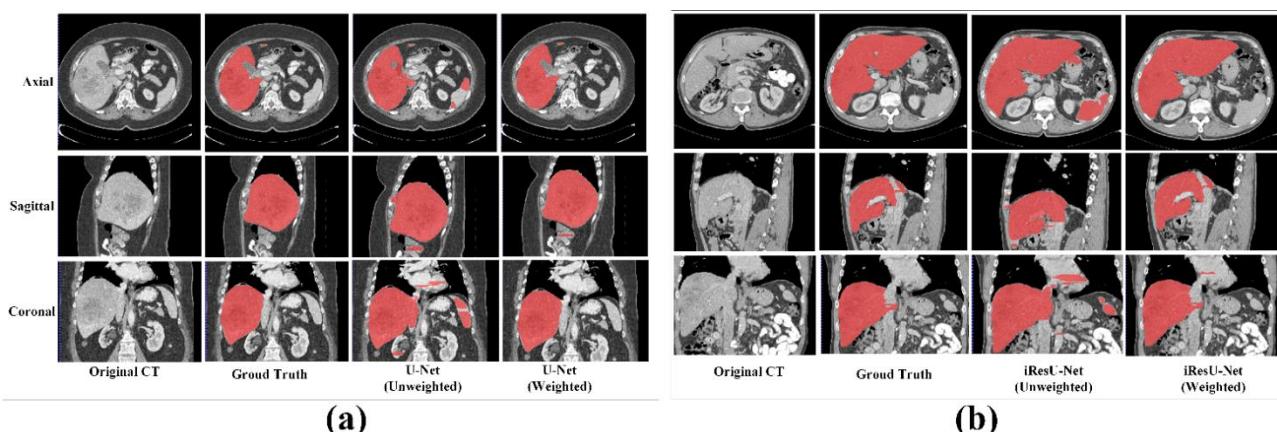
For CT data with pathological abnormalities, U-Net shows high false negatives at the lesion site and the contours of the liver, and there is almost no way to segment the liver completely. However, the proposed iResU-Net shows superior performance. Meanwhile, the 3D iResU-Net can also segment this type of data well, with some false positives and misjudgment.

#### 4.3.3. Impact of weighted heat map

In this section, we will verify the influence of the weighted heat map in the weighted loss function. U-Net and the iResU-Net proposed in this paper are used in the unweighted and weighted heat maps of Dice, respectively. Training, verification, and testing are performed on the inverse coefficient loss function.

**Table 3.** Comparative results of U-Net and iResU-Net on unweighted and weighted heat maps.

Models	VOE (%)	RVD (%)	ASD (mm)	MSD (mm)
U-Net (Unweighted)	18.65 $\pm$ 7.35	7.95 $\pm$ 5.65	7.54 $\pm$ 4.32	38.40 $\pm$ 6.35
U-Net (Weighted)	15.39 $\pm$ 5.46	5.49 $\pm$ 4.24	5.48 $\pm$ 3.23	32.84 $\pm$ 5.98
iResU-Net (Unweighted)	12.96 $\pm$ 3.95	1.25 $\pm$ 3.65	2.65 $\pm$ 2.85	21.89 $\pm$ 5.69
iResU-Net (Weighted)	10.83 $\pm$ 3.70	-0.25 $\pm$ 2.74	2.12 $\pm$ 1.98	19.52 $\pm$ 4.01



**Figure 10.** Comparative results of U-Net and iResU-Net on unweighted and weighted heat maps (a) unweighted (b) weighted.

