Neurocomputing 512 (2022) 466–481

Contents lists available at ScienceDirect

Neurocomputing

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

Combining multi-view ensemble and surrogate lagrangian relaxation for real-time 3D biomedical image segmentation on the edge

Shanglin Zhou [a], Xiaowei Xu [b,c,*], Jun Bai [a], Mikhail Bragin [d]

[a] Department of Computer Science and Engineering, University of Connecticut, USA
[b] Department of Cardiovascular Surgery, Guangdong Provincial People's Hospital, Guangdong Academy of Medical Sciences, Guangzhou, China, 510080
[c] Guangdong Provincial Key Laboratory of South China Structural Heart Disease, Guangdong Cardiovascular Institute, Guangzhou, China, 510080
[d] Department of Electrical and Computer Engineering, University of Connecticut, USA

ARTICLE INFO

Article history:
Received 15 January 2022
Revised 22 July 2022
Accepted 4 September 2022
Available online 17 September 2022
Communicated by Zidong Wang

2010 MSC:
00–01
99–00

Keywords:
Medical image segmentation
Deep model compression
Surrogate lagrangian relaxation
Edge processing

ABSTRACT

Real-time 3D biomedical image segmentation is always preferred considering the exponentially growing medical imaging data for the past decade. Recently deep learning has significantly boosted the performance of automatic medical image segmentation with high computation and memory requirements, especially for 3D biomedical images. Meanwhile, the privacy and security of patient data have always been the primary concern in medical applications among hospitals and clinics, and there also exists some applications which need real-time processing in clinic practice. Thus, 3D biomedical image segmentation is typically required to be performed locally (i.e. on the edge) with limited computation and memory resources. In this paper, we propose to combine multi-view ensemble and Surrogate Lagrangian relaxation (SLR) for real-time 3D biomedical image segmentation on the edge. Instead of directly dealing with 3D biomedical images, our segmentation conducts on the three 2D domains of the 3D images with an ensemble strategy. In addition, Surrogate Lagrangian relaxation is proposed to compress the model to enable high efficiency and real-time processing. Experiments on a typical edge Nvidia GPU show that our method achieves real-time processing which is 1.5× faster with an improvement of 9% on accuracy compared with single-view models. It also saves 26× computational resources and 6× memory resources compared to 3D segmentation models.

1. Introduction

With pervasive medical imaging applications in health care, biomedical image segmentation has always been one of the most important tasks in biomedical imaging research [1,2]. Biomedical image segmentation extracts different tissues, organs, pathologies, and biological structures, to support medical diagnosis, surgical planning, and treatments. In common practice, pathologists and radiologists perform segmentation manually, which is time-consuming and tedious, especially for 3D images. The problem becomes more prominent in terms of cost and reproducibility considering the exponentially growing medical imaging data for the past decade [3–5]. Therefore, automatic real-time biomedical image segmentation is highly desirable.

Recently deep learning has significantly boosted the performance of automatic medical image segmentation [6,1,7,8,6,9–14]. However, on one hand, performing deep learning computation for such an application usually requires extremely high computation cost [15]. For example, segmenting a 3D Computed Tomography (CT) volume with a typical neural network 3D U-Net [2] would involve around 2.2 Tera (10^{12}) high precision floating point operations, taking days for such processing on a desktop-level computer [16]. And for computing on the cloud, 3D U-Net still takes about a hundred milliseconds to segment such a CT image [16]. On the other hand, deep learning models usually stack layers with millions of parameters, which requires high memory. U-Net [17], the most widely used deep convolutional neural network in medical image segmentation, has 30*M* parameters with a model size being 386*MB*. SegNet [18], another popular but smaller medical image segmentation model, has 30*M* parameters with a model size being 117*MB*. Some other models derived from these two methods, such as R2U-Net [19], which combines residual block and recurrent convolution to replace the original sub-module in U-Net, contain 39*M* parameters; and Attention UNet [20], which adds attention mechanism, contains 34*M* parameters.

* Corresponding author.
E-mail address: xiao.wei.xu@foxmail.com (X. Xu).

https://doi.org/10.1016/j.neucom.2022.09.039
0925-2312/

Meanwhile, the privacy and security of patient data have always been the primary concern in medical applications among hospitals and clinics [21,22. In addition, there are also some applications such as real-time ultrasound quality control and diagnosis in the clinic [23,24. As such, protocols typically require medical image processing tasks such as denoising, segmentation, and diagnosis to be performed locally, i.e., *on the edge*. However, local machines and devices are usually with rather limited computation resources including computing and memory capacity compared with those in the cloud [15]. The constrained resources brings many challenges to the design of medical image segmentation algorithms, e.g., the algorithms need to process the 3D images in time (real-time if possible) while with limited computing and memory resources.

