Abdominal Multi-Organ Segmentation Using Multi-Scale and Context-Aware Neural Networks

Yuhan Song, Armagan Elibol, [1] **Nak Young Chong** [2]

School of Information Science, Japan Advanced Institute of Science and Technology, Japan 923-1292 (e-mail: yuhan-s@jaist.ac.jp, aelibol@jaist.ac.jp, nakyoung@jaist.ac.jp)

Abstract: Recent advancements in AI have significantly enhanced smart diagnostic methods, bringing us closer to achieving end-to-end diagnosis. Ultrasound image segmentation plays a crucial role in this diagnostic process. An accurate and robust segmentation model accelerates the process and reduces the burden of sonographers. In contrast to previous research, we consider two inherent features of ultrasound images: (1) different organs and tissues vary in spatial sizes, and (2) the anatomical structures inside the human body form a relatively constant spatial relationship. Based on those two ideas, we proposed two segmentation models combining multi-scale convolution neural network backbones and a spatial context feature extractor. We discuss two backbone structures to extract anatomical structures of different scales: the Feature Pyramid Network(FPN) backbone and the Trident Network backbone. Moreover, we show how Spatial Recurrent Neural Network(SRNN) is implemented to extract the spatial context features in abdominal ultrasound images. Our proposed model has achieved dice coefficient score of 0.919 and 0.931, respectively.

Keywords: Medical Imaging and Processing, Diagnostic Ultrasound, Image Segmentation, Feature Pyramid Network, Trident Network

1. INTRODUCTION

1.1 Motivation

As many countries face the challenges of population aging with healthcare staff shortages, the demand for remote patient monitoring drives the development of AI-assisted diagnosis. In clinical practice, ultrasound imaging is one of the most common imaging modalities due to its effectiveness, non-invasive, and non-radiation nature. Medical ultrasound imaging requires an accurate delineation or segmentation of different anatomical structures for various purposes, like guiding the interventions. However, compared with other modalities, it is relatively harder to process because of low contrast, acoustic shadows, and speckles, to name a few (Almajalid et al. (2018)). It can be challenging even for experienced sonographers to detect the exact contour of tissues and organs. It usually takes years of study and practice to train a qualified sonographer. Therefore, an automated and robust ultrasound image segmentation method is expected to help with locating and measuring important clinical information. Along these lines, we are developing a control algorithm for the robot arm to perform automatic ultrasound scans (see Fig. 1). Because this system is expected to operate automatically without human intervention, an evaluation metric for the robot's movement is necessary. To this end, a segmentation algorithm needs to be incorporated into the robot trajectory control system.

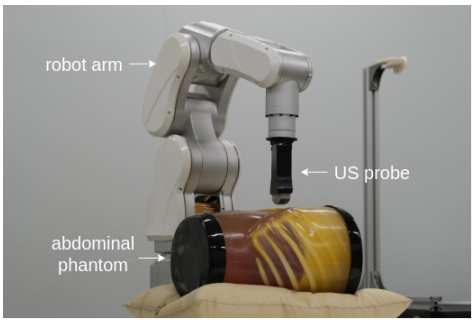


Fig. 1. Our robot-assisted ultrasound imaging system

1.2 Traditional Methods

Traditional ultrasound image segmentation methods focus on the detection of textures and boundaries based on morphological or statistical methods. Mishra et al. (2003) proposed an active contour solution using low pass filters and morphological operations to predict the cardiac contour. Mignotte et al. (2001) developed a boundary estimation algorithm based on a Bayesian framework, where the estimation problem was formulated as an optimization algorithm to maximize the posterior possibility of being a boundary. Previously Mignotte and Meunier (2001) used statistical external energy in a discrete activate contour

[1] The author is currently with IAS-8: Data Analytics and Machine Learning, Institute for Advanced Simulation, Forschungszentrum Jülich, 52428 Jülich, Germany

[2] Corresponding author

for the segmentation of short-axis parasternal images, in which a shifted Rayleigh distribution was used to model gray-level statistics. Boukerroui et al. (2003) also proposed a Bayesian framework to conduct robust and adaptive region segmentation, taking the local class mean with a slow spatial variation into consideration to compensate for the nonuniformity of ultrasound echo signals.

1.3 Learning-based Methods

Besides those vulnerabilities mentioned above, traditional ultrasound image segmentation methods are time-consuming and prone to irregular anatomical structure shapes, and usually requires manual initialization operations. Compared with morphological and statistical methods, convolutional neural network (CNN) based solutions are powerful and flexible because of their strong nonlinear learning ability. Zhang et al. (2016) conducted coarse and fine lymph node segmentation based on two series-connected fully convolutional networks. Huang et al. (2021) proposed a modified deep residual U-Net model to predict the contour of abdominal organs and tissues. They train their model initially on a tendon dataset, then fine-tune it on a breast tumor dataset. After getting a pre-trained model, they adapt the model to detect different anatomical structures using transfer learning. Lei et al. (2021) proposed a male pelvic multi-organ segmentation method on transrectal ultrasound images. In their research, a fully convolutional one-state (FCOS) object detector originally designed for generalized object detection is adapted for ultrasound image segmentation.

1.4 Research Contributions

In the context of abdominal ultrasound image segmentation, most of the existing methods are targeted at specific organs or anomalies. Chen et al. (2022) designed a multi-scale and deep-supervised CNN architecture for kidney image segmentation. They implemented a multi-scale input pyramid structure to capture features at different scales and developed a multi-output supervision module to enable the network to predict segmentation results from multi-scales. Huang et al. (2019) developed a detection algorithm for pulmonary nodules based on deep three-dimensional CNNs and ensemble learning. However, the importance of multi-organ segmentation is still ignored. On one hand, the segmentation result can be used as a guide for the remote ultrasound scan system, which is the cornerstone of realizing an automatic remote diagnostic system. An automatic diagnostic system will reduce burdens on sonographers, enabling them to concentrate on the analysis of pathology. On the other hand, abdominal organ segmentation can also provide information on specific organs and tissues, which can be used to assist in the diagnosis of certain diseases. Therefore, in this paper, an abdominal multi-organ segmentation method is proposed. Our contribution can be listed as follows: (a) We proposed multi-organ segmentation methods based on two multiple scale structures, Feature Pyramid Network, and Trident Network; (b) We combined these two backbones respectively with an SRNN module, which helps improve the performance significantly by bringing spatial context information. The rest of the paper is organized as follows:

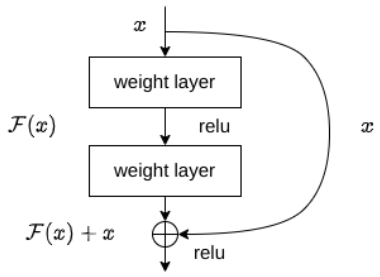


Fig. 2. Shortcut connection in ResNet

In section 2, we introduced the primary related works that have made up of our proposed methodology. In section 3, the illustrations and explanations of our proposed network structures have been given. In section 4, we tested the performance of the designed model, especially we focused on how much improvement the SRNN module has brought in. In section 5, we draw a conclusion that our proposed abdominal multi-organ segmentation method has achieved promising results.

2. RELATED WORK

2.1 ResNet

We use ResNet (He et al. (2016)) as the feature extractor for both of the proposed networks. ResNet is a deep residual learning framework to solve the degradation problem of deep networks. A common issue in deep learning is that deeper neural networks are harder to train. With the layers going deeper, accuracy would drop rapidly. In other words, appending more layers to a suitably deep model will increase the training error. He et al. (2016) addressed the degradation problem by reformulating some of the layers as learning residual functions with reference to the layer inputs, instead of learning unreferenced functions.