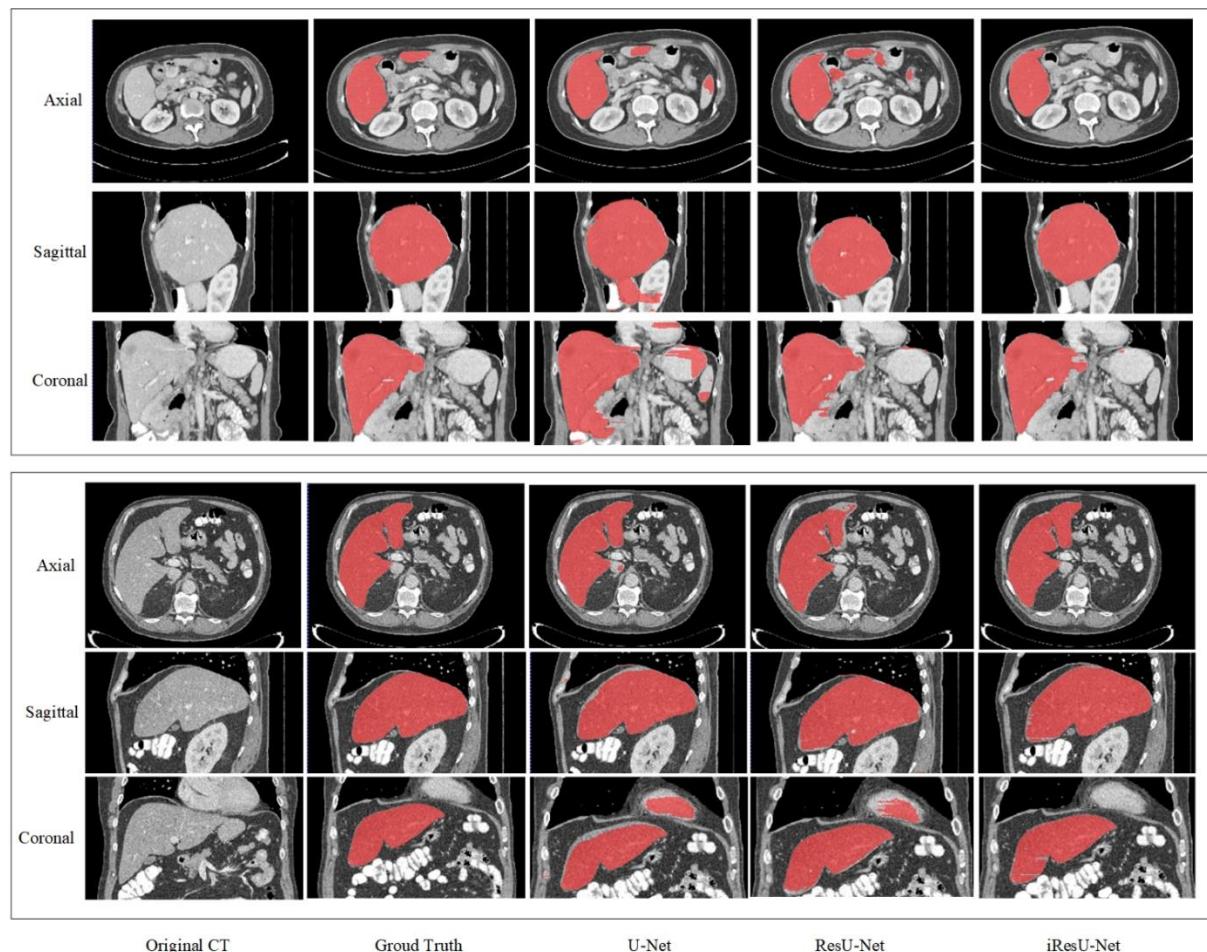
Table 3 lists the quantitative results of U-Net and iResU-Net on unweighted and weighted heat maps. It can be seen that when the weighted heat map is introduced, both U-Net and iResU-Net have obtained consistent performance improvement on all evaluation metrics. Moreover, Figure 10 shows the qualitative results. It can be seen that, compared to the methods without weighted heat maps, the approaches employing weighted heat maps reduce the segmentation errors effectively.

#### 4.3.4. Ablation study

In this section, we perform the ablation experiments to verify the effectiveness of our proposed iResU-Net. There are three models: (1) U-Net (2) U-Net with the residual block (ResU-Net). (3) U-Net combining residual block with BN layer (iResU-Net).