Currently, most related works focus on real-time segmentation without consideration of constraint resources on the edge. Fradi et al. [23] performed real-time bone image segmentation using ultrasonic Computed Tomographic images. Li et al. [25] proposed automatic 2D tongue image segmentation for real-time remote diagnosis. Hu et al. [26] used real-time tumor margin identification for image-guided robotic brain tumor resection based on 2D MRI images. Islam et al. [27] performed real-time instrument segmentation in robotic surgery using auxiliary supervised deep adversarial learning based on 2d camera-captured images. Xie et al. [28] proposed near real-time hippocampus segmentation using a patch-based canonical neural network based on 2D images. Jha et al. [29] explored real-time polyp segmentation in video capsule endoscopy and colonoscopy. Anas et al. [30] proposed a deep learning approach for real-time prostate segmentation in 2D freehand ultrasound-guided biopsy. Jami et al. [31] performs melanoma segmentation on 2d skin images in the scenario of mobile health. There are also some works about resource efficient networks for real-time segmentation, which, however, focus on only 2D images. Xu et al. [32] performed mobile telemedicine with compressed cellular neural networks for real-time 2D X-ray segmentation. Ni et al. [33] proposed an attention-guided lightweight network for real-time segmentation of robotic surgical instruments based on camera captured 2D images. Zhou et al. [34] proposed a lightweight attention encoder–decoder network for 2D ultrasound image segmentation.

In this paper, different from exiting works, we focus on 3D biomedical image segmentation which can achieve real-time processing on the edge. Particularly, we propose to combine multi-view ensemble and Surrogate Lagrangian Relaxation (SLR) [35]. With multi-view ensemble, we split the three-dimensional images into a series of two-dimensional images in three different planes and apply 2D segmentation model, respectively to balance the relationship between segmentation speed and accuracy. For each single-view model, we further propose the SLR-based weight pruning method [36] to improve the running speed while keeping the model resource-efficient. Experiments on a typical edge Nvidia GPU show that our method achieves real-time processing which is 1.5× faster with an improvement of 9% on accuracy compared with single-view models. It also saves 26× computational resources and 6× memory resources compared to 3D segmentation models.

We summarize our contributions as:

- To the best of our knowledge, this is the first work that explores real-time 3D biomedical image segmentation considering both computation and memory requirement on the edge;
- We propose to combine multi-view ensemble and Surrogate Lagrangian relaxation for real-time 3D biomedical image segmentation;
- We propose SLR-based model compression technique to enable real-time processing;
- We have conducted various experiments, and results show that real-time segmentation can be achieved with little accuracy loss.

2. Related work

2.1. Medical image segmentation

Inspired by the U-Net architecture [17], fully convolutional networks (FCNs) have dominated biomedical image segmentation. DCAN [37], for example, created a contour recognition decoding branch for unified multi-task learning with well-defined object boundaries. Active learning has been intensively investigated to alleviate the work of human annotations. Based on uncertainty and similarity estimation, Suggestive Annotation [38] actively picked the most representative examples to alleviate the potential limit from the datasets. MILD-Net [39] created a complex structure by including a minimal information loss unit to compensate for data loss during downsampling. [40] presented a two-stream chained segmentation approach that effectively fuses the CT and PET modalities via early and late 3D deep-network-based fusion. Meanwhile, [41] designed multi-stage architecture and attention blocks to deal with small areas segmentation in WCE image. Also, deep CNNs with dense blocks have been applied to obtain high performance using a 3D segmentation from MRIs [42]. As for vessel segmentation, more and more methods pay attention to the multi-scale context information to improve the segmentation of the thick vessels and thin vessels. For example, [43] introduced a pyramid scale aggregation block to aggregate coarse-to-fine context information in each layer of the network. [44] separated the segmentation of thick vessels and thin vessels in different brunches to balance potential imbalance between them.

