TransFusionNet: Semantic and Spatial Features Fusion Framework for Liver Tumor and Vessel Segmentation Under JetsonTX2

Xun Wang, Xudong Zhang, Gan Wang, Ying Zhang, Xin Shi, Huanhuan Dai, Min Liu, Zixuan Wang, and Xiangyu Meng

***Abstract*—Liver cancer is one of the most common malignant diseases worldwide. Segmentation and reconstruction of liver tumors and vessels in CT images can provide convenience for physicians in preoperative planning and surgical intervention. In this paper, we introduced a TransFusionNet framework, which consists of a semantic feature extraction module, a local spatial feature extraction module, an edge feature extraction module, and a multi-scale feature fusion module to achieve fine-grained segmentation of liver tumors and vessels. In addition, we applied the transfer learning approach to pre-train using public datasets and then fine-tune the model to further improve the fitting effect. Furthermore, we proposed an intelligent quantization scheme to compress the model weights and achieved high performance inference on JetsonTX2. The TransFusionNet framework achieved mean IoU of 0.854 in vessel segmentation task, and achieved mean IoU of 0.927 in liver tumor segmentation task. When profiling the Computational Performance of the quantized inference, our quantized model achieved 4TFLOPs on Node with NVIDIA RTX3090 and 132GFLOPs on JetsonTX2. This unprecedented segmentation effect solves the accuracy and performance bottleneck of automated segmentation to a certain extent.**

Manuscript received 28 February 2022; revised 1 September 2022; accepted 11 September 2022. Date of publication 16 September 2022; date of current version 7 March 2023. This work was supported in part by the National Key R&D Program of China under Grants 2021YFA1000100 and 2021YFA1000103, in part by the National Natural Science Foundation of China under Grants 61972416, 61873280, and 61873281, and in part by the Natural Science Foundation of Shandong Province under Grant ZR2019MF012. *(Corresponding authors: Zixuan Wang; Xiangyu Meng.)*

Xun Wang is with the Department of Computer Science and technology, China University of Petroleum, Qingdao, Shandong 266580, China, and also with the High Performance Computer Research Center, University of Chinese Academy of Sciences, Beijing 100190, China (e-mail: wangsyun@upc.edu.cn).

Xudong Zhang, Gan Wang, Ying Zhang, Xin Shi, Huanhuan Dai, and Xiangyu Meng are with the Department of Computer Science and technology, China University of Petroleum, Qingdao, Shandong 266580, China (e-mail: bigdongsir@163.com; s20070048@s.upc.edu.cn; zhangy9808@163.com; shix1104@163.com; daihuanhuan0901@163.com; x.meng@s.upc.edu.cn).

Min Liu is with the School of Mathematics and Physics, Qingdao University of Science and Technology, Qingdao, Shandong 266061, China (e-mail: liumin_qust@163.com).

Zixuan Wang is with the Minimally Invasive Interventional Therapy Center, Qingdao Municipal Hospital, Qingdao, Shandong 266011, China (e-mail: prince_room@sina.com).

Digital Object Identifier 10.1109/JBHI.2022.3207233

***Index Terms*—Liver tumor, liver vessel, medical image segmentation, transformer, 3D reconstruction, embedded microprocessor, computer-aided diagnosis.**

I. Introduction

LIVER cancer is the sixth most common primary cancer worldwide and the fourth leading cause of cancer death [1]. Therefore, there is an urgent need for effective prevention programs and treatments to reduce the harm caused by liver cancer. In the early stage of liver cancer, potential risks of coming serious liver cancer can be eliminated by surgical removal of the tumor or local treatment. In recent years, computer-assisted liver surgery (e.g., ablation and embolization) has been increasingly used for the treatment of primary and secondary liver tumor patients which are not eligible for common surgeries [2]. Computed Tomography (CT), as part of computer-assisted liver surgery, is a commonly implemented for clinical diagnostic approach to improve the visualization on liver, vessels and tumors [3]. Prior to the computer-assisted liver surgery, it is necessary for physicians to have information about the liver tumor contour and about its vessel system. Segmentation and 3-dimensional (3D) reconstruction according to CT images of patients is one of the most effective methods which help physicians to make preoperative planning and intraoperative navigation. However, there are some challenging obstacles in computer-assisted liver interventions. The most critical one is that segmentation of liver vessels and tumors from CT images is manual, which is time-consuming, and labor-intensive. It may lead to the inability to precisely pinpoint the vessels that supplies nutrition for the hepatic tumor, thus affecting hepatic embolization procedure, ablation and so on. As a result, there is an urgent need for an intelligent auxiliary diagnostic key embedded component which can be flexibly deployed in any CT instrument. Meanwhile inferential results of liver tumor and artery can be quickly generated with guaranteed precision, which assist physicians to complete rapid diagnosis and carry out next liver surgery plan.

In previous studies, many methods have emerged for segmenting liver vessels or tumors, but none of which considers segmenting vessels and tumors at the same time. This is due to the complicated background, heterogeneous shape and surrounding vessels irregularity of the tumor making it difficult to segment the hepatic vessels that supply nutrition for the

tumor [4]. Manual liver vessel and tumor segmentation is time consuming, tedious and sometimes impossible when there are plenty of patients. Traditional methods attempt to segment livers or tumors by active contour methods, tracking methods, and feature learning methods. Active Contour Model (ACM) is to detect object boundaries based on curve evolution theory and level set approach. Cheng et al. [5] implemented ACM with precise shape dimension constraints based on CT scan models for contour point detection of vessel cross-sections to plot vessel boundaries. Chung et al. [6] proposed an active contour method to segment portal vein and hepatic vein based on the regional intensity distribution of the image and the probability map of vessel occurrence. However, the active contour model tends to fall into the local optimum problem when extracting complex regions in the vector field, and cannot handle gray scale inhomogeneous images well. The tracing method starts by manual initialization or image preprocessing to initiate a single or specified number of seed points in the vessel, and then finds subsequent points based on the image derived data as a way to trace the vessel [3]. Tracking methods mainly include model-based algorithms [7], [8], [9], least cost path-based algorithms [10], [11]. However, if the initial seed points of these methods are not correctly positioned, the final segmentation results can be seriously affected.

In order to segment vessels or tumors from CT images, feature learning methods need to perform feature extraction from images and labels based on real segmentation to train machine learning models such as random forest (RF) [12], [13] and support vector machine (SVM) [14], [15] for automatically segmenting vessels or tumors from CT images. However, the robustness and generalization ability of machine learning models are limited. In recent years, many deep learning models, like convolutional neural networks [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], have gradually shown promosing performances in the field of medical image segmentation. Currently, segmentation models based on fully convolutional networks [26] and UNet [27], [28] architectures are the most effective ones. Huang et al. [29] combined 3D-UNet with data enhancement techniques, a variant of dice coefficient, to reduce the effect of high imbalance in some extent between hepatic vessels and background classes. Zhou et al. [30] proposed UNet++, a model that combines a deep supervised encoder and decoder and links the sub-networks of both through a series of hops as a way to reduce the semantic gap between the encoder and decoder feature mappings. Recently, Transformer [31], [32], [33], [34], [35], [36] has made great achievements in the field of deep learning, and TransUNet proposed by Chen et al. [37] applies transformer as an encoder to extract global contextual features and combines it with convolutional neural network for decoding. For segmentation of liver vessels and tumors, a high degree of accuracy must be achieved to enable clinical applications. In view of above mentioned methods including other UNet-based methods [38], [39], [40], performances still can be improved in terms of accuracy and efficiency despite of some attempts in architecture.

Deep learning has shown excellent capabilities in solving complex learning problems. Due to the need for real-world Applications, networks are becoming larger, which poses a major challenge for deploying deep learning models on the client side [41]. In recent years, there are many advanced deep learning models deployed on FPGAs or ASICs. For example, Transformer is deployed in FPGA [42], [43] and ASIC [44]. Wei et al. [45] introduced a fast and efficient lightweight network called Turbo Unified Network (ThunderNet). This model implements fast and efficient inference on the Jetson platform. Huang et al. demonstrate EDSSA-an Encoder-Decoder semantic segmentation networks accelerator architecture [46] which can be implemented with flexible parameter configurations and hardware resources on the FPGA platforms that support Open Computing Language (OpenCL) development. Ma et al. propose a specific dataflow of hardware CNN acceleration to minimize the data communication while maximizing the resource utilization to achieve high performance [47]. Tsai et al. presented the design of FPGA-based accelerator for DNN, which takes the advantages of low latency and low usage, and keeps the 96% recognition rate [48]. In the process of model transplantation, the tradeoff between speed, volume and accuracy of model inference is the focus of various researchers. With the development of medical image segmentation methods, related studies show significant precision in different lesion segmentation of multimodal medical data. Lightweight model deployment and transplantation of high-precision medical segmentation models into embedded micro-devices will greatly promote the development of automated surgery and automated diagnosis.