ResNet makes use of feedforward networks with "shortcut connections", which makes the network easier to optimize and able to gain extra accuracy from considerably increased depth. In the aforementioned paper, the authors let the shortcut connections simply perform identity mapping and produce the sum of the output from the original layers and the lateral layers as illustrated in Fig. 2, where x represents the input, $F(x)$ the abstract representation of the residual block, and ReLU the activation function. x is passed directly to the output and called "identity shortcut connection". Inserting the shortcut connection to the plain backbone, they managed to train models with over 1000 layers. ResNet's capacity to extract deep features made it possible in our work to combine both semantic and abstract information for analyzing ultrasound images. in this paper, we build our network models on the backbone of ResNet-101.

2.2 Feature Pyramid Network

In the abdominal section, different anatomical structures vary in shape and size, which may cause class imbalance for the traditional CNN segmentation algorithms like U-Net in Ronneberger et al. (2015). Although the total amount of instances may be almost equal in training,

(a) Ultrasound image (b) Ground-truth

Fig. 3. Example of class imbalance problem

(a) Segmentation result (b) Ground-truth

Fig. 4. U-Net segmentation example

the relatively large organs and tissues occupy many more pixels in the ultrasound images. As shown in Fig. 3, the violet part is the liver which occupies most of the pixels, and the green part is the gallbladder. This will make the algorithm classify as many pixels into the liver as possible since a majority class has a much bigger influence on the final score than the minority class. Fig. 4 shows the segmentation result of an ultrasound image containing liver (violet) and kidney (yellow). Compared with the ground truth, the result tends to ignore the kidney to focus on drawing the true mask of the liver. This example illustrates the necessity of introducing multi-scale object detection frameworks, FPN structure for example(Lin et al. (2017)).

FPN takes the feature maps from multiple layers of the encoder rather than only from the deepest output. This pyramid network structure is scale-invariant in the sense that an object's scale changes with shifting its level in the feature pyramid. In other words, smaller objects are usually easier to be detected from smaller yet deeper feature maps, and vice versa. Compared with other pyramid network structures, FPN not only utilizes the relation between scale and layer depth but also uses a top-down pathway to construct higher-resolution layers from a semantic layer. This solves the problem that feature maps composed of low-level structures (closer to the original level) are too naive for accurate object detection. As the reconstructed layers are semantically strong, but the locations of objects are not precise after all the down-sampling and up-sampling, the authors then added lateral connections between reconstructed layers and the corresponding feature maps to help the decoder predict the locations better. The overall structure of FPN is illustrated in Fig. 5.

2.3 Trident Network

Another network architecture designed to address the challenge brought by scale variation is the multi-branch network structures. Li et al. (2019) proposed a trident network structure for object detection. Instead of employing multi-scale inputs, as seen in methods like FPN, the

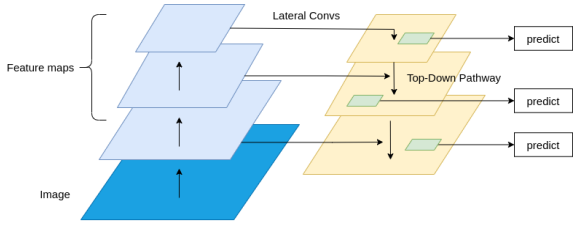


Fig. 5. Feature pyramid network structure

trident network structure modifies the original semantic feature extractor by generating scale-specific feature maps. The trident network structures make use of dilated convolutions, as originally proposed by Yu and Koltun (2015). Fig. 6 shows the fundamental structure of the trident network in the work of Li et al. (2019). The Trident network utilizes a parallel multi-branch architecture, where the Trident branches share the same parameters while extracting features specific to their receptive fields. By doing this, the Trident block allows the network to capture objects at different scales by processing the input image at multiple resolutions simultaneously. Each branch in the trident block focuses on a specific scale range, enabling the network to specialize in detecting objects of different sizes.

In the case of employing ResNet as a feature extractor, consider a single residual block configured in the bottleneck style, as described by He et al. (2016). This block consists of three convolutional layers with kernel sizes of 1×1, 3×3, and 1×1. To establish an equivalent trident block, we create multiple parallel residual blocks, each with varying dilation rates applied to the 3×3 convolutional layers. The arrangement of these trident blocks allows for effective control over the receptive fields of distinct branches. We substitute the blocks in the res4 stage of the backbone network with trident blocks. This substitution selection is based on the motivation that larger strides introduce a substantial divergence in receptive fields, aligning with our specific requirements.

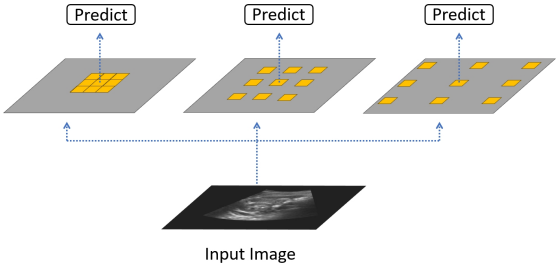


Fig. 6. Trident network structure

2.4 Spatial Recurrent Neural Network

Another important property of ultrasound images is that the anatomical structures form a constant spatial relationship under the same scan pattern. For example, in the mid-sagittal plane scanning, the mesenteric artery is usually at the bottom of the liver, and the pancreas is located at the side of the liver. Experienced sonographers rely heavily on such spatial context information to locate the target organs. However, those spatial correlation patterns are not absolutely constant regarding different scan positions and angles from the start point of different medical interests.

This prior knowledge inspired us to take spatial context information into consideration.

To extract context information, the spatial recurrent neural network (SRNN) is introduced. Many studies have explored the utilization of RNNs to gather contextual information. Schuster and Paliwal (1997) proposed a bidirectional recurrent neural network (BRNN) that passes both forward and backward across a time map to ensure the information is propagated across the entire timeline. When it comes to the context of spatial information, Graves and Schmidhuber (2008) proposed a multi-dimensional RNN to recognize handwriting. Byeon et al. (2015) built a long short-term memory (LSTM) RNN structure for scene labeling.

Bell et al. (2016) proposed an object detection network structure called Inside-Outside Net (ION). Besides taking the information near an object's region of interest, the introduction of contextual information has improved the performance, for which a module of four directional RNNs is implemented. Fig. 7 shows the propagation of the RNNs. The structures are placed laterally across the feature maps and move independently in four cardinal directions: right, left, down, and up. The outputs from the RNNs are then concatenated and computed as a feature map containing both local and global contextual information.

3. MULTI ORGAN SEGMENTATION NETWORK

3.1 Problem Definition

Given an input abdominal ultrasound image, I, the objective is to segment it into different regions corresponding to distinct abdominal organs. Let S denote the segmented image, where each pixel S_{ij} corresponds to a label indicating the organ class or background.

3.2 Mathematical Formulation

Input : An image I represented as a matrix of pixel intensities.

Output : A segmented image S, where $S_{ij} \in \{0, 1, 2, \ldots, N\}$, and each number corresponds to a different organ class or background (0 for background, 1 for the first organ, etc.). In this research context, N equals to 5.

Objective : Maximize the accuracy of segmentation. This can be represented as maximizing the dice coefficient score D (defined in equation. 8), which measures the similarity between the predicted segmentation S and the ground truth labels.

3.3 Network Structure Overview

Fig. 8 and Fig. 9 are illustrating the structure of our proposed models.