2.2. Multi-view based segmentation

Multi-view-based segmentation has been widely used in a variety of applications. For instance, [45] presented a shape-aware multi-view autoencoder, which exploits the spatial context from the long-axis images to guide the segmentation on the short-axis images and achieves accurate and robust segmentation of the myocardium to improve the robustness of cardiac segmentation, outperforming baseline models (e.g., 2D U-Net, 3D U-Net) while achieving higher data efficiency. Based on a multi-view convolutional neural network, CardiacNET [46] created a method with an adaptive fusion strategy and a new loss function strategy, which shown to greatly improve the segmentation accuracy on the existing benchmark for LA and proximal pulmonary veins (PPVs) segmentation, can need for ablation therapy planning and clinical guidance in atrial fibrillation (AF) patients. Automatic segmentation of brain images is also a mainstream topic in recent years. [47] presented a semi-supervised algorithm and a DCNNs architecture for the 3D MRI whole brain segmentation problem, which is designed to exploit a large amount of unlabeled data, while the goal of the novel architecture is to enlarge the receptive field of the model and utilize more structure information of brains. The same is true of other medical disciplines. MV-CNN [48], for example, created a multi-view convolutional neural network for lung nodule segmentation, which can segment various types of nodules, including juxta-pleural, cavitary, and non-solid nodules. The results show encouraging performance. [49] proposed a deep-learning-based automated method for Multiple Sclerosis (MS) lesion segmentation is presented, which can help diagnosis and patient follow-up while reducing the time-consuming need for manual segmentation. Fetal ultrasound (US) is the primary imaging modality to monitor fetal development, [50] proposed a method to extract the human placenta at late gestation using

multi-view 3D US images based on 3D convolutional neural networks. It is a fully automatic method for extracting whole placenta volumes at late gestation, with a dice overlap of 0.8 and placental volumes comparable to MR. Aiming at the perfection and strengthening of methodology, [51] proposed a new multi-view spatial aggregation framework for joint localization and segmentation of multiple OARs using H&N CT images, which to iteratively improve the segmentation accuracy and consistency within and across image slices. As for microvascular networks segmentation, [52] used a suitable image analysis workflow to handle vessel segmentation from light-sheet fluorescence microscopy (LSFM) data in very large tissue volumes. They provided a systematic analysis of multi-view deconvolution image processing workflow to control and evaluate the accuracy of the reconstructed vascular network using various low to high level, metrics, to achieve sufficient for a reliable quantitative 3-D vessel segmentation for their possible use for perfusion modeling. 3DMV [53] presented a novel method for 3D semantic scene segmentation of RGB-D scans in indoor environments using a joint 3D-multi-view prediction network, the final result on the ScanNet 3D segmentation benchmark [54] increases from 52.8% to 75% accuracy compared to existing volumetric architectures.

2.3. Model compression

As deep neural networks (DNNs) are more and more widely studied and explored, DNNs are getting larger and heavier that not only take a long time to train, but also need a large space to store. Many mathematical investigations have demonstrated that there exists a significant margin of redundancy in DNNs across filters and channels [55–57]. Several prior works have focused on reducing weight storage using the weight pruning technique while keeping negligible accuracy loss. One representative work is [55]. It uses a three-step method to iteratively prune the unimportant weights, cut redundant connections in DNNs, and then retrain the DNN to recover accuracy. 9× weight reduction is achieved on AlexNet for the ImageNet dataset without accuracy degradation. As indices are needed in this work to locate which weight to pruning, it has been extended in several works [58]. For instance, [59] proposed an energy efficiency-aware pruning method that aimed to facilitate energy-efficient hardware implementations with certain accuracy degradation. [60] proposed a structured sparsity learning technique that partially overcomes the problem of irregular structure of the network after pruning. [61] employed an evolutionary algorithm for weight pruning that incorporated randomness in both pruning and growing of weights. However, all these prior weight pruning techniques are either highly heuristic or need a long retraining phase to recover the accuracy. [62] proposed a systematic framework that achieves a faster convergence rate and higher compression ratio. [36] further improved this method by leveraging Surrogate Lagrangian Relaxation [35], which further accelerates the convergence speed and even enables retrain-free model compression.

3. Proposed method

In this section, we will describe the detail of our segmentation framework, including the U-Net based multi-view ensemble, and the proposed SLR for weight pruning. The overall structure is summarized in Fig. 1.

As shown in Fig. 1, we firstly parse the 3D biomedical image (e.g., a cardiac CT image) into three 2D domains: axial view, sagittal view, and coronal view [46]. Each view is a series of 2D images. Then, the same CNN architecture is applied on each view for the pixel-wise segmentation. Because the problem is constrained from the 3D domain to the 2D domain, the computational burden is sufficiently reduced. Model compression technique is also leveraged on all the three CNN models to further enable real-time segmentation. The resulting segmentation output of each view is combined through a fusion strategy, which is designed to maximize the information content from different views as well as correct the information captured by different models. The details of the model structure, the model compression technique, and the fusion operation are given in the following subsections.