To fully harness the effects of Liver tumor vessel segmentation and 3D reconstruction, some urgent problems need to be solved: (1) How to design an accurate and fast automatic segmentation and 3D reconstruction method for liver tumor and vessel? (2) How to design a segmentation framework capable of learning spatial semantic fusion features to improve the segmentation accuracy of tumor and vascular details? (3) How to design efficient model quantification methods to enable high performance model inference and 3D reconstruction of tumors and vessels? (4) How to optimize the computational and storage overhead of the model to build a lightweight model and deploy the segmentation model to JetsonTX2 devices?

The main contributions of this paper are:

1) We propose the TransFusionNet framework that combines spatial, semantic and edge features of CT images to achieve accurate fine-scale segmentation of liver tumors and intrahepatic arterial vasculature.
2) We propose an intelligent quantization scheme based on reinforcement learning to compress the weights of the model, so that the model achieves the best inference performance on both JetsonTX2 and node with NVIDIA RTX3090 GPU.
3) By carefully quantifying, our model achieved high performance liver vessel tumor inference and 3D reconstruction on Node with NVIDIA RTX3090 GPU and JetsonTX2 device.

II. Methods

In order to better complete the segmentation task of liver tumors and blood vessels, we design a novel segmentation

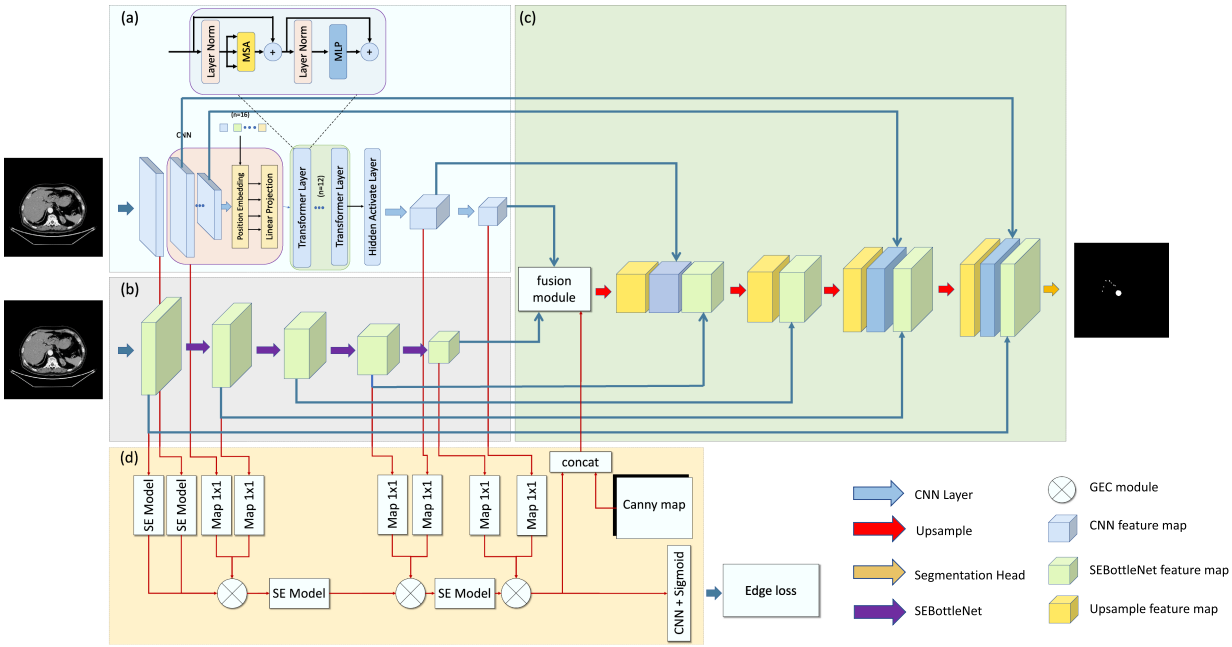

Fig. 1. Overview of the TransFusionNet model. (a) Transformer-based feature extractor. (b) Multi-layer local feature extraction module. (c) Fusion decoder for multiscale feature. (d) Edge Extraction Module.

architecture called TransFusionNet. We introduced a new feature extraction module fused with Transformer and CNN. Based on this module, the network can effectively extract image spatial features and semantically related features. At the same time, we proposed an Edge Extraction Module (EEM) which can significantly capture the edge feature of the image to cooperate with the training of the segmentation network. The model designed based on our ideas can effectively learn rich feature information, and can effectively ensure the segmentation accuracy of the edges of difficult-to-segment objects, which are critical in vascular and liver tumor segmentation tasks. The framework of our model is shown in Fig. 1.

A. Transformer-Based Semantic Feature Extraction Module

We introduce an encoder that can learn the global feature representation, which consists of a feature embedding module based on a feature extraction backbone and a feature extraction module that senses the semantically related information representation of the image based on the transformer [31]. This module adopts a brand-new feature extraction idea, by semantically representing the features of the picture and learning the global representation of semantic features.

The input image $i \in \mathbb{R}^{C\times H\times W}$ is first fed into the feature extraction backbone network. The network can extract the spatial information features of CT images and output the feature map $x \in \mathbb{R}^{C'\times H'\times W'}$. We divide the feature map x learned by the backbone into a series of patches $x_p^i \in \mathbb{R}^{C\times P^2}, i = 1, \ldots, N$, where the size of each patch is $P \times P$, and the number of patches denote by $N = \frac{H'\times W'}{P^2}$. For each patch, we use a convolution operation with a kernel size of $P \times P$ to obtain the information E_{info}^i of i-th patch to form an information matrix $\{E_{info}^1, E_{info}^2, \ldots, E_{info}^N\}$. In order to better learn location information using Transformer, Dosovitskiy et al. [49] perform a learnable location embedding for each patch to obtain the location matrix $\{E_{pos}^1, E_{pos}^1, \ldots, E_{pos}^N\}$ of the N patches. The feature of the i-th patch can be formulated by the following equation:

$$E^i = E_{info}^i + E_{pos}^i. \tag{1}$$

We adopt this position encoding method, so that the feature extraction module can effectively learn the position information of the features. We next feed the above obtained feature matrix $E = \{E^1, E^2, \ldots, E^N\}$ of x into multi Transformer layers to learn semantically representation of the feature map. In comparison to traditional convolution operation, transformer adopts a multi-head self-attention mechanism, and its core formulation is shown in (2):

$$y = \sum_{i=1}^{h}\sum_{j=1}^{w}\sum_{k=1}^{n}(softmax(Q_{ijk}^T \times K_{ijk}) \times V_{ijk}), \tag{2}$$

where h and w denote by width and height of the feature matrix E after feature extraction and location embedding. And n is the number of self-attention mechanism heads. $Q_{ijk}, K_{ijk}, V_{ijk}$ denote the query, key and value obtained by three linear transformations of the input E_{ij} in each self-attended head, respectively. $y \in \mathbb{R}^{C\times H\times W}$ denotes the output after one multi-headed self-attention. We stacked 12 transformer layers, and the output of the last layer can theoretically learn to incorporate a rich context feature representation of the CT image under a wider range of perceptual fields. We then feed the output of the Transformer layers into a three-layer convolution operation. The final output

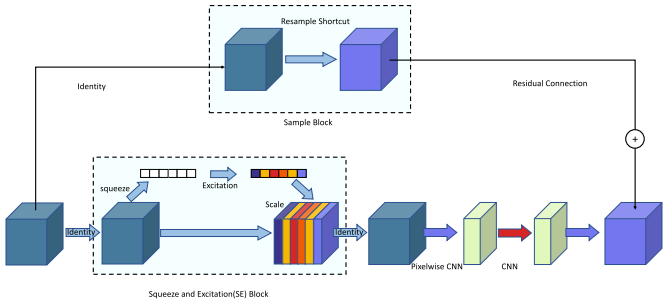

Fig. 2. BottleNet network structure with SEblock.

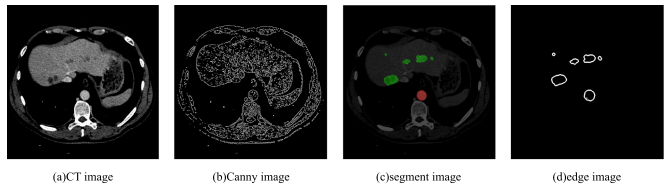

Fig. 3. An example of training dataset (a) input CT image. (b) Image canny map extracted from CT image by Canny algorithm. (c) Segmentation labels of tumors and blood vessels. (d) Edge label for tumors and blood vessels.

feature map consist of global high-level abstract information, effectively solving the problem of missing information caused by perceptual field defects in traditional deep CNN networks.