**Table 4.** Comparative results of ablation experiments.

Models	VOE (%)	RVD (%)	ASD (mm)	MSD (mm)
U-Net	15.39 $\pm$ 5.46	5.49 $\pm$ 4.24	5.48 $\pm$ 3.23	32.84 $\pm$ 5.98
ResU-Net	12.63 $\pm$ 4.21	1.58 $\pm$ 3.25	4.32 $\pm$ 2.56	26.25 $\pm$ 4.65
iResU-Net	10.83 $\pm$ 3.70	-0.25 $\pm$ 2.74	2.12 $\pm$ 1.98	19.52 $\pm$ 4.01



**Figure 11.** Comparative results of ablation experiments.

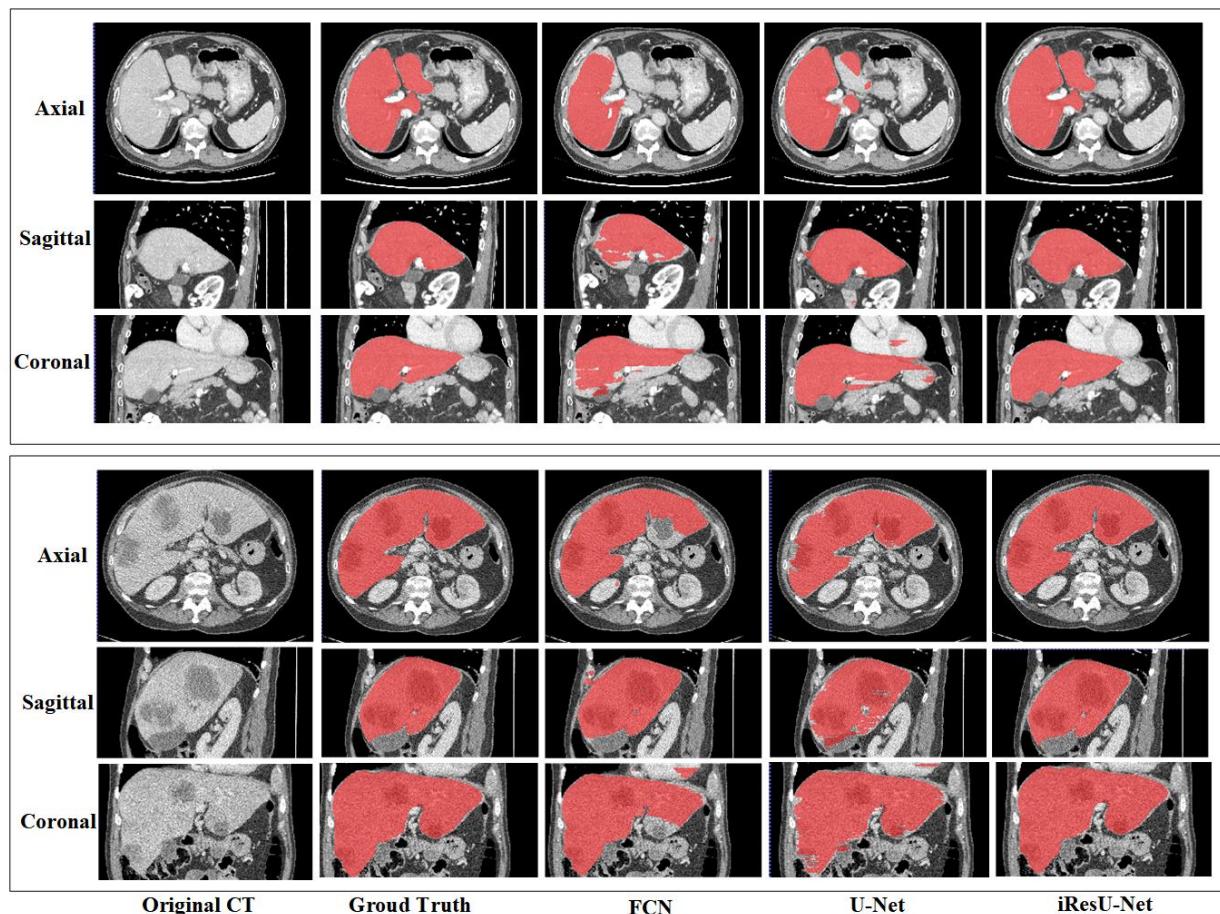
Table 4 shows the quantitative segmentation results. It can be seen that with the continuous improvement of the network structure, the segmentation effect of the network all obtains consistent performance improvement. Specifically, iResU-Net achieves the best performance compared with the other two models. In addition, Figure 11 shows the qualitative segmentation results. It can be seen that with the continuous improvement of the network, our proposed iResU-Net outperforms the other two models.