Feature Pyramid Stucture On the left side is the ResNet-101 backbone as the feature extractor. The input image is propagated from bottom to top, with the network generating feature maps of lower resolution and richer semantic information. We define the layers producing feature maps of the same size as one stage. We choose the output of the last layer of each stage to represent the output of the entire stage except the shallowest stage because it is computationally time-consuming and of a low semantic feature. Each of the blue cubes represents an output of the stage called {res2, res3, res4, res5}, respectively. The feature maps go separately through a 1x1 convolution layer and the SRNN module.

The green cubes represent the feature maps after the convolution operation, and the red cubes represent the context feature maps. The deep feature map is concatenated with the context feature map, then it will be concatenated with the spatial context feature map. The concatenated feature map will go through a normalization operation and be compressed to reduce depth channels. The semantic feature map from the upper layer, spatially coarser but semantically stronger, is upsampled by a scale factor of 2. The upsampled feature map from the upper pyramid level and the feature map from the current pyramid level are added together as the new feature map. Additionally, we introduce an important improvement for this FPN structure from another work of ours(Song et al. (2023)). Similarly with semantic features, the spatial context information is coarser at the deep level and more precise from those high-resolution levels. Thus, we also build a bottom-up pathway(red links) to deliver precise spatial context information to the higher levels. The new semantic feature maps and spatial context feature maps will be concatenated together as the final output(yellow cubes).

Trident Network Structure In the trident network, the input image will first be propagated through the ResNet-101 backbone stages. The stem, res2, and res3 stages are consistent with the original ResNet design. Starting from the res4 stage, the normal bottleneck blocks are replaced with Trident blocks. The feature maps will then be fed into 3 branches of convolution layers. Later, the feature maps will be concatenated at the output layer of the last bottleneck block in the res4 stage.

After getting the concatenated feature map, it will be fed into the SRNN module to generate a spatial context feature map. Then the original features and spatial context features will again go through a concatenation and convolution stage then become the final output. The final proposals or detections from multiple branches will be combined using Non-maximum Suppression. Again the RPN, Fast-RCNN, and Mask-RCNN heads are omitted and these weights are shared among the branches.

RPN and R-CNNs After extracting semantic and spatial features, these pyramid feature maps are then sent to region proposal networks (RPN), region-based detectors (Fast R-CNN) and mask heads (Mask R-CNN). Unlike the classic object detectors, the FPN attaches RPN and Fast R-CNN to each of the output layers. The parameters of the heads are shared across all feature pyramid levels for simplicity, but the accuracy is actually very close with or without sharing parameters (Lin et al. (2017)). This is indirect proof that all the levels of the pyramid share similar semantic levels. The structure of proposers and anchor/mask generators are omitted in the graphs since they are not our main interest.