B. Local Spatial Feature Extraction Module

Transformer-based extraction module is a very powerful for semantically information feature, because the Transformer feature extraction module has advantages in learning semantically related features. In many ways, however, Transformer is not an effective replacement for traditional convolutional operations. For extraction of more subtle feature in some images such as features of interest regions and tiny vessel feature, CNN is nothing but the perfect solution. We designed a local residual network encoder based on multi-layer SEBottleNet stacking, as shown in Fig. 2. The encoder consists of a six feature extraction module. A max pooling operation is performed to extract the high-level feature representation after feed feature map to each feature extraction block. The input CT image $x \in \mathbb{R}^{H\times W}$ is first fed forward to a CNN module for high-level feature extraction, and the feature map $u \in R^{C\times H\times W}$ is obtained. Then, the feature map u is fed into a deep residual feature extractor stacked by five layers of SEBottleNet, each of which is used for learning the context features under the local perception field. BottleNet residual network [50] retains all the advantages of residual network and significantly reduces computation interval and computational burden. We introduced the Squeeze and Excitation (SE) [51] in the BottleNet to enhance the interdependence between feature map channels. The structure of the SEBottleNet is shown as in Fig. 2. The mean value $e_c \in \mathbb{R}^C$ of the feature embedding for each channel in the feature map $U \in \mathbb{R}^{C\times H\times W}$ can be obtained from the Squeeze section, as shown in the following equation:

$$e_c = \frac{1}{W \times H} \sum_{i=1}^{W} \sum_{j=1}^{H} u_c(i,j). \tag{3}$$

Where the $u_c(i,j) \in \mathbb{R}^C$ is the pixel in feature map U. The Excitation section can learn the feature weights e_c for each channel by s_c:

$$s_c = \delta(\mathcal{G}(e_c, \mathcal{W})). \tag{4}$$

Finally, the vector product $\tilde{O}$ of s and u is obtained by the Scale operation, and this is the final output of the SE module:

$$\tilde{O}_c = s_c \times u_c, \tag{5}$$

where $\tilde{O}_c$ is the feature map of a feature channel.

The SEBottleNet residual network splits the traditional convolutional operation into multiple modules to ensure that each module has a different feature extraction task. We introduced the Squeeze and Excitation module in the middle of the module to better learn the importance of the feature map channel dimensions, so that SEBottleNet has a stronger learning focus in the feature extraction process. Through the continuous stacking of SEBottleNet and maxpool, the encoder can continuously extract the local feature representation of the input CT image. Meanwhile, since each SEBottleNet is set with residual connections, it enables the encoder to effectively mitigate the degradation problem caused by network deepening.

C. Edge Extraction Module

Since the hepatic arterial vessels are very small, further refining the segmentation of the vessels and liver is a challenging task. In order to allow the model to learn more detailed spatial features, we introduce the EEM, which is specially designed to learn the edge features of blood vessels and tumor regions of interest and fuse the edge features to the segmentation network. The structure of this module is shown in Fig. 1(d). The EEM takes the feature maps of feature extraction layers and the CT edge map (Fig. 3(b)) extracted by the Canny algorithm [18] as the input, and predicts the edge result $e \in \mathbb{R}^{H\times W}$. This module predicts edge information and combines the predicted feature maps into the segmentation network. To accomplish this task, we process segmentation annotations to obtain edge annotations e_r (Fig. 3(d)), which can be used as a supervision condition for this module.

In this module, we used Gated Excitation Convolution (GEC) layer. GEC is the most important unit in EEM and it can filter out some irrelevant information to focus on extracting image edge features. GEC is applied between the EEM and the feature extraction module. It uses gating mechanisms to deactivate its own activations that are not deemed relevant by the higher-level information contained in the extraction module [52]. At the same time, we introduce an excitation module in the gating activation layer to learn the importance of different feature maps.

We define $t_i, c_i \in \mathbb{R}^{C\times \frac{H}{2^i}\times \frac{W}{2^i}}$ as the feature maps of the Transformer module and the local feature extraction module, and i denote the number of locations. Before using the GEC module, t_i and c_i were fed into a convolutional layer $C_{1\times 1}$ to obtain the image-dimensional feature maps $t_i', c_i' \in \mathbb{R}^{H\times W}$. Let

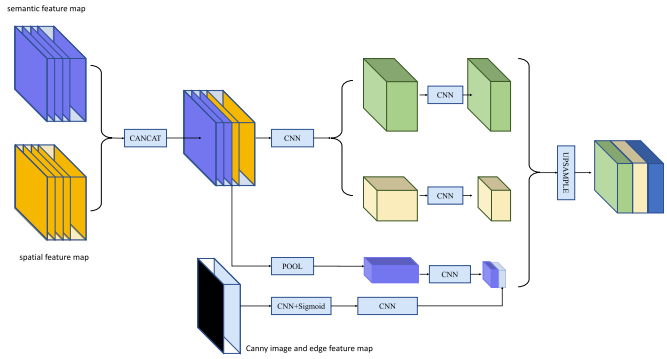

Fig. 4. Network framework of fusion module.

$e_i \in \mathbb{R}^{C\times H\times W}$ denote the feature map synthesized by EEM. Given the feature maps t_i', c_i' and e_i, an excitation convolutional layer is applied to generate the sigmoid activation $\alpha_i \in \mathbb{R}^{H\times W}$:

$$\alpha_i = \sigma(C_{1\times1}(F_{se}(cat(t_i', c_i', e_i)))), \tag{6}$$

where F_{se} denotes squeeze and excitation option as shown in (3)–(5). Finally, α_i and e_i is fed into Gated Convolution layer [52], [53], [54] and generate the e_i'. The Gated Convolution layer is compute as

$$e_i' = C_{1\times1}((e_i' \times \alpha_i) + e_i'). \tag{7}$$

Theoretically, GEC can be simply regard as a collection of attention for the spatial dimension and channel dimension of the feature map. Through GEC operation, the attention maps α_i selectively preserve the edge semantic features. We cancel the GEC operation on the shallow feature maps of the feature extractor, since the images feed to the convolution layer mainly learns the general low-level features, at the same time, the output feature map retains rich edge information. As the network deepens, the feature map will retain high-level features. Using GEC operation can effectively weight the useful edge information of high-level features in theory.

The Canny operator can effectively filter out the irrelevant features of the image to obtain the canny image as shown in Fig. 3(b). We think it is applicable for medical image segmentation. Therefore, we firstly concatenate the canny image and the last GEC module output e_n. Then we feed them together with the output feature maps of the two feature extractors to the Fusion Module. At the same time, the edge extraction module uses edge loss as the loss function and edge label as the supervision to optimize the prediction edge map.

D. Multi-Scale Feature Fusing Module

In this section, we introduce the multi-scale feature fusion decoding module to sample the semantic features learned by the three modules. This module takes the feature maps extracted by the three modules as the input and outputs the predicted category distribution map $\hat{y} \in \mathbb{R}^{K\times H\times W}$, where K represent the semantic classes.

We introduce a fusion module, which mainly fuses the feature maps of the three feature extraction modules. Fig. 4 shows the structure of this module. We design the module with reference to the spatial pyramid pooling (SPP). Firstly, the module uses $C_{1\times1}$ and $C_{3\times3}$ convolution to extract features from the concatenation result of the semantic feature map and spatial feature map respectively. Next, we feed it into the pooling layer and fused the edge feature map. Through the above operations, the feature maps of three different receptive fields are obtained. Finally, we sample and concatenate these three feature maps to output the fused feature map. Theoretically, the feature map output by these module can retain rich spatial features, semantic related features and edge features.

In the process of continuous feature extraction layer by layer in the coded network, the low-level information of the feature map is continuously filtered and the high-level information is extracted. UNet uses skip connections to conduct the feature maps of the encoding module of each stage to the decoding module of the corresponding stage, and the network can fully learn the feature maps of different levels of the image. We adopt the skip connection operation from UNet and introduce skip connections to different feature encoders to allow the whole network to better learn the feature information of different encoders at different levels. The skip connection introduced in the local feature extraction module is similar to the traditional UNet module, which combines the short-range skip connection (residual connection) and the long-range skip connection of SEBottleNet. As for the Transformer based feature extraction module, we first introduce skip connections in the encoding process of the backbone network to connect the intermediate feature maps in the forward propagation process of the backbone embedding network, which improves the low-level feature loss in the feature embedding process of the backbone network. Next, we add skip connections to the feature maps with global feature representations after Transformer feature encoding fusion to fuse the global low-level features. Eventually, after continuously fusing low-level feature maps of different scales, the decoder can learn the semantic information of images from coarse to fine.

E. Multi Task Training Strategy

We propose the EEM to cooperate with the segmentation task of the model, so we train the model to complete semantic segmentation and edge information segmentation at the same time. We introduce the joint optimization of edge loss and segmentation loss respectively. At the same time, in order to better ensure the consistency of multi task learning optimization, we set up a regularization methods to balance the two losses.