#### 4.3.5. Comparison with other methods

To further verify the effectiveness of our model, we compared the proposed method with two classic models, including the FCN and U-Net, with the same training and testing dataset.

**Table 5.** Comparative results with other methods.

Models	VOE (%)	RVD (%)	ASD (mm)	MSD (mm)
FCN	26.73 $\pm$ 8.25	9.58 $\pm$ 6.42	7.25 $\pm$ 6.32	41.25 $\pm$ 8.21
U-Net	15.39 $\pm$ 5.46	5.49 $\pm$ 4.24	5.48 $\pm$ 3.23	32.84 $\pm$ 5.98
iResU-Net	10.83 $\pm$ 3.70	-0.25 $\pm$ 2.74	2.12 $\pm$ 1.98	19.52 $\pm$ 4.01



**Figure 12.** Comparison of different methods on the difficult-to-segment cases.

Table 5 shows the results of the three models. It can be seen that our proposed model showed significant advantages compared to the other two methods. For example, the FCN obtained the lowest VOE, RVD, and MSD scores and delivered the most significant difference from the ground truth. On the contrary, the VOE, RVD, and MSD of our proposed iResU-Net have significantly improved compared with U-Net. Furthermore, Figure 12 shows a typical segmentation case of the three methods with difficult-to-segment cases. It can be seen that our proposed approach showed the least over-under-segmentation errors.

## 5. Conclusion

This paper proposes an improved iResU-Net for liver CT segmentation. Aiming at the shortcomings of U-Net in performance, the BN layer is introduced to remove the internal covariate shift in the network, to enhance the generalization ability of the model and speed up the model convergence; In addition, a residual module is introduced to speed up the convergence through the identity mapping. At the same time, the opposite number of the Dice coefficient is used to replace the original cross-entropy loss function to enhance the learning ability while suppressing the imbalance of pixels. Moreover, the weighting method based on morphology is proposed to force the model to refine the edge of the target. Finally, to verify the scalability of the proposed model on 3D data, the network's input is changed to multi-scale input to improve the performance. The experimental results show that the improved iResU-Net achieves a higher mean value and a minor standard deviation on each segmentation metric, proving good performance and robustness. It also delivers better results in various difficult-to-segment instances. In addition, the improved 3D iResU-Net segmentation effect for 3D data is better than U-Net, indicating that the model is highly scalable on 3D data. Nevertheless, the segmentation edge of 3D iResU-Net is relatively rough, mainly because the bilinear up-sampling method is used in the network's output. Therefore, in future research, other ways will be studied to refine the segmentation edge.

## Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (No. 61741106), the Heilongjiang Provincial Natural Science Foundation of China (No. LH2019F023), the Fundamental Research Foundation for Universities of Heilongjiang Province (No. LGYC2018JQ004), and the Heilongjiang Province University Student Innovation and Entrepreneurship Training Program Project (No. 202110214030).

## Conflict of Interest

The authors declare there is no conflict of interest in this study.