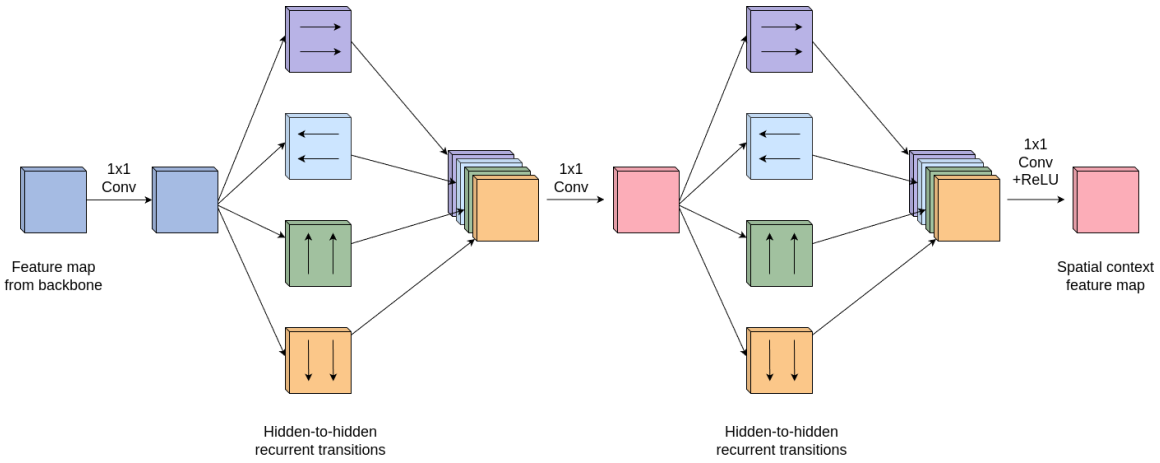


Fig. 7. Spatial RNN module

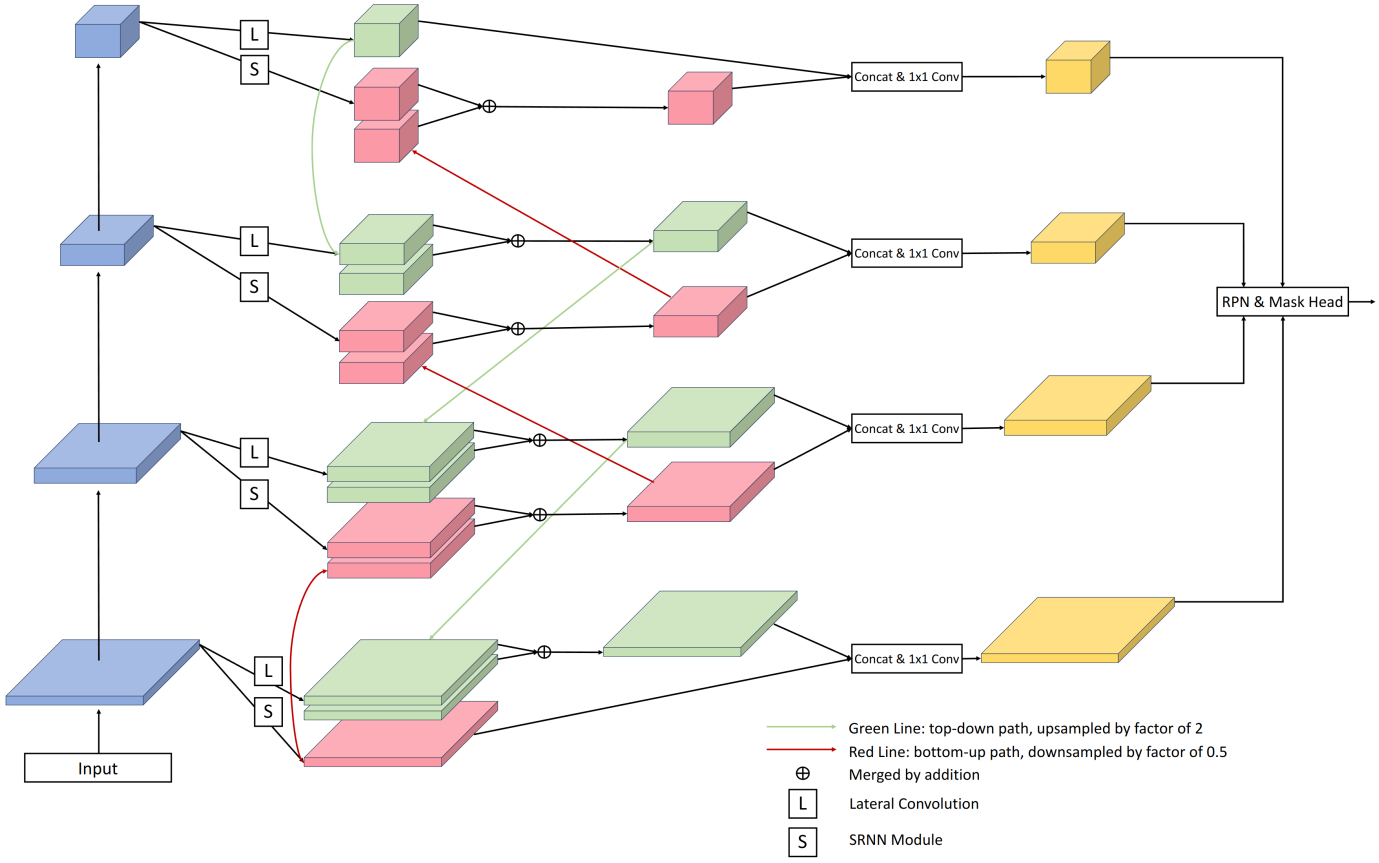


Fig. 8. Proposed network structure 1 based on feature pyramid network

3.4 SRNN Structure

The SRNN module follows the idea of the ION network structure. Fig. 10 shows how the RNNs extract the contextual information. We first perform a 1x1 convolution to simulate the input-to-hidden data translation in the RNN. Then, four RNNs are propagated through the different directions mentioned above. The outputs from the RNNs are fused into an intermediate feature map. Until this step, each pixel contains the context information aiming at its four principal directions: right, left, up, and down. Another round of the process is then repeated to further propagate the aggregated spatial context information in each principal direction. Finally, a feature map containing the overall context information is generated. For comparison, in the feature map on the left in Fig. 10, each pixel only contains information about itself and its neighbors (depending on the perspective field). After the first round of RNN propagation, the pixels get the context information from its 4 directions. Finally, RNNs propagate through the context-rich pixels to extract the full-directional context information. Therefore, the last feature map is globally context-rich.

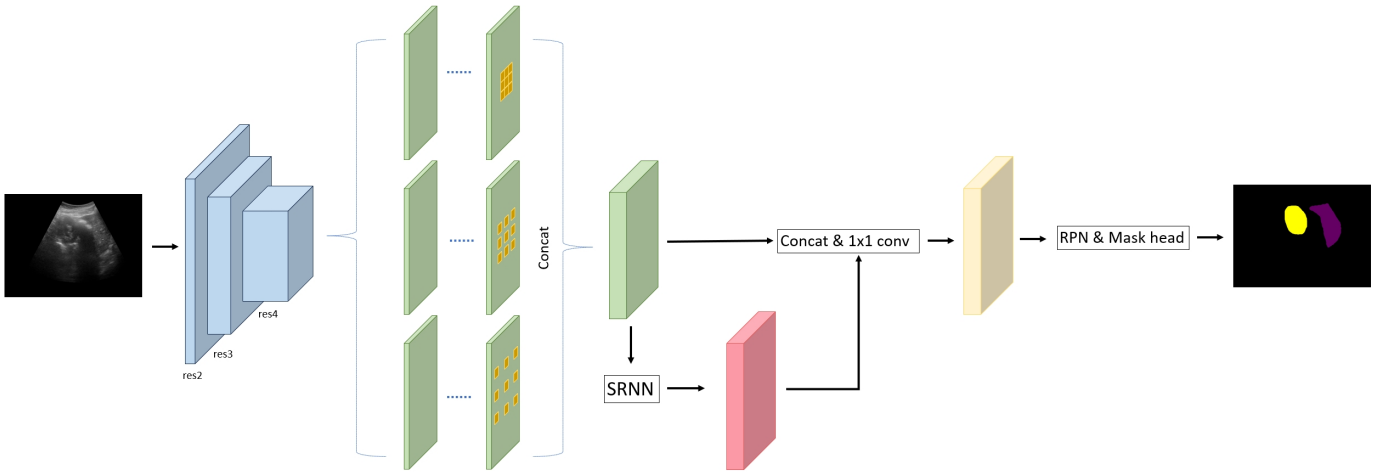


Fig. 9. Proposed network structure 2 based on trident network

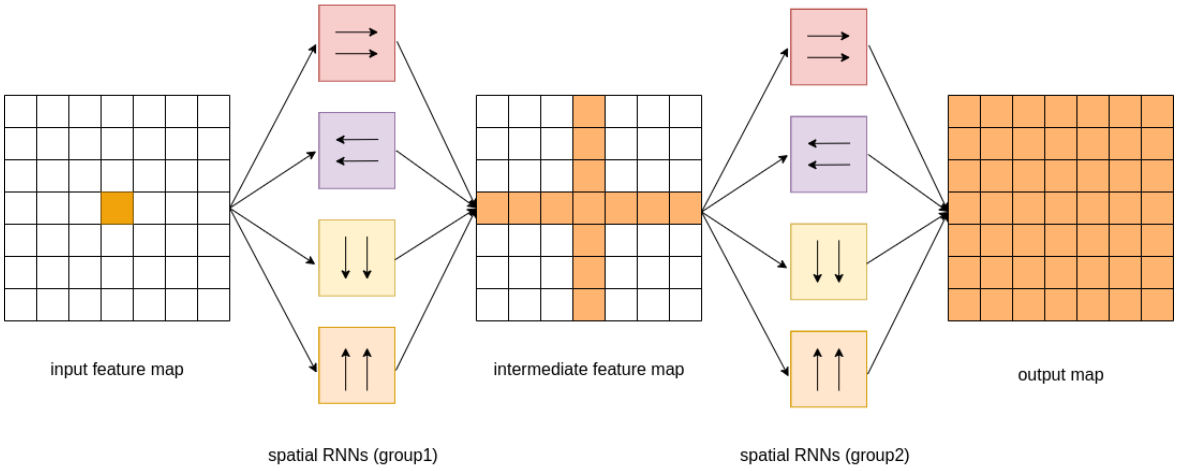


Fig. 10. Illustration of the IRNN propagation

3.5 IRNN

An RNN is specialized for processing sequential data. The data fed into the input nodes is propagated through the hidden nodes, updating the internal states using past and present data. There are variants of RNN such as gated recurrent units (Cho et al. (2014)), LSTM (Hochreiter and Schmidhuber (1997)), and plain tanh RNNs. The RNN in this work follows the model in Le et al. (2015) due to its efficiency and simplicity of training. This RNN structure is called IRNN, because the recurrent weight matrix is initialized to the identity matrix. IRNN has a good performance for long-range data dependencies (Bell et al. (2016)). IRNN is composed of the rectified linear unit, and the recurrent weight matrix is initialized to the identity matrix. Therefore, gradients are propagated backward with full strength at initialization. We adopt four independent IRNNs that propagate through four different directions. Given below is the update function for the IRNN moving from left to right. The rest IRNNs follow a similar equation according to the propagation direction:

$$h_{i,j}^{right} \leftarrow max(W_{hh}^{right} h_{i,j-1}^{right} + h_{i,j}^{right}, 0), \qquad (1)$$

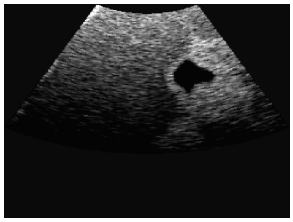
where W is the hidden transition matrix and $h_{i,j}$ is the feature at pixel(i, j).

Each direction on independent rows or columns is computed in parallel, and the output from the IRNN is computed by concatenating the hidden state from the four directions at each spatial location.