We use Dice and Cross Entropy (CE) as the loss function of segmentation task to predict semantic segmentation y:

$$\mathcal{L}_{seg}^{\theta,\psi} = \lambda_1 \mathcal{L}_{Dice}(y, \hat{y}) + \lambda_2 \mathcal{L}_{CE}(y, \hat{y}), \tag{8}$$

where $y \in \mathbb{R}^{H\times W}$ denotes the real semantic label map of liver tumor and vessel. In the (8), λ_1 and λ_2 represent hyper parameters. As for edge prediction, we use Binary Cross Entropy (BCE) loss. In this experiment, the model mainly focuses on tumor and vascular segmentation. We extract their common edges to obtain edge label $\hat{e} \in \mathbb{R}^{H\times W}$ (Fig. 3(d)) and take $\hat{e}$ as the loss

supervision. Therefore, the edge loss can be expressed as:

$$\mathcal{L}_{edge}^{\theta,\psi} = \lambda_3 \mathcal{L}_{BCE}(e, \hat{e}), \tag{9}$$

where, e represents the edge predict map of the edge extraction module. It is worth noting that in the optimization process, the parameters of feature extraction modules and edge extraction modules will be optimized based on loss. Next, we input the feature map output by the edge extraction module into the fusion module to predict the segmentation results. Therefore, the prior knowledge learned by the edge extraction module is retained in y. At the same time, the segmentation loss will pay more attention to the edge features in the optimization process.

We introduce regularization methods to make the model cooperate better in the training process. As mentioned above, $y \in \mathbb{R}^{K\times H\times W}$ represents the predicted segmentation map and $e \in \mathbb{R}^{H\times W}$ represents the predicted edge graph. Therefore, we introduce shape regularization, which can be expressed as:

$$\mathcal{L}_{sreg}^{\theta,\psi} = \lambda_4 ||Sigmoid(\oplus y) \times e - e||, \tag{10}$$

where $\oplus$ represents the pixel-wise addition of y without the background label map which can be implemented using kernel fixed convolution operator. This operation outputs a label map containing the predict region of the tumor and blood vessels. In the beginning of model training, since the edge extraction module was random initialized and cannot accurately predict the e, (10) does not play any role. We therefore introduce a dynamic adjustment strategy, which is set λ_4 to 0 before 100 epoch and $\lambda_4 \geq 0$ after 100 epoch.

Finally, the loss function of the model is:

$$\mathcal{L}_{total} = \mathcal{L}_{seg}^{\theta,\psi} + \mathcal{L}_{edge}^{\theta,\psi} + \mathcal{L}_{sreg}^{\theta,\psi}. \tag{11}$$

We set the epoch to 300, the initial learning rate to 0.001 (using the cosine annealing learning rate decay method), and the batch size to 8. The model is trained using an SGD optimizer with a momentum of 0.9 and a weight decay of 1e-4.

F. Applying Transfer Learning to TransFusionNet

The TransFusionNet can significantly learn full-resolution context feature information, and its segmentation effect in the public dataset of blood vessels and liver tumors is significant. However, due to the scarcity of the enhanced CT images of liver cancer after the screening and the difficulty of labeling tumors and blood vessels, we obtained CT images of 18 patients. Too little data will inevitably affect the performance of the model and deepen the over-fitting problem. For this purpose we introduce a transfer learning strategy, which does not require exactly representative training data and is able to take advantage of the similarity between datasets to capture specific prior knowledge during the training phase of the model in order to construct new segmentation models.

We first pre-trained the models using the public datasets LITS and 3Dircadb to obtain a liver tumor segmentation model and a liver vascular segmentation model, respectively. Then, we use our liver tumor data and liver vascular data to retrain the model obtained by pre-training. When we need to perform segmentation of liver tumor and blood vessels of CT images, we only need to input one CT image, and the model will segment the tumor and blood vessel parts of CT images respectively.

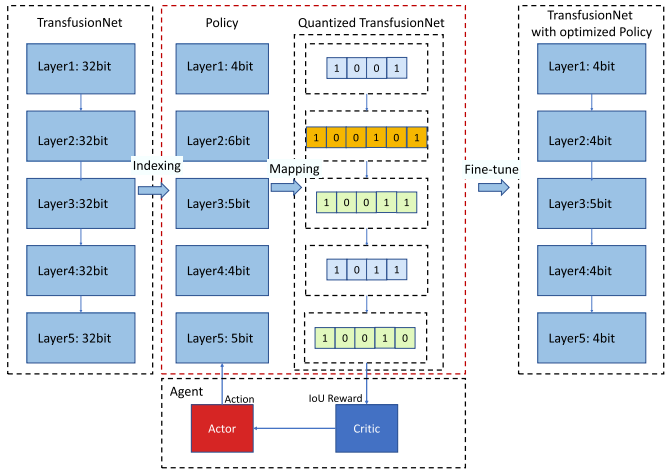

Fig. 5. Overview of TransFusionNet quantization using reinforcement learning. We set the IOU difference as the reward of the model. The Agent can automatically search for the optimal bitwidth strategy based on the reward. Finally, we quantize the model according to the optimal bitwidth and fine-tune the model to construct a fast and low storage overhead model.

G. Quantification and Fine-Tuning of Inference Models

TransFusionNet involves different kinds of feature extraction modules and feature fusion sampling modules with high computational and storage overhead for training and inference, which brings challenges for the deployment of the model on embedded devices. We proposed a model quantization scheme based on Hardware-Aware Automated Quantization (HAQ) [55] to compress the CNN and Dense layer of the framework and optimize the computational and storage overhead. Fig. 5 shows the quantization method.

We searched the TransFusionNet layer-by-layer and create the index of each basic quantitative layer. We define the state $\mathcal{S}_k$ of the CNN layer k was

$$\mathcal{S}_k = \{k, c_{in}, c_{out}, d_{kernel} s, d, n_{params}, i_{dw}, i_w, a_{k-1}\}, \tag{12}$$

where k was the layer global index, c_{in} was input channel size, c_{out} was output channel size, d_{kernel} was the kernel size, s was the kernel stride, d was the feature map size, n_{params} was the parameters, i_{dw} was the indicator for depthwise convolution, i_w was the indicator for weight, and a_{k-1} was action during the last step. Meanwhile, the state $\mathcal{S}_k$ of Dense layer k was denoted as

$$\mathcal{S}_k = \{k, f_{in}, f_{out}, d, n_{params}, i_w, a_{k-1}\}, \tag{13}$$

where k was the layer global index, f_{in} was the input feature size, f_{out} was the output feature size, d was the feature size, n_{params} was the parameters, i_w was the indicator for weight, and a_{k-1} was action during the last step.

We adopt continuous action space and round to discrete value which was

$$b_k = round(b_{\min} + a_k \times (b_{\max} - b_{\min})), \tag{14}$$

where $b_{\max}$ and $b_{\min}$ was the min and max bandwidth. Given the action a_k from the agent, the quantitative strategy is denoted as

$$\mathcal{Q}(w, a_k, c) = round(clip(w, c)/s) \times s, \tag{15}$$

where $clip(w, c)$ was to truncate the weight w into $[-c, c]$, c was the optimal value that minimizes the distance between origin weight w and quantized weight, and s was the scale factor which was denoted as $s = c/(2^{a_k-1} - 1)$. After each quantification we retrain the model for one more epoch and use IoU (Intersection over Union) in the (20) as the metrics to analyze the segmentation performance. We then calculate the reward function:

$$\mathcal{R} = \mu(IoU_{origin} - IoU_{quant}). \tag{16}$$

In our experience, we set μ to 0.1.

For the Agent setting, we refer to the deep deterministic policy gradient (DDPG) [56] method to construct an off-policy actor-critic algorithm based on the continuous control problem. We specify that one step represents the decision of agent to assign bit-widths for a specific layer, and one episode indicates the agent completes the assignment for all layers. Therefore, the Q-function in the exploration process is represented as

$$\hat{Q}_i = \mathcal{R}_i + \gamma \times Q(\mathcal{S}_{i+1}, \mu(\mathcal{S}_{i+1})|\theta^{\mathcal{S}}), \tag{17}$$

where γ is the discount factor, and we set to 1 in our experiment. Based on the Q-function, the loss function can be approximated by

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N} ||\hat{Q}_i - Q(\mathcal{S}_i, a_i|\theta^{\mathcal{S}})||_1, \tag{18}$$

where N denotes the number of step in an episode.

By optimizing the loss function, we obtain the optimal bit-width quantization strategy for each layer of TransFusionNet. Finally, based on the optimal strategy we obtain the quantization bit-width of each layer of the network, and after quantization and fine-tuning, we obtain the light-weight model of TransFusionNet.

H. 3D Reconstruction of Tumor and Vessels Using Light-Weighted TransFusionNet

Medical image segmentation has a wide range of applications in medical research and practice fields such as medical research, clinical diagnosis, pathological analysis, computer-assisted surgery, and three-dimensional simulations. In this section, we use light-weighted TransFusionNet to predict the liver vessel and tumor segmentations under JetsonTX2 and make 3D reconstruction of tumors and vessels based on the segmentations.