3.1. Multi-view ensemble

U-Net based Segmentation Model. The leveraged U-Net [1] based segmentation CNN model is shown in Fig. 2. ResNet-18 [63] is used as the backbone model. The entire network structure is constructed based on the ResNetUNet structure from Usuyama et al. [64]. As shown in Fig. 2, ResNet-18 blocks in the left half are inserted into the architecture for encoder purposes to reduce the image dimensions and extract feature representations. All ResNet-18 blocks are formulated as *Conv* → *BatchNorm* → *ReLu* → *Conv* → *BatchNorm* with skip connection, and two blocks are connected using ReLu (rectified linear unit) Layer. The ConvReLu blocks on the right half perform decoder roles, with goals being semantically project the learned discriminative features onto the pixel space. Two blocks are connected using upsampling layers to convert the images to their original sizes. Concatenation between the higher resolution features from the encoder network and the corresponding upsampled features from the decoder network enables the network to learn representations more complicated.

Loss Function. For the loss function, Chosen loss functions have a major role in deep network image segmentation. Recently, cross-entropy has been combined with other loss functions in CNN-based image segmentation and classifications to obtain high performance [65,66]. In this work, we have used the weighted sum of binary cross-entropy loss and dice loss [9] as our loss function, as shown in Eq. 1. Binary cross-entropy loss is used because of our binary class segmentation problem. Dice loss is fused in because of its ability of handling the data with imbalanced class. It is developed from the Sørensen-Dice coefficient [67,68], which measures the relative overlap between the prediction and the ground truth, and becomes a widely used metric in the computer vision community to calculate the similarity between two images [69]. As shown in Eq. 2, y_{true} and y_{pred} respectively represent pairs of corresponding pixel values of ground truth (target mask) and prediction. The numerator is the sum of correctly predicted pixels. It is the overlap between prediction and ground truth, so it considers loss information locally. The denominator is the sum of total pixels of the predicted mask and the ground truth mask, so it considers the loss information globally. These together lead to high accuracy as loss information is considered locally and globally.

$$Loss = [BCE \times bce\ weight] + [Dice \times (1 - bce\ weight)] \tag{1}$$

$$Dice\ Loss = 1 - \frac{2\sum_{pixels} y_{true} \times y_{pred}}{\sum_{pixels} y_{true}^2 + \sum_{pixels} y_{pred}^2} \tag{2}$$

3.2. Surrogate lagrangian relaxation for weight pruning

In order to reduce the model size and achieve real-time segmentation, surrogate Lagrangian Relaxation-based (SLR-based) model compression technique is adopted.

Consider an N-layer deep neural network. The weights at each convolutional layer are denoted as $\mathbf{W}_i$, and the collection of biases