4. EXPERIMENTS

4.1 Dataset

A dataset of high quality is one of the key factors to train a neural network. Unfortunately, there are few open-source abdominal ultrasound image datasets. Most of the relevant researchers have not made their datasets publicly for the protection of patients' privacy. In this work, we use the dataset provided by Vitale et al. (2019) containing both artificial ultrasound images which are translated from CT images, and a few images generated from real ultrasound scans. In the work of Vitale et al. (2019), they applied generative neural networks trained with a cycle consistency loss, and successfully improved the realism in ultrasound simulation from computed tomography (CT). We use 926



(a) CT-generated image

(b) Regular image

Fig. 11. Example of images in dataset

labeled artificial ultrasound images and 61 labeled real ultrasound scans, in which we can have the annotations of the liver, kidney, gallbladder, spleen, and vessels. Different organs are assigned segmentation masks of different colors. Notably, there are actually more categories of anatomical structures in the original dataset. But we only pick 5 of them because instance numbers of adrenals and bones contained are no more than twenty, not feasible to train a reliable predictor. Table 1 shows the name of the anatomical structures and the corresponding instance number. We mixed and separated the dataset into 3 subsets: 787 images for training, 100 images for testing, and 100 images for validation. Since there is a huge difference in the image quality between CT-generated ultrasound images and real ultrasound images(see Fig. 11), we believe the performance of the proposed model can still be improved if we can have access to high-quality ultrasound image datasets.

Table 1. Dataset

Name	Liver	Kidney	Gallbladder	Vessels	Spleen
Number	591	377	219	289	172
Color	violet	yellow	green	red	pink

4.2 Detectron2

Wu et al. (2019) from FaceBook AI Research(FAIR) released a powerful object detection tool called *detectron2* containing many network architectures and training tools. We build the backbone framework based on the implementation of FPN in *detectron2*. And we develop our SRNN structure inserted into the FPN framework as a new feature extractor.

4.3 Loss Functions

Our training procedure incorporates several loss functions, some of which are outlined below:

Objectness loss For the detection of the object's appearance, the binary cross-entropy loss is used in the RPN head. This loss is only for the classification of objects and backgrounds. In RPN, this loss will be computed on the objectness logit feature map and the ground truth objectness logits. If the pixel is from some target object, it will be marked as "1", otherwise "0". The formulation is:

$$L_{obj} = -\frac{1}{n}\sum_{i=1}^{n} y_i \log\left(p(y_i)\right) + (1-y_i)\log\left(1-p(y_i)\right), \quad (2)$$

where y is the label ("1" for foreground, "0" for background), and $p(y)$ is the predicted probability of instance existence for all the grid points in the feature map.

Anchor and bounding box loss Both RPN and ROI(Box) heads use a smooth l1 loss for the proposed anchors and bounding boxes. The anchors and bounding boxes are represented as a tensor of length 4: (x, y, w, h), namely the x, y coordinates and width/height of the anchor or bounding box. Then with the ground truth information, 4 deltas (d_x, d_y, d_w, d_h) are calculated by

$$\begin{aligned} d_x &= (g_x - p_x) \\ d_y &= (g_y - p_y) \\ d_w &= \log\left(g_w/p_w\right) \\ d_h &= \log\left(g_h/p_g\right) \end{aligned} \quad (3)$$

where g represents the ground truth and p stands for the predicted anchor or bounding box. The deltas will be stacked together to compute the smooth l1 loss, given by

$$L_1^{smooth}(x) = \begin{cases} 0.5x^2 & if \quad |x| < \beta \\ |x| - 0.5 * \beta & otherwise \end{cases} \quad (4)$$

where β is a pre-defined smooth parameter.

Classification loss Softmax cross-entropy loss is calculated for all the foreground and background prediction scores:

$$L_{CE} = -\sum_{i=1}^{n} y_i log(p_i) \quad (5)$$

where y_i is the true label and p_i is the softmax probability for the i^{th} class.

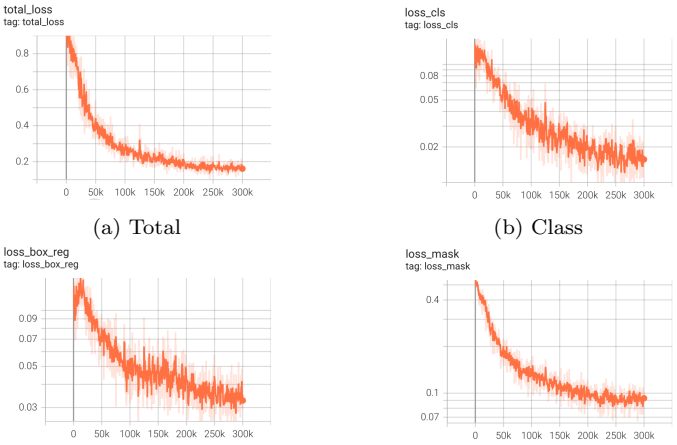
Mask loss The mask loss is defined as the average binary cross-entropy loss. Eq. 6 computes the mask loss for the k^{th} class:

$$L_{mask} = -\frac{1}{m^2}\sum_{1\le i,j\le m} [y_{i,j}\log\hat{y}^k_{i,j} + (1-y_{i,j})\log(1-\hat{y}^k_{i,j})] \quad (6)$$

where $y_{i,j}$ is the label of a cell(i,j) in the true mask for the region of size $m \times m$, and $\hat{y}^k_{i,j}$ represents the value of the same cell in the predicted mask.

4.4 Experiment Setup

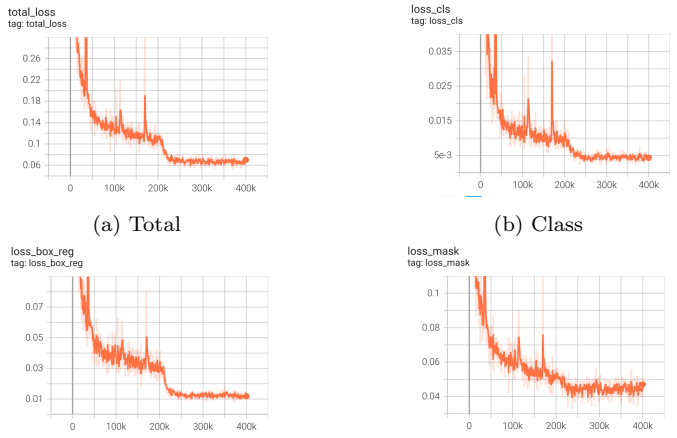
Our experiment builds upon *detectron2* framework in the PyTorch environment. We modified the original FPN in *detectron2* by adding the SRNN module. We train the model on Ubuntu 22.04 PC with an Intel Core i7-13700KF CPU and a NVIDIA GeForce RTX 3090Ti GPU(Driver Version: 535.129.03, CUDA Version: 12.2). The standardized region proposal network(RPN), Fast R-CNN, and Mask R-CNN heads are attached after the feature extractor as the proposal generators. Specifically, the output feature maps are from {res2, res3, res4, res5} of the ResNet layers. The size of the anchor generators are set to $32 \times 32, 64 \times 64, 128 \times 128$, and 256×256. For each feature map, FPN gives 1000 proposals. The region of interests(ROI) box head follows the structure of Fast R-CNN with 2 fully convolutional layers and 7×7 pooler resolution. The Mask R-CNN head has 4 convolutional layers and a pooler resolution of 14×14. The ROI heads score threshold is set to 0.5 for both box and mask heads. We have made some modifications to the model, enabling it to run under the *detectron2* framework. For example, we use a small learning rate 0.0025 to ensure there will



(a) Total (b) Class

(c) Bbox (d) Mask

Fig. 12. Loss curves of FPN



(a) Total (b) Class

(c) Bbox (d) Mask

Fig. 13. Loss curves of Trident Network

not be NaN and infinity scores in the final result and reduce the ROI head batch size from 512 to 128, which is computationally efficient while the accuracy is nearly the same. The batch size is set to 1. The FPN model is trained for 300k epochs, taking around 15 hours to converge. For the trident network model, we adopt the output from the res4 stage in ResNet as the backbone feature map. The dilation rates are set to 1, 2, and 3 respectively in the three convolution branches. The Trident Network model is trained for 400k epochs, but it still converges at 300k epochs, taking around 22 hours. Fig. 12 shows the converge curve of the FPN model, and Fig. 13 shows the curve of the Trident Network model. Compared with the loss curves of FPN, there are more fluctuations in the loss curves of Trident Network model. That is another point worth noting. Although the network structure graph of Trident Network seems to be simpler than the design of FPN, the actual complexity of these two models are exactly the contrast. Since we replace all the normal bottleneck blocks in the res4 stage of ResNet backbone with specially designed Trident bottlenecks, the number of parameters in Trident Network model are four times greater than that in FPN model. In scenarios with large models and small datasets, fluctuations in the loss curves are quite common.