We define segmentation model $\mathcal{F} : \mathbb{R}^{N\times H\times W} \rightarrow \mathbb{R}^{N\times C\times H\times W}$, which has been fully trained. Next, the arterial phase CT-enhanced image $x \in \mathbb{R}^{N\times H\times W}$ feed into $\mathcal{F}$ to predict the liver and vessel label map $o \in \mathbb{R}^{N\times C\times H\times W}$ where $C = 3$ in this segmentation task. We need to construct 3D reconstruction result according to the segmentation label map o. Thus, to obtain segmentation result $y \in \mathbb{R}^{N\times H\times W}$ from label map o, we use the following equation:

$$y = \mathcal{G} * argmax(o). \tag{19}$$

Where $\mathcal{G}$ denotes Gaussian filter. Finally, we save the reconstruction results y in $.nrrd$ format and use 3Dslicer for visual display.

I. Evaluation Metrics

In order to better evaluate our model segmentation performance from multiple perspectives, we selected 5 evaluation metrics including: IoU, DSC, VOE, Recall, Precision. IoU is the calculation of the intersection of the real annotation and the segmentation result. The calculation method is

$$IoU = \frac{R_{pre} \cap R_{real}}{R_{pre} \cup R_{real}}, \tag{20}$$

where R_{pre} represents the segmentation result predicted by the model, and R_{real} represents the actual segmentation result. The DSC (Dice Similariy Coefficient) represents the ratio of the area where the segmented image and the real image intersect to the total area. The calculation method is

$$DSC = \frac{2 \times (R_{pre} \cap R_{real})}{R_{pre} + R_{real}}, \tag{21}$$

where R_{pre} represents the segmentation result predicted by the model, and R_{real} represents the actual segmentation result. VOE (Volumetric Overlap Error) represents the difference between the area of the segmented image and the real image, and usually represents the error rate of segmentation. The specific calculation method is

$$VOE = \frac{2 \times (R_{pre} - R_{real})}{R_{pre} + R_{real}}. \tag{22}$$

Precision is the proportion of pixels that are actually not in the region of interest correctly judged as not in the region of interest. It measures the ability to correctly judge the pixels that are not in the region of interest in the segmentation experiment. Its calculation method is

$$Precision = \frac{I - R_{pre} \cup R_{real}}{I - R_{real}}. \tag{23}$$

Where I is the original input image. Recall is the proportion of pixels that are correctly judged as pixels in the region of interest. It measures the ability to correctly segment the region of interest.Its calculation method is

$$Recall = \frac{R_{pre} \cap R_{real}}{R_{real}}. \tag{24}$$

For the computational performance analysis, we use floating point operations per second (FLOPs) as the evaluation metric. The number of floating point operations (FLOP) of the inference model can be collected using $fvcore$. Then for a patient with n CT images, the FLOPs are calculated as

$$FLOPs = \frac{n \times FLOP}{t}, \tag{25}$$

where n denote the number of CT slices, t denote the total inference latency.

III. Experimental and Discussion

A. Experimental Setup

1) Dataset: The LITS (Liver and Liver Tumor Segmentation, https://competitions.codalab.org/competitions/17094) dataset contains 130 cases of tumors, metastases, and cysts. These CT scans have large spatial resolution and field of view (FOV) differences [4]. 3Dircadb (3D Image Reconstruction for Comparison of Algorithm Database, https://www.ircad.fr/research/3d-ircadb-01/) is a public dataset that can be used to train and test liver vessel segmentation methods, including 20 patients in different image resolutions, vessel structure, intensity distribution and liver vessel comparison CT enhancement [29]. At the same time, we collected CT-enhanced images of 18 patients and constructed Liver Tumor Blood Vessel (LTBV) dataset. In order to clearly distinguish the blood vessels in the liver and reduce the burden of labeling, we only retained the arterial phase images of 18 patients. Due to the scarcity of trainable samples, we only divide into training dataset and test dataset. Our research is carried out following the principles of the Declaration of Helsinki, and we have got approval from the Ethics Committee of Qingdao Municipal Hospital. The LITS and 3Dircadb datasets cover a wide range of CT images with different resolution differences and field of view (FOV) differences. We use these two datasets for model pre-training, and use our private dataset for the fine-tuning training of the model for hepatic artery and tumor segmentation tasks. The above three datasets are divided into training set and test set according to the ratio of 8:2.

2) Machine Configurarion: The training and quantization of the model were run on servers with two NVIDIA Tesla V100 (32 GB) GPUs, and the inference of the model was run on servers with one NVIDIA 3090 GPU. To test the portability of the models, we used the JetsonTX2, a new embedded device introduced by NVIDIA. Fig. 6 shows the architecture of the jetsonTX2. This device is equipped with Nvidia Tegra X2 processor which consists of a GPU with a pascal architecture, NVIDIA Denver 2 ARM CPU with two cores and ARM Cortex-A57 with four cores. Denver 2 ARM CPU is suitable for administration tasks, ARM Cortex-A57 is for multi-threaded computationally intensive tasks, and the GPU contains 256 CUDA cores for highly computationally intensive tasks. The entire device integrates 8 GB LPDDR4 memory and 32 GB eMMC storage, which achieved 0.63TFLOPs with low power consumption.

B. Performance Comparison With State-of-The-Art Methods

We choose 5 segmentation models to compare with our method, the 5 models are SegNet [16], UNet [27], UNet++ [30], UNet3+ [57], TransUNet [37]. At the same time, we divide the proposed method into TransFusionNet(TFN) and TransFusionNet with edge module(TFNEdge), and evaluate them respectively. We first compare the segmentation effects of five models on blood vessels and tumors based on two public datasets:LITS (Tumor) and 3Dircadb (Vessel). Next, we use the LTBV dataset to fine-tune the five models and compare the segmentation effects of the five models.

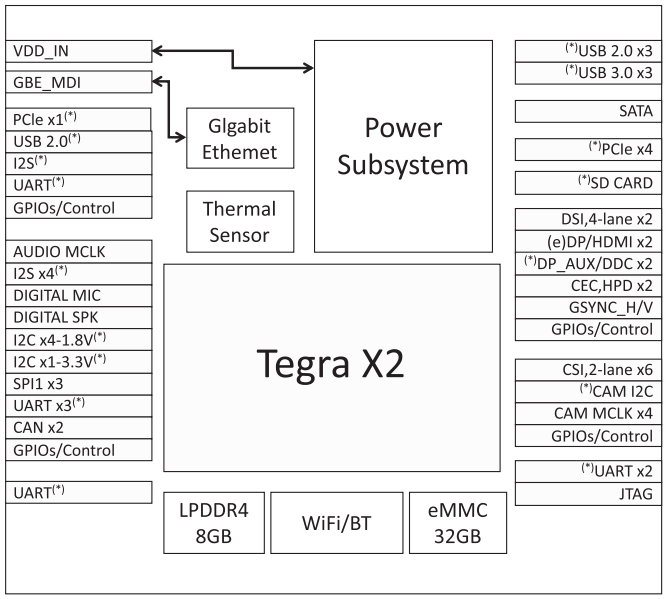

Fig. 6. The architecture of JetsonTX2.

TABLE I
Comparison Between Our Method and Various Methods on LITS and 3Dircadb Datasets

Dataset	Methods	IoU	DSC	VOE	Precision	Recall
3Dircadb	SegNet [16]	0.839	0.907	-0.067	0.938	0.879
	UNet [27]	0.846	0.913	-0.079	**0.951**	0.880
	UNet++ [30]	0.831	0.904	-0.062	0.934	0.879
	UNet3+ [57]	0.853	0.917	-0.059	0.945	0.894
	TransUNet [37]	0.847	0.913	-0.066	0.944	0.885
	TFN	0.854	0.918	-0.041	0.938	0.901
	TFNEdge	**0.863**	**0.921**	**-0.051**	0.947	**0.901**
LITS	SegNet [16]	0.805	0.887	-0.035	0.904	0.875
	UNet [27]	0.832	0.905	-0.024	0.917	0.897
	UNet++ [30]	0.828	0.902	-0.020	0.912	0.896
	UNet3+ [57]	0.821	0.895	-0.002	0.899	0.898
	TransUNet [37]	0.834	0.905	-0.040	**0.923**	0.889
	TFN	**0.840**	**0.910**	**-0.018**	0.919	**0.904**
	TFNEdge	**0.840**	**0.910**	**-0.018**	0.919	**0.904**

1) Performance Comparison of Liver Tumor and Blood Vessel Segmentation Based on Public datasets:LITS and 3Dircadb: The performance of TransFusionNet and the other four methods on two public datasets is shown in Table I. The experimental results show that the IoU of TransFusionNet on the 3Dircadb dataset can reach 0.854, and the DSC can reach 0.918, which is 0.8% and 1.1% higher than the IoU and DSC of the baseline method UNet. The IoU is 2.3% and 0.7% higher than UNet++ and TransUNet, respectively. However the IoU can reach 0.863 when using TransFusionNet with edge extraction module. On the LITS dataset, that is, when performing liver tumor segmentation, the IoU and DSC of TransFusionNet can reach 0.840 and 0.910. As can be seen from Table I, the VOE of TransFusionNet on the two datasets, that is, the error rate is also far lower for other models.