## References

1. A. Das, S. K. Sabut, Kernelized fuzzy c-means clustering with adaptive thresholding for segmenting liver tumors, *Procedia Comput. Sci.*, **92** (2016), 389–395. doi: 10.1016/j.procs.2016.07.395.

2. E. Göçeri, *A comparative evaluation for liver segmentation from SPIR images and a novel level set method using signed pressure force function*, Ph.D thesis, Izmir Institute of Technology, 2013.
3. E. Goceri, M. Z. Unlu, C. Guzelis, O. Dicle, An automatic level set based liver segmentation from MRI data sets, in *2012 3rd International conference on image processing theory, tools and applications (IPTA)*, (2012), 192–197. doi: 10.1109/IPTA.2012.6469551.
4. E. Dura, J. Domingo, E. Göçeri, L. M. Bonmati, A method for liver segmentation in perfusion MR images using probabilistic atlases and viscous reconstruction, *Pattern Anal. Appl.*, **21** (2018), 1083–1095. doi: 10.1007/s10044-017-0666-z.
5. E. Dura, J. Domingo, G. Ayala, L. M. Bonmati, E. Goceri, Probabilistic liver atlas construction, *Biomed. Eng. Online*, **15** (2017), 1–25. doi: 10.1186/s12938-016-0305-8.
6. S. Zhou, J. Wang, S. Zhang, Y. Liang, Y. Gong, Active contour model based on local and global intensity information for medical image segmentation, *Neurocomputing*, **186** (2016), 107–118. doi: 10.1016/j.neucom.2015.12.073.
7. J. Domingo, E. Dura, E. Göçeri, Iteratively learning a liver segmentation using probabilistic atlases: preliminary results, in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, (2016), 593–598. doi: 10.1109/ICMLA.2016.0104.
8. E. Göçeri, M. Z. Ünlü, O. Dicle, A comparative performance evaluation of various approaches for liver segmentation from SPIR images, *Turk. J. Electr. Eng. Comput. Sci.*, **23** (2015). 741–768. doi: 10.3906/elk-1304-36.
9. J. Tang, A multi-direction GVF snake for the segmentation of skin cancer images. *Pattern Recognit.*, **42** (2009), 1172–1179. doi: 10.1016/j.patcog.2008.09.007.
10. Z. Tu, Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering, in *Tenth IEEE International Conference on Computer Vision (ICCV'05)*, **1** (2005), 1589–1596. doi: 10.1109/ICCV.2005.194.
11. X. Ying, T. M. Monticello, Modern imaging technologies in toxicologic pathology: An overview, *Toxicol. Pathol.*, **34** (2006), 815–26. doi: 10.1080/01926230600918983.
12. E. Göçeri, Impact of deep learning and smartphone technologies in dermatology: Automated diagnosis, in *2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, (2020), 1–6. doi: 10.1080/01926230600918983.
13. E. Goceri, Capsnet topology to classify tumours from brain images and comparative evaluation, *IET Image Process.*, **14** (2020), 882–889. doi: 10.1049/iet-ipr.2019.0312.
14. T. Ojala, M. Pietikainen, D. Harwood, Performance evaluation of texture measures with classification based on Kullback discrimination of distributions, in *Proceedings of 12th international conference on pattern recognition*, **1** (1994), 582–585. doi: 10.1109/ICPR.1994.576366.
15. J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, (2015), 3431–3440. doi: 10.1109/CVPR.2015.7298965.
16. O. Ronneberger, P. Fischer, T. Brox, U-Net: convolutional networks for biomedical image segmentation, in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, Cham, **9351** (2015), 234–241. doi: 10.1007/978-3-319-24574-4\_28.
17. A. Ben-Cohen, I. Diamant, E. Klang, M. Amitai, H. Greenspan, Fully convolutional network for liver segmentation and lesions detection, in *Deep Learning and Data Labeling for Medical Applications*, Springer, Cham, **10008** (2016), 77–85. doi: 10.1007/978-3-319-46976-8\_9.

18. Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, O. Ronneberger, 3D U-Net: learning dense volumetric segmentation from sparse annotation, in *International conference on medical image computing and computer-assisted intervention*, Springer, Cham, **9901** (2016), 424–432. doi: 10.1007/978-3-319-46723-8\_49.
19. Q. Dou, H. Chen, Y. Jin, L. Yu, J. Qin, P. A. Heng, 3D deeply supervised network for automatic liver segmentation from CT volumes, in *International conference on medical image computing and computer-assisted intervention*. Springer, Cham, **9901** (2016), 149–157. doi: 10.1007/978-3-319-46723-8\_18.
20. C. Zhao, Y. Xu, H. Zhou, J. Tang, Y. Zhang, J. Han, et al., Lung Segmentation and Automatic detection of COVID-19 using radiomic features from chest CT images, *Pattern Recognit.*, **119** (2021), 108071. doi: 10.1016/j.patcog.2021.108071.
21. C. Zhao, A. Vij, S. Malhotra, J. Tang, H. Tang, D. Pienta, et al., Automatic extraction and stenosis evaluation of coronary arteries in invasive coronary angiograms, *Comput. Biol. and Med.*, **136** (2021), 104667. doi: 10.1016/j.compbio.2021.104667.
22. A. Hoogi, A. Subramaniam, R. Veerapaneni, D. L. Rubin, Adaptive estimation of active contour parameters using convolutional neural networks and texture analysis, in *IEEE Transactions on Medical Imaging*, **36** (2017), 781–791. doi: 10.1109/TMI.2016.2628084.
23. P. Hu, F. Wu, J. Peng, P. Liang, D. Kong, Automatic 3D liver segmentation based on deep learning and globally optimized surface evolution, *Phys. Med. Biol.*, **61** (2016), 8676–8698. doi: 10.1088/1361-6560/61/24/8676.
24. F. Lu, F. Wu, P. Hu, Z. Peng, D. Kong, Automatic 3D liver location and segmentation via convolutional neural network and graph cut, *Int. J. CARS*, **12** (2017), 171–182. doi: 10.1007/s11548-016-1467-3.
25. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna, Rethinking the inception architecture for computer vision, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), 2818–2826. doi: 10.1109/CVPR.2016.308.
26. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), 770–778.
27. F. Milletari, N. Navab, S. A. Ahmadi, V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation, in *2016 Fourth International Conference on 3D Vision (3DV)*, (2016), 565–571. doi: 10.1109/3DV.2016.79.
28. Q. Jin, Z. Meng, C. Sun, H. Cui, R. Su, RA-UNet: A hybrid deep attention-aware network to extract liver and tumor in CT scans, *Front. Bioeng. Biotechnol.*, 2020. doi: 10.3389/fbioe.2020.605132.
29. P. F. Christ, M. E. A. Elshaer, F. Ettlinger, S. Tatavarty, M. Bickel, P. Bilic, et al., Automatic liver and lesion segmentation in CT using cascaded fully convolutional neural networks and 3D conditional random fields, in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, Cham, **9901** (2016), 415–423. doi: 10.1007/978-3-319-46723-8\_48.
30. S. S. M. Salehi, D. Erdogmus, A. Gholipour, Tversky loss function for image segmentation using 3D fully convolutional deep network, in *International Workshop on Machine Learning in Medical Imaging*, Switzerland, Springer, (2017), 379–387. doi: 10.1007/978-3-319-67389-9\_44.
31. E. Goceri, Diagnosis of skin diseases in the era of deep learning and mobile technology, *Comput. Biol. Med.*, **134** (2021), 104458. doi: 10.1016/j.compbio.2021.104458.

32. E. Goceri, Deep learning based classification of facial dermatological disorders, *Comput. Biol. Med.*, **128** (2021), 104118. doi: 10.1016/j.combiomed.2020.104118.



AIMS Press

©2022 author name, licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)