5. RESULTS

5.1 *Quantitative Result*

We evaluated the performance of our trained model using the dice coefficient, as follows:

$$D = \frac{2|X \cap Y|}{|X| + |Y|} \tag{7}$$

Dice coefficientSorensen (1948) is one of the most widely used evaluation methods in the research field of image segmentation. We use the dice coefficient as our evaluation metric for the convenience of comparison with other research. The dice coefficient is twice the number of elements common to two sets X and Y, divided by the sum of the number of elements in each set. In our work, X and Y are the predicted classification map and the ground truth. Therefore, the numerator is regarded as the intersection pixels of the predicted mask and the ground truth, and the denominator is the sum of mask pixels in both. Considering we have 5 object classes (background not included), the coefficient score is computed separately and then averaged as the final score. As there might be no appearance of certain classes, we added a smoothing parameter ϵ to avoid zeros in the denominator. This smoothing parameter can be arbitrarily small, and we set it to 1×10^{-6}. The influence on the evaluation result from the smoothing parameter can be ignored as long as it is smaller than the given setting. The modified equation is given in (8), where n is the number of classes:

$$D = \frac{\sum_{i=1}^{n} \frac{2|X_i \cap Y_i|}{|X_i| + |Y_i| + \epsilon}}{n} \tag{8}$$

There are few similar types of research surrounding abdominal multi-organ segmentation. Neither any relevant benchmark nor competition exists. Therefore, we separately pick some comparable results from different research aiming at single-organ segmentation. Respectively, the segmentation result of the liver is compared with the work of Man et al. (2022). Marsousi et al. (2015) proposed a segmentation network targeting at kidney. Their result is taken into comparison as well. Moreover, the segmentation performance of gallbladder and spleen are compared with the work of Lian et al. (2017) and Yuan et al. (2022). The segmentation performance of vessels is not compared with other works, because most of the relevant research is focused on the segmentation of cardiac arteries. The numerical result may not seem encouraging compared with those well-aimed research. On one hand, the segmentation performance is limited by our lack of high-quality data. In our research, we have only 172 instances of training spleen samples, not to mention that most of the ultrasound images are pseudo-ultrasound images interpreted from CT images. On the other hand, the specifically targeted researchers usually introduced some prior knowledge into their segmentation algorithm like the detection of boundaries. Table 2 shows the dice score of each class.

5.2 *Different Implementations*

Improvement by SRNN We trained a pure FPN model and a pure Trident Network model for comparison to demonstrate the improvement brought by SRNN. The

Organ/Tissue	Related Work	FPN	FPN+SRNN	TridentNet	TridentNet+SRNN
Liver	0.821 by Man et al.	0.907	0.920	0.927	0.935
Kidney	0.5 by Marsousi et al.	0.806	0.878	0.904	0.896
Gallbladder	0.893 by Lian et al.	0.799	0.917	0.908	0.918
Vessels	–	0.801	0.912	0.935	0.934
Spleen	0.93 by Yuan et al.	0.810	0.898	0.899	0.937
Average	–	0.840	0.906	0.917	0.924

Table 2. Evaluation Result

results are also included in Table 2. We can see that regarding FPN stucture,the improvement of performance by SRNN is significant. The proposed model outperformed the pure FPN model with 6.6% similarity score gains. Meanwhile, the performance of pure Trident Network is already impressing. However, the introducing of SRNN model is less effective compared with FPN scenario. That is probably because we extract all spatial features from the res4 stage. The feature maps from res4 stage are of coarse resolution so that the spatial context is not precise, which is contradictory to our motivation of using feature maps from res4 mentioned in Section 2.3. Also, such trident structure has already embedded spatial context information in the perceptions. Therefore the SRNN did not bring in considerable improvements. However, considering the Trident Network model contains parameters 4 times greater than the FPN model, the trade-off between accuracy and efficiency shall be made regarding different scenarios. Moreover derived from the nature of RNNs, as the dataset scale goes up, the performance could be further improved. We will keep exploring the potential of the FPN&SRNN model in the universal scenarios out of medical image analysis applications.

RPN or One-stage Mask Generator We tested our feature extractor backbone with SOLOv2(Wang et al. (2020)), which is an one-stage mask generator. Table 3 shows the results of using one-stage mask generator(without RPN). We found the difference is negligible between using RPN and one-stage mask generator. Although this particular work focused on segmentation rather than detection, the RPN and Fast R-CNN are still helpful for other medical tasks like the size measurement of anatomical structures.

Generator Type	FPN+SRNN	TridentNet+SRNN
solov2	0.910	0.922
RPN+Mask R-CNN	0.906	0.924

Table 3. Comparison of RPN and one-stage mask generator

Depth of ResNet We tested replacing the feature extractor with a ResNet 50 backbone. Typically, a deeper neural network structure is expected to outperform a simpler one. However, due to the limited scale of our dataset, we consider it necessary to balance the complexity of the model with the available data. A more complex model like ResNet 101 might overfit given our dataset's size, whereas ResNet 50, being less complex, can potentially provide a better fit. This approach ensures that our model remains robust and generalizable, avoiding the pitfalls of overfitting that often accompany the use of deeper networks with limited data. Table 4 shows the performance of ResNet 50 backbones.

Organ/Tissue	FPN+SRNN	TridentNet+SRNN
Liver	0.936	0.915
Kidney	0.826	0.896
Gallbladder	0.892	0.926
Vessels	0.879	0.922
Spleen	0.882	0.950
Average	0.890	0.907

Table 4. Results Using ResNet 50 Backbone

ROI Threshold	FPN+SRNN	TridentNet+SRNN
0.50	0.906	0.924
0.60	0.912	0.925
0.70	0.917	0.925
0.80	0.916	0.931
0.90	0.919	0.931
0.95	0.919	0.931

Table 5. Results Using Different ROI Thresholds

After replacing the feature extractor backbone with ResNet 50, we observed a reduction of approximately 1.5 percent across both network structures. This decline indicates that while ResNet 50 offers a simpler, potentially more suitable architecture for our limited dataset, it does not capture the complexity of features as effectively as the more intricate original network. The decrease in the Dice score reflects this trade-off between model complexity and the dataset's ability to support it. Despite the slight decrease in accuracy, this adjustment offers valuable insights into the balance between model architecture and dataset size, guiding future optimizations and modifications in our approach to medical image segmentation.

ROI Heads Score Threshold The ROI threshold essentially dictates the sensitivity of the model to the regions of interest in an image, impacting the precision of the segmentation process. The experiment results presented in the Table 5 the influence of varying the Region of Interest (ROI) threshold.