TABLE II
COMPARISON BETWEEN OUR METHOD AND VARIOUS METHODS ON LTBV DATASETS

Dataset	Methods	IoU	DSC	VOE	Precision	Recall
Vessel	SegNet [16]	0.803	0.881	-0.056	0.907	0.858
	UNet [27]	0.812	0.893	**-0.013**	0.902	0.890
	UNet++ [30]	0.809	0.892	-0.058	0.919	0.868
	UNet3+ [57]	0.821	0.897	0.022	0.892	**0.909**
	TransUNet [37]	0.818	0.897	-0.049	0.920	0.876
	TFN	0.822	0.899	-0.054	0.925	0.877
	TFNEdge	**0.854**	**0.901**	-0.040	**0.933**	0.895
Tumor	SegNet [16]	0.905	0.948	0.002	0.922	0.931
	UNet [27]	0.915	0.954	-0.018	0.963	0.946
	UNet++ [30]	0.912	0.952	0.003	0.952	0.954
	UNet3+ [57]	0.827	0.899	-0.037	0.918	0.885
	TransUNet [37]	0.920	0.955	-0.023	0.967	0.945
	TFN	**0.927**	**0.961**	**-0.011**	0.966	**0.955**
	TFNEdge	0.917	0.954	-0.022	**0.975**	0.945

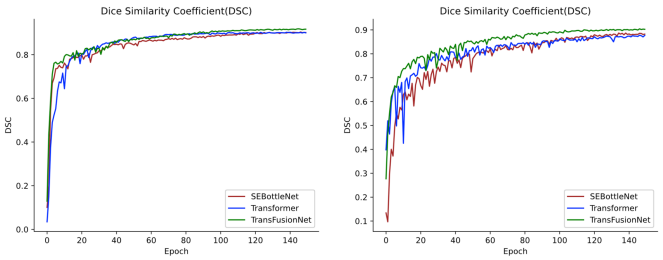

Fig. 7. Ablation experimental results for different feature extraction structures. (a) DSC on 3Dircadb dataset. (b)DSC on LITS dataset.

2) Performance Comparison of Liver Tumor and Blood Vessel Segmentation Based on Private dataset:LTBV: We used five models pretrained on LITS and 3Dircadb datasets to perform fine-tune training on the LTBV dataset. The performance of each model on LTBV dataset as shown in Table II. From Table II, we can see that the IoU of TransFusionNet on the vessel dataset reached 0.822, and the DSC can reach 0.899. This is 1.9% and 1.8% higher than the IoU and DSC of the baseline method SegNet, and is higher than the IoU and VOE of TransUNet. They are 0.4% and 0.5% higher respectively. On the tumor dataset, the IoU and DSC of TransFusionNet are as high as 0.927 and 0.961, which shows our method can still achieve the best results after LTBV transfer learning.

C. Ablation Study of TransFusionNet

1) Performance of Semantic and Local Spatial Feature Extraction Module: We conduct ablation experiments on the TransFusionNet semantic feature extraction module and local feature extraction module, with the aim of testing the effect of the above two feature extraction modules on the segmentation performance of TransFusionNet. From Fig. 7(a) we can see that the Transformer-based semantic feature extraction module performs better than the SEBottleNet module on the vascular dataset of 3Dircadb. We believe that the Transformer can learn global contextual feature representation of the CT image, especially it encodes the image location information, which certainly helps to enhance the segmentation of the image in global reception. On the LITS tumor dataset, as shown in Fig. 7(b), the segmentation accuracy of the SEBottleNet-based local spatial extraction module is higher, which is attributed to the fact that its internal CNN and local residuals are more interested in some finer features in the image, such as tumor edge features. From Fig. 7, we can see that our works achieves an effective improvement in the segmentation accuracy of liver vessels and tumors by combining the Transformer module and SEBottleNet module.

TABLE III
ABLATION STUDY RESULTS FOR SKIP CONNECTION USED ON DIFFERENT MODULE OF OUR METHOD

Dataset	Skip connections used	IoU	DSC	VOE	Precision	Recall
3Dircadb	$None$	0.654	0.780	-0.163	0.862	0.732
	CNN	0.767	0.858	0.059	1.066	0.886
	$Trans$	0.785	0.870	0.019	0.880	0.868
	CNN+Trans	**0.854**	**0.918**	**-0.041**	**0.938**	**0.901**
LITS	$None$	0.805	0.887	-0.035	0.904	0.875
	CNN	0.832	0.905	-0.024	0.917	0.897
	$Trans$	0.828	0.902	-0.020	0.912	0.896
	CNN+Trans	**0.840**	**0.910**	**-0.018**	**0.919**	**0.904**

2) Performance of Edge Extraction Model: From results in Tables I and II, we can find that the TFNEdge module shows the best effect in vascular segmentation task, but the outcome in tumor segmentation task is not as good as TFN. Anyway, the tumor segmentation effect of TFNEdge module also exceeds the state-of-the-art model. This shows that the edge extraction module can play a good role in the task of small target segmentation. However, when segmenting large targets, due to the function of the edge loss function, the network will pay more attention to edge optimization and affect the global control of the whole target.

3) Performance of Skip Connection: In the Encoder-Decoder structure, the encoder learns to extract the high-frequency image representation of the feature map, and the decoder continuously learns feature recovery based only on the high-frequency feature coding output from the encoder. The role of low-frequency feature information is ignored in the process of encoding and decoding, yet low-frequency features often have their non-negligible role. The role of skip connection is to allow the network to learn low-frequency features during the encoding and decoding process. In this experiment, we use two datasets, LITS and 3Driadb, to analyze the segmentation contribution of skip connections on each module of TFN. According to the characteristics of TFN, we design four comparison models. As shown in the second column of Table III, $None$ indicates that all skip connections of the semantic and spatial feature extraction modules are cancelled, CNN indicates that only skip connections of the spatial feature extraction module are retained, $Trans$ indicates that only skip connections of the semantic feature extraction module are retained, and $CNN + Trans$ indicates that all skip connections are retained, which is the TFN model. Table III shows the experiment results. According to the results in the Table III, we can find that the model retaining the global local skip connections has a significant improvement compared to the model with the jump links removed. This result proves the importance of skip connections for TransFusionNet

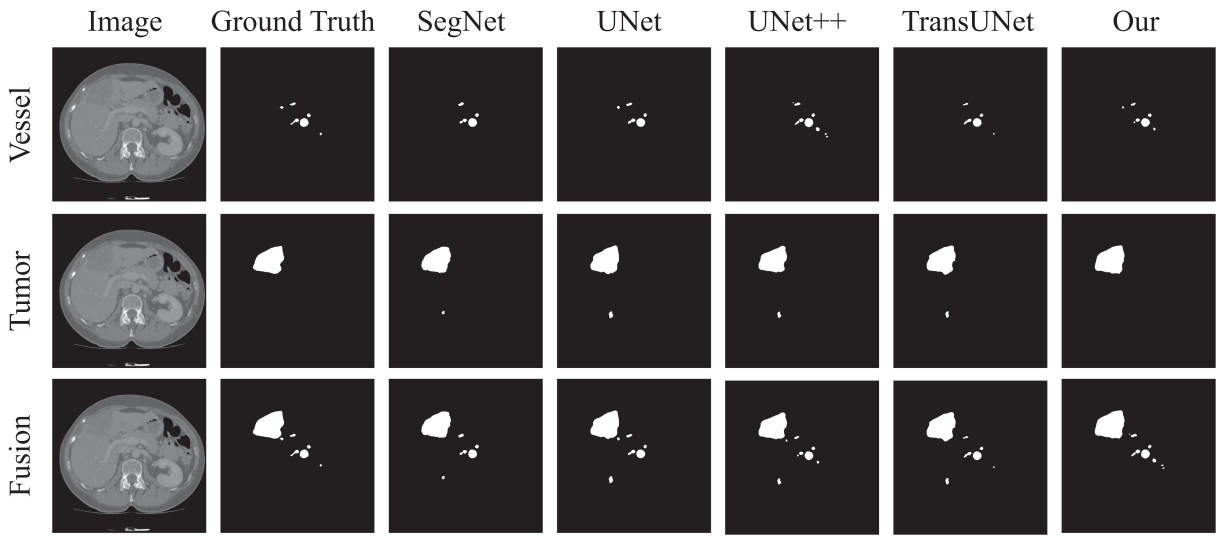

Fig. 8. Qualitative analysis of the 2D segmentation results of blood vessels, tumors and fusions of TransFusionNet and other comparison models from visual perspective.

and also shows that the low frequency features of the image have a significant impact on the segmentation results.

D. Visualizations

From the above quantitative experimental methods, our model has the best performance in the segmentation of liver blood vessels and liver tumors. Next, we use TransFusionNet and other comparable models on a test case of the LTBV dataset to segment liver tumors and blood vessels and then visualize them. The first row of Fig. 8 is the segmentation result of Vessel, the second row is the segmentation result of Tumor, and the third row is the result of fusing the first two rows of segmentation results in the same coordinate system.

From the perspective of visual analysis, SegNet and UNet are not accurate in segmenting the details of blood vessels. Although UNet++ can identify some details of blood vessels, the error rate is too high. TransFusionNet almost perfectly segmented the details of blood vessels, which is more accurate than TransUNet. This is attributed to SEBottleNet's extraction of local receptive field information and the importance of different channels. In tumor segmentation, UNet++ has a great segmentation result for the edge of the tumor, while SegNet and UNet perform poorly in this respect. All comparison models have some wrong tumor segmentation, and TransFusionNet not only avoids these wrong segmentation but also can segment the edge and contour of the tumor accurately. We believe that after TransFusionNet extracts global and local information, the multi-scale feature fusion decoder almost perfectly restores the feature of the image, so that the segmentation accuracy is significantly improved, and the error rate is low.

In summary, the above comparison models are not accurate in segmenting tumors and blood vessels. They are easy to misclassify some areas that are not tumors, and they are not sensitive to the recognition of some fine blood vessels areas, resulting in incomplete blood vessel segmentation results. TransFusionNet can accurately segment the liver tumor regardless of its integrity or vascular continuity.

TABLE IV
COMPUTATIONAL PERFORMANCE COMPARISON OF ORIGINAL AND QUANTIZED MODELS

	Node with 3090			JetsonTX2		
	Latency (s)	IoU	TFLOPs	Latency(s)	IoU	GFLOPs
TransFusionNet	1.802	0.884	3.997	55.037	0.884	130.872
quantizied	1.720	0.855	4.188	54.495	0.855	132.174

E. Computational Performance Analysis

In this experiment, we analyzed the computational performance of the model on node with 3090 and JetsonTX2 device. We selected the same patient sample containing 132 CT slices to run 5 times inference tests, and obtain the mean IoU and latency of the segmentation. We then used $fvcore$ to calculate the FLOP of TransFusionNet, which is about 54 GFLOP. Based on the latency and FLOP, we calculated the peak performance of both models on different devices. Table IV shows the detailed results of our experiments.

According to the results in Table IV, we found that compared to the original model, the quantized model latency is reduced by 0.08 s on node with 3090 and 0.54 s on JetsonTX2. Peak computational performance is improved by 0.191 TFLOPs on node with 3090 and 1.302 GFLOPs on JetsonTX2. The computational efficiency on JetsonTX2 reached 21%. Furthermore, mean IoU is reduced by 0.03 which is negligible. Interestingly, we found that the improvement in inference performance of the quantized model on the JetsonTX2 device is not very significant. Further analysis reveals that the access memory of our model in the device occupies about 5 GB, which is close to the storage limit of JetsonTX2 and thus hinders the inference performance improvement of JetsonTX2.

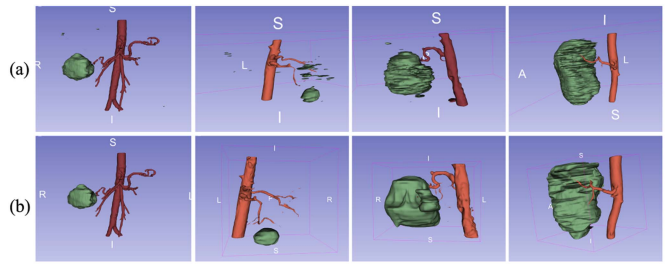

Fig. 9. Visual comparison of reconstruction results. (a) Reconstruction result using automatic reconstruction algorithm under JetsonTX2 system. (b) Reconstruction results using manual annotations.

F. Visual Analysis of 3D Reconstruction of Segmentation Results

To validate the effectiveness of TransFusionNet, we conducted a 3D visualization study of the segmentation model. Following the methods in Section II-H, we selected the test set of liver cancer patient samples applying the lightweight model on JetsonTX2 to make inference predictions and save the results. Based on the predicted results, we used 3DSlicer to display their 3D reconstructions. Meanwhile, we invited physicians to manually annotate the test cases and construct ground-truth 3D reconstruction results. Fig. 9 shows a comparison between the reconstruction results under embedded microprocessor and manual annotations of a typical patient. Except for some noise and loss of vessel details, the reconstruction results were very close to the actual annotation results. However, a detailed manual annotation requires a lot of time and effort, which significantly reflect the efficiency and accuracy of our proposed algorithm.

IV. Conclusion

In this work, we proposed TransFusionNet which can effectively extract the semantic and spatial features of CT images and achieve precise segmentation. Furthermore, we proposed intelligent quantization scheme and implement the model on JetsonTX2. The IoU reached 0.854 in the vessel segmentation and 0.927 in the liver tumor segmentation. Compared with the state-of-the-art segmentation methods, TransFusionNet has an accuracy improvement of 0.01–0.02. Meanwhile, model inference achieved the peak performance of 132GFLOPs under JetsonTX2. Our experiment is used for the segmentation of liver tumors and blood vessels, but TransFusionNet can also be applied to the segmentation of other tissues. The unprecedented accuracy and efficiency of TransFusionNet, as realized in this work by combining multiple feature extraction methods and implement on JetsonTX2, Brings a new idea to the development of future smart medical devices.

Although we have completed the segmentation of liver tumors and blood vessels under JetsonTX2, the accuracy and speed of the of inference still needs to be further improved. Due to the numerous and small branches of intrahepatic vessels, it is difficult for deep learning algorithm to perceive the characteristics of intrahepatic vessels. At the same time, there is a certain accuracy gap between the quantized model and the TranfusionNet which may lead loss of the key tissue feature. In our next work, we will investigate the model quantification method combining numerical analysis and reinforcement learning to improve the inference speed of the model with guaranteed accuracy. In addition, we will investigate unsupervised or semi-supervised segmentation based algorithms to reduce the workload of data annotation.

References

[1] X. Li, P. Ramadori, D. Pfister, M. Seehawer, L. Zender, and M. Heikenwalder, "The immunological and metabolic landscape in primary and metastatic liver cancer," *Nature Rev. Cancer*, vol. 21, no. 9, pp. 541–557, 2021.

[2] D. A. Gervais, S. N. Goldberg, D. B. Brown, M. C. Soulen, S. F. Millward, and D. K. Rajan, "Society of interventional radiology position statement on percutaneous radiofrequency ablation for the treatment of liver tumors," *J. Vasc. Interventional Radiol.*, vol. 20, no. 7, pp. S342–S347, 2009.

[3] M. Ciecholewski and M. Kassjański, "Computational methods for liver vessel segmentation in medical imaging: A review," *Sensors*, vol. 21, no. 6, 2021, Art. no. 2027.

[4] H. Jiang, T. Shi, Z. Bai, and L. Huang, "AHCNet: An application of attention mechanism and hybrid connection for liver tumor segmentation in ct volumes," *IEEE Access*, vol. 7, pp. 24898–24909, 2019.

[5] Y. Cheng, X. Hu, J. Wang, Y. Wang, and S. Tamura, "Accurate vessel segmentation with constrained b-snake," *IEEE Trans. Image Process.*, vol. 24, no. 8, pp. 2440–2455, Aug. 2015.

[6] M. Chung, J. Lee, J. W. Chung, and Y.-G. Shin, "Accurate liver vessel segmentation via active contour model with dense vessel candidates," *Comput. Methods Programs Biomed.*, vol. 166, pp. 61–75, 2018.

[7] C. Bauer, T. Pock, E. Sorantin, H. Bischof, and R. Beichel, "Segmentation of interwoven 3d tubular tree structures utilizing shape priors and graph cuts," *Med. Image Anal.*, vol. 14, no. 2, pp. 172–184, 2010.

[8] S. Esneault, C. Lafon, and J.-L. Dillenseger, "Liver vessels segmentation using a hybrid geometrical moments/graph cuts method," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 2, pp. 276–283, Feb. 2010.

[9] M. A. Lebre, A. Vacavant, M. Grand-Brochier, H. Rositi, and B. Magnin, "Automatic segmentation methods for liver and hepatic vessels from CT and MRI volumes, applied to the couinaud scheme," *Comput. Biol. Med.*, vol. 110, pp. 42–51, 2019.

[10] J. N. Kaftan, H. Tek, and T. Aach, "A two-stage approach for fully automatic segmentation of venous vascular structures in liver ct images," in *Proc. Int. Soc. Opt. Eng.*, 2009, Art. no. 725911.

[11] Y.-Z. Zeng et al., "Liver vessel segmentation and identification based on oriented flux symmetry and graph cuts," *Comput. Methods Programs Biomed.*, vol. 150, pp. 31–39, 2017.