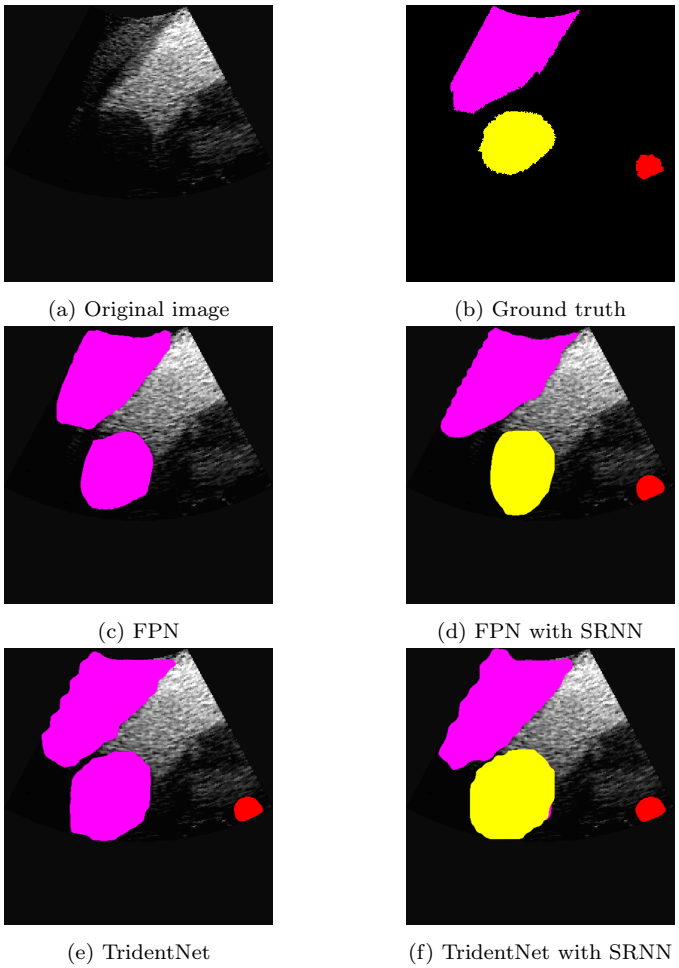
As the ROI threshold increases from 0.50 to 0.95, we observe an incresing in the performance of both models. For the FPN+SRNN architecture, the performance initially improves as the threshold increases, peaking at an ROI threshold of 0.90 with a score of 0.919. This suggests that a higher threshold, up to a certain point, aids in more accurate segmentation by possibly reducing false positives and focusing on more confidently predicted regions.

In contrast, the TridentNet+SRNN architecture shows a more consistent increase in performance as the threshold rises, achieving its best score of 0.931 at an ROI threshold of 0.80 and maintaining this score for higher thresholds. This indicates a robustness in the TridentNet+SRNN architecture, where it benefits from a stricter threshold,

effectively filtering out less confident predictions and enhancing overall segmentation accuracy.

5.3 Qualitative Result

We tested our proposed models using both artificial and real ultrasound images from our evaluation dataset. Fig. 14 shows an example of the semantic segmentation on ultrasound image. Fig. 14a is the original ultrasound image, and Fig. 14b is the corresponding ground truth (pink for spleen, yellow for kidney and red for vessels). Fig. 14c and Fig. 14d are the segmentation results generated by the pure FPN and FPN with SRNN. Fig. 14e and Fig. 14f are the segmentation results generated by the pure Trident Network w/o SRNN. Fig. 15 shows another example of comparison(violet or liver, yellow for kidney, green for gallbladder, and red for vessels). From these two examples. we can see the model knows better spatial context information after we brought in the SRNN module.



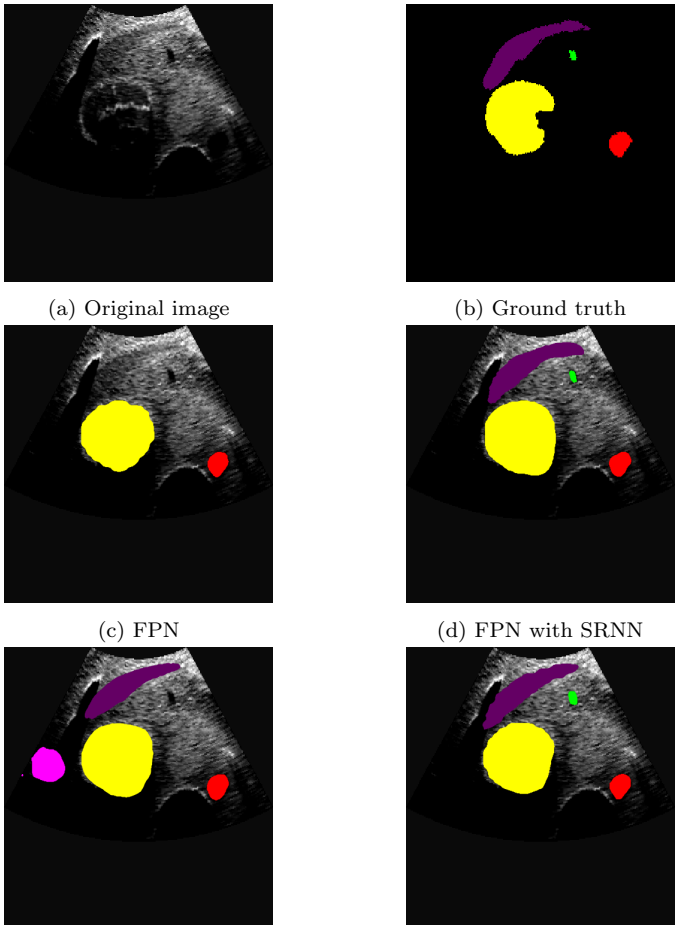
(a) Original image (b) Ground truth

(c) FPN (d) FPN with SRNN

(e) TridentNet (f) TridentNet with SRNN

Fig. 14. Test results

Furthermore, our proposed models were tested on the ultrasound images collected manually from an abdominal phantom in our laboratory. Fig. 16, Fig. 17 and Fig. 18 are showing an outstanding performance of our model: Fig. 16 is an ultrasound image collected from the phantom, and Fig. 17 and Fig. 18, respectively are the semantic masks generated by FPN+SRNN model and TridentNet+SRNN model.



(a) Original image (b) Ground truth

(c) FPN (d) FPN with SRNN

(e) TridentNet (f) TridentNet with SRNN

Fig. 15. Test results

6. CONCLUSIONS AND FUTURE WORK

We proposed multi-scale multi-organ/tissue segmentation methods combined with the utilization of SRNN. From the experimental results, we can see that the introduction of spatial context information has improved the performance of the models both in quantitative and qualitative comparison. The findings of this work would benefit from further research including different scan patterns, since a prior knowledge of the ultrasound scan pattern would help add more precise spatial context information.

ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI Grant Number JP23K03756 and the Asian Office of Aerospace Research and Development under Grant/Cooperative Agreement Award No. FA2386-22-1-4042.

REFERENCES

Almajalid, R., Shan, J., Du, Y., and Zhang, M. (2018). Development of a deep-learning-based method for breast ultrasound image segmentation. In *IEEE International Conference on Machine Learning and Applications*, 1103–1108.

Bell, S., Zitnick, C.L., Bala, K., and Girshick, R.B. (2016). Inside-outside net: Detecting objects in context with

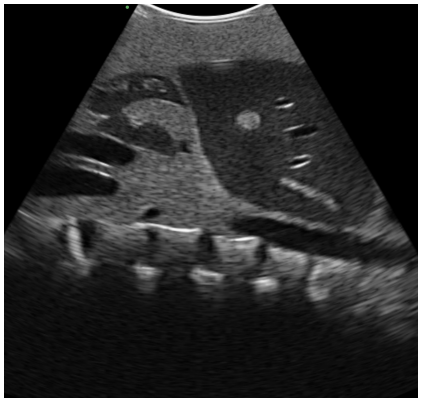


Fig. 16. In vitro image

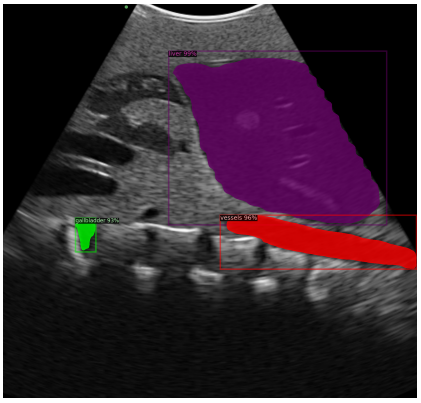


Fig. 17. FPN with SRNN

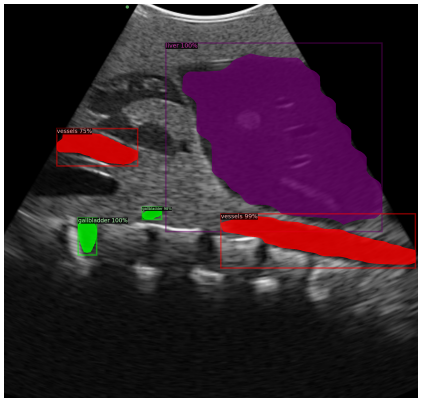


Fig. 18. TridentNet with SRNN

skip pooling and recurrent neural networks. *IEEE Conference on Computer Vision and Pattern Recognition*, 2874–2883.

Boukerroui, D., Baskurt, A., Noble, J., and Basset, O. (2003). Segmentation of ultrasound images—multiresolution 2d and 3d algorithm based on global and local statistics. *Pattern Recognition Letters*, 24(4), 779–790.

Byeon, W., Breuel, T.M., Raue, F., and Liwicki, M. (2015). Scene labeling with lstm recurrent neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 3547–3555.

Chen, G., Yin, J., Dai, Y., Zhang, J., Yin, X., and Cui, L. (2022). A novel convolutional neural network for kidney ultrasound images segmentation. *Computer Methods and Programs in Biomedicine*, 218, 106712.

Cho, K., van Merrienboer, B., Bahdanau, D., and Bengio, Y. (2014). On the properties of neural machine translation: Encoder–decoder approaches. In *SSST@EMNLP*.

Graves, A. and Schmidhuber, J. (2008). Offline handwriting recognition with multidimensional recurrent neural networks. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou (eds.), *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9, 1735–80.

Huang, H., Chen, H., Xu, H., Chen, Y., Yu, Q., Cai, Y., and Zhang, Q. (2021). Cross-tissue/organ transfer learning for the segmentation of ultrasound images using deep residual u-net. *Journal of Medical and Biological Engineering*, 41. doi:10.1007/s40846-020-00585-w.

Huang, W., Xue, Y., and Wu, Y. (2019). A cad system for pulmonary nodule prediction based on deep three-dimensional convolutional neural networks and ensemble learning. *PLOS ONE*, 14(7), 1–17.

Le, Q.V., Jaitly, N., and Hinton, G.E. (2015). A simple way to initialize recurrent networks of rectified linear units. *ArXiv*, abs/1504.00941.

Lei, Y., Wang, T., Roper, J., Jani, A., Patel, S., Curran, W., Patel, P., Liu, T., and Yang, X. (2021). Male pelvic multi-organ segmentation on transrectal ultrasound using anchor-free mask cnn. *Medical Physics*, 48. doi: 10.1002/mp.14895.

Li, Y., Chen, Y., Wang, N., and Zhang, Z. (2019). Scale-aware trident networks for object detection. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 6053–6062. URL `https://api.semanticscholar.org/CorpusID:57573786`.

Lian, J., Ma, Y., ma, Y., Shi, B., Liu, J., Yang, Z., and Guo, Y. (2017). Automatic gallbladder and gallstone regions segmentation in ultrasound image. *International Journal of Computer Assisted Radiology and Surgery*, 12. doi:10.1007/s11548-016-1515-z.

Lin, T.Y., Dollár, P., Girshick, R.B., He, K., Hariharan, B., and Belongie, S.J. (2017). Feature pyramid networks for object detection. *IEEE Conference on Computer Vision and Pattern Recognition*, 936–944.

Man, L., Wu, H., Man, J., Shi, X., Wang, H., and Liang, Q. (2022). Machine learning for liver and tumor segmentation in ultrasound based on labeled ct and mri images. In *2022 IEEE International Ultrasonics Symposium (IUS)*, 1–4. doi:10.1109/IUS54386.2022.9957634.

Marsousi, M., Plataniotis, K.N., and Stergiopoulos, S. (2015). Atlas-based segmentation of abdominal organs in 3d ultrasound, and its application in automated kidney segmentation. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2001–2005. doi: 10.1109/EMBC.2015.7318778.

Mignotte, M. and Meunier, J. (2001). A multiscale optimization approach for the dynamic contour-based boundary detection issue. *Computerized Medical Imaging and Graphics*, 25(3), 265–275.

Mignotte, M., Meunier, J., and Tardif, J.C. (2001). Endocardial boundary e timation and tracking in echocardiographic images using deformable template and markov random fields. *Pattern Anal. Appl.*, 4, 256–271.

Mishra, A., Dutta, P., and Ghosh, M. (2003). A ga based approach for boundary detection of left ventricle with echocardiographic image sequences. *Image and Vision Computing*, 21(11), 967–976.

Ronneberger, O., Fischer, P., and Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *ArXiv*, abs/1505.04597.

Schuster, M. and Paliwal, K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), 2673–2681.

Song, Y., Elibol, A., and Chong, N.Y. (2023). Two-path augmented directional context aware ultrasound image segmentation. In *2023 IEEE International Conference*

on Mechatronics and Automation (ICMA), 1815–1822. doi:10.1109/ICMA57826.2023.10215672.

Sorensen, T.A. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *Biol. Skar.*, 5, 1–34.

Vitale, S., Orlando, J., Iarussi, E., and Larrabide, I. (2019). Improving realism in patient-specific abdominal ultrasound simulation using cyclegans. *International Journal of Computer Assisted Radiology and Surgery*. doi:10.1007/s11548-019-02046-5.

Wang, X., Zhang, R., Kong, T., Li, L., and Shen, C. (2020). Solov2: Dynamic and fast instance segmentation. *Advances in Neural information processing systems*, 33, 17721–17732.

Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., and Girshick, R. (2019). Detectron2. `https://github.com/facebookresearch/detectron2`.

Yu, F. and Koltun, V. (2015). Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*.

Yuan, Z., Puyol-Antón, E., Jogeesvaran, H., Smith, N., Inusa, B., and King, A.P. (2022). Deep learning-based quality-controlled spleen assessment from ultrasound images. *Biomedical Signal Processing and Control*, 76, 103724. doi: https://doi.org/10.1016/j.bspc.2022.103724.

Zhang, Y., Ying, M.T.C., Yang, L., Ahuja, A.T., and Chen, D.Z. (2016). Coarse-to-fine stacked fully convolutional nets for lymph node segmentation in ultrasound images. In *IEEE International Conference on Bioinformatics and Biomedicine*, 443–448.