[12] D. Mahapatra, "Analyzing training information from random forests for improved image segmentation," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1504–1512, Apr. 2014.

[13] A. Smith, "Image segmentation scale parameter optimization and land cover classification using the random forest algorithm," *J. Spatial Sci.*, vol. 55, no. 1, pp. 69–79, 2010.

[14] X. Wanga, T. Wang, and J. Bua, "Color image segmentation using pixel wise structural-support-vectormachine (S-SVM) classification," *Pattern Recognit.*, vol. 44, no. 4, pp. 777–787, 2011.

[15] Y. Zhiwen, H. Wong, and W. Guihua, "A modified support vector machine and its application to image segmentation [j]," *Image Vis. Comput.*, vol. 29, no. 1, pp. 29–40, 2011.

[16] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[17] X. Meng, X. Li, and X. Wang, "A computationally virtual histological staining method to ovarian cancer tissue by deep generative adversarial networks," *Comput. Math. Methods Med.*, vol. 2021, 2021, Art. no. 4244157.

[18] S. Qiao et al., "A pseudo-siamese feature fusion generative adversarial network for synthesizing high-quality fetal four-chamber views," *IEEE J. Biomed. Health Inform.*, early access, Jan. 14, 2022, doi: 10.1109/JBHI.2022.3143319.

[19] S. Qiao, S. Pang, G. Luo, S. Pan, T. Chen, and Z. Lv, "FLDS: An intelligent feature learning detection system for visualizing medical images supporting fetal four-chamber views," *IEEE J. Biomed. Health Inform.*, to be published, doi: 10.1109/JBHI.2021.3091579.

[20] S. Qiao et al., "RLDS: An explainable residual learning diagnosis system for fetal congenital heart disease," *Future Gener. Comput. Syst.*, vol. 128, pp. 205–218, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X21003915

[21] X. Li, P. Han, G. Wang, W. Chen, S. Wang, and T. Song, "SDNN-PPI: Self-attention with deep neural networks effect on protein-protein interaction prediction," *BMC Genomic.*, vol. 23, no. 1, 2022, Art. no. 474.
[22] X. Wang et al., "Imgg: Integrating multiple single-cell datasets through connected graphs and generative adversarial networks," *Int. J. Mol. Sci.*, vol. 23, no. 4, 2022, Art. no. 2082.
[23] F. Meng, D. Xu, and T. Song, "ATDNNS: An adaptive time–frequency decomposition neural network-based system for tropical cyclone wave height real-time forecasting," *Future Gener. Comput. Syst.*, vol. 133, pp. 297–306, 2022.
[24] J. Wang et al., "De novo molecular design with deep molecular generative models for PPI inhibitors," *Brief. Bioinf.*, vol. 23, no. 4, 2022, Art. no. bbac285.
[25] T. Song, R. Zhang, Y. Dong, X. Tao, H. Lu, and B. Liu, "Mesh2Measure: A novel body dimensions measurement based on 3D human model," in *Proc. Int. Conf. Intell. Technol. Interactive Entertainment*, 2021, pp. 80–99.
[26] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 3431–3440.
[27] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.- Assist. Interv.*, 2015, pp. 234–241.
[28] T. Song, W. Wei, F. Meng, J. Wang, R. Han, and D. Xu, "Inversion of ocean subsurface temperature and salinity fields based on spatio-temporal correlation," *Remote Sens.*, vol. 14, no. 11, 2022, Art. no. 2587.
[29] Q. Huang, J. Sun, H. Ding, X. Wang, and G. Wang, "Robust liver vessel extraction using 3D U-net with variant dice loss function," *Comput. Biol. Med.*, vol. 101, pp. 153–162, 2018.
[30] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet : A nested u-net architecture for medical image segmentation," in *Proc. Deep Learn. Med. Image Anal. Multimodal Learn. Clin. Decis. Support*, 2018, pp. 3–11.
[31] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5998–6008.
[32] P. Shanchen, Z. Ying, S. Tao, Z. Xudong, W. Xun, and R.-P. Alfonso, "AMDE: A novel attention-mechanism-based multidimensional feature encoder for drug–drug interaction prediction," *Brief. Bioinf.*, vol. 23, no. 1, 2022, Art. no. bbab545.
[33] S. Tao, Z. Xudong, D. Mao, R.-P. Alfonso, W. Shudong, and W. Gan, "DeepFusion: A deep learning based multi-scale feature fusion method for predicting drug-target interactions," *Methods*, vol. 90, pp. 11–12, 2022.
[34] X. Zhang et al., "Molormer: A lightweight self-attention-based method focused on spatial structure of molecular graph for drug–drug interactions prediction," *Brief. Bioinf.*, vol. 23, 2022, Art. no. bbac296.
[35] G. Wang et al., "Multi-TransDTI: Transformer for drug–target interaction prediction based on simple universal dictionaries with multi-view strategy," *Biomolecules*, vol. 12, no. 5, 2022, Art. no. 644.
[36] X. Wang, Z. Zhang, C. Zhang, X. Meng, X. Shi, and P. Qu, "TransPhos: A deep-learning model for general phosphorylation site prediction based on transformer-encoder architecture," *Int. J. Mol. Sci.*, vol. 23, no. 8, 2022, Art. no. 4263.
[37] J. Chen et al., "Transunet: Transformers make strong encoders for medical image segmentation," 2021, *arXiv:2102.04306*.
[38] C. Li et al., "ANU-Net: Attention-based nested u-net to exploit full resolution features for medical image segmentation," *Comput. Graph.*, vol. 90, pp. 11–20, 2020.
[39] D. Jha, M. A. Riegler, D. Johansen, P. Halvorsen, and H. D. Johansen, "DoubleU-Net: A deep convolutional neural network for medical image segmentation," in *Proc. IEEE 33rd Int. Symp. Comput.-Based Med. Syst.*, 2020, pp. 558–564.
[40] T. Song, F. Meng, A. Rodriguez-Paton, P. Li, P. Zheng, and X. Wang, "U-next: A novel convolution neural network with an aggregation U-net architecture for gallstone segmentation in CT images," *IEEE Access*, vol. 7, pp. 166823–166832, 2019.
[41] C. Wang, L. Gong, Q. Yu, X. Li, Y. Xie, and X. Zhou, "DLAU: A scalable deep learning accelerator unit on FPGA," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 36, no. 3, pp. 513–517, Mar. 2017.
[42] P. Qi et al., "Accommodating transformer onto FPGA: Coupling the balanced model compression and FPGA-implementation optimization," in *Proc. Great Lakes Symp. VLSI*, 2021, pp. 163–168.
[43] Z. Zhao, R. Cao, K.-F. Un, W.-H. Yu, P.-I. Mak, and R. P. Martins, "An FPGA-based transformer accelerator using output block stationary dataflow for object recognition applications," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, early access, Aug 03, 2022, doi: 10.1109/TCSII.2022.3196055.
[44] M. Zhou, W. Xu, J. Kang, and T. Rosing, "TransPIM: A memory-based acceleration via software-hardware co-design for transformer," in *Proc. IEEE Int. Symp. High- Perform. Comput. Architect.*, 2022, pp. 1071–1085.
[45] W. Xiang, H. Mao, and V. Athitsos, "ThunderNet: A turbo unified network for real-time semantic segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, 2019, pp. 1789–1796.
[46] H. Huang, Y. Wu, M. Yu, X. Shi, and X. Liu, "EDSSA: An encoder-decoder semantic segmentation networks accelerator on opencl-based FPGA platform," *Sensors*, vol. 20, no. 14, 2020, Art. no. 3969.
[47] Y. Ma, Y. Cao, S. Vrudhula, and J.-S. Seo, "Optimizing the convolution operation to accelerate deep neural networks on FPGA," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 7, pp. 1354–1367, Jul. 2018.
[48] T.-H. Tsai, Y.-C. Ho, and M.-H. Sheu, "Implementation of fpga-based accelerator for deep neural networks," in *Proc. IEEE 22nd Int. Symp. Des. Diagn. Electron. Circuits Syst.*, 2019, pp. 1–4.
[49] A. Dosovitskiy et al., "An image is worth 16x16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
[50] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 630–645.
[51] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7132–7141.
[52] T. Takikawa, D. Acuna, V. Jampani, and S. Fidler, "Gated-SCNN: Gated shape CNNs for semantic segmentation," *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 5228–5237.
[53] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 4470–4479.
[54] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier, "Language modeling with gated convolutional networks," in *Proc. 34th Int. Conf. Mach. Learn.*, 2017, pp. 933–941.
[55] K. Wang, Z. Liu, Y. Lin, J. Lin, and S. Han, "Haq: Hardware-aware automated quantization with mixed precision," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 8612–8620.
[56] T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*.
[57] H. Huang et al., "Unet 3+: A full-scale connected unet for medical image segmentation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, 2020, pp. 1055–1059.