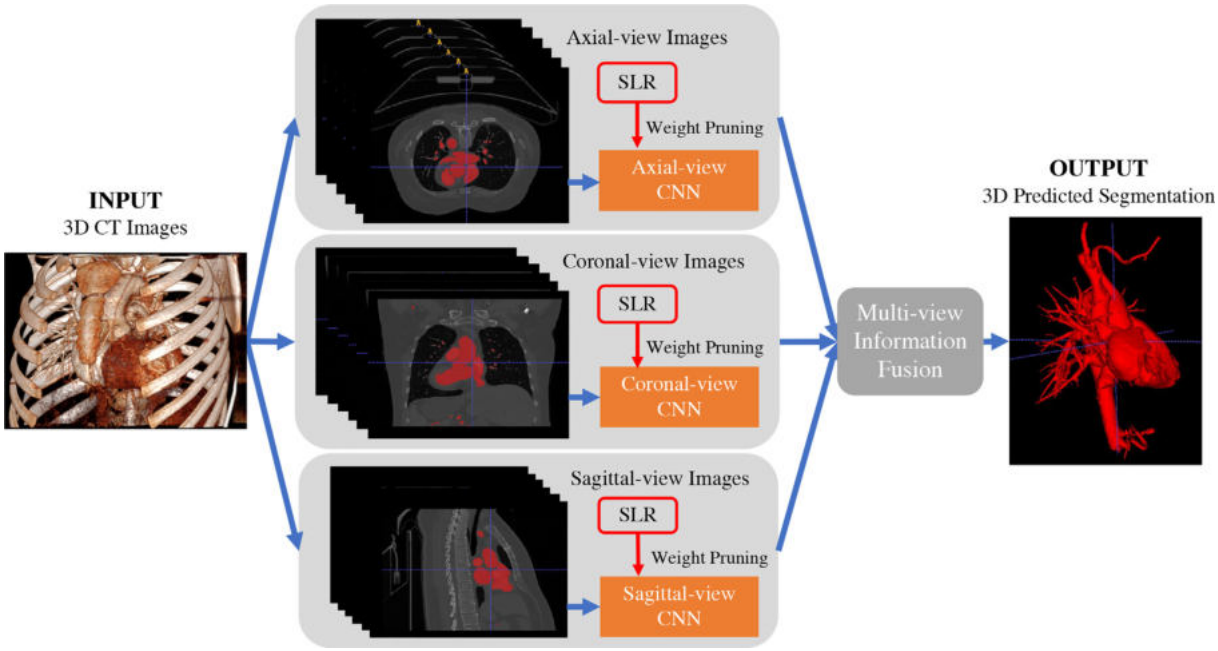

Fig. 1. Overview of the proposed training procedure. With the input being a series of 3D images, we first split the 3D images into 2D images through three different views, and feed them to three single-view 2D models. For each single-view model, SLR weight pruning is applied for lighter weight models. After the three single-view prediction results are obtained, we use the multi-view information fusion strategy to get the 3D predicted segmentation result.

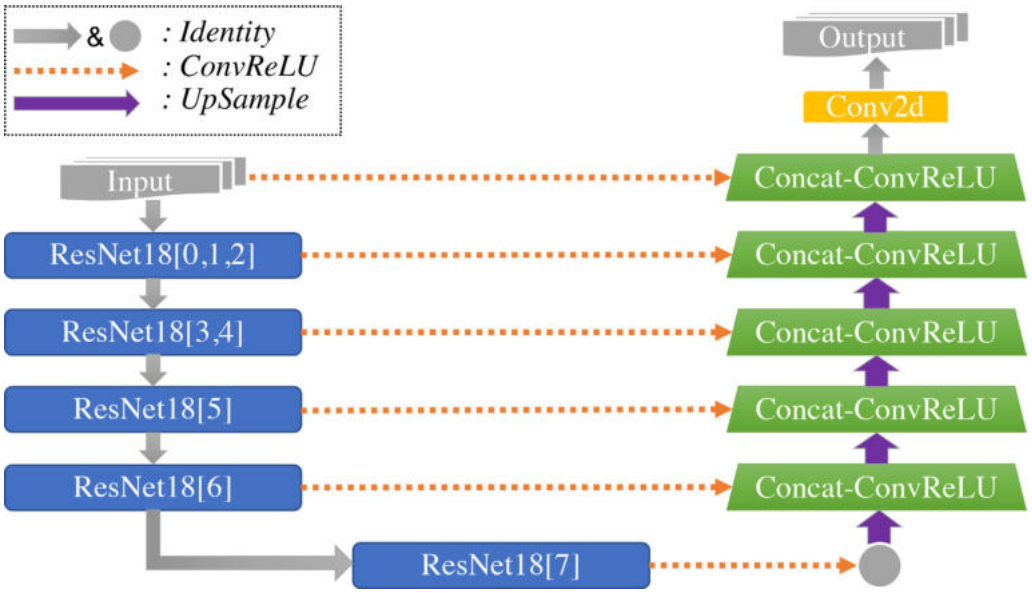

Fig. 2. Details of the U-Net based segmentation model. We use ResNet-18 as the backbone of the U-Net from [1]. ResNet18[i] means the i-th block of ResNet-18 architecture.

in each layer is denoted as $\mathbf{b}_i$, where $i \in 1, 2, \ldots, N$. Then the loss function can be written as $f(\{\mathbf{W}_i\}_{i=1}^N, \{\mathbf{b}_i\}_{i=1}^N)$. The objective of irregular weight pruning can be done by minimizing the loss function and making it subject to constraints on the cardinality of weights in each convolution layer. This can be formulated as Eq. 3, where the constraint restricts the number of nonzero elements in the matrix $\mathbf{W}_i$ being less than α_i, which is the desired number of weights in the i-th layer of the DNN model:

$$\begin{aligned} &\text{minimize}_{\{\mathbf{W}_i\},\{\mathbf{b}_i\}} \quad f(\{\mathbf{W}_i\},\{\mathbf{b}_i\}), \\ &\text{subject to} \qquad\quad \text{card}(\mathbf{W}_i) \leqslant \alpha_i, \quad i = 1, \ldots, N. \end{aligned} \tag{3}$$

This can further be equivalently rewritten in an unconstrained form as Eq. 4:

$$\text{minimize}_{\{\mathbf{W}_i\},\{\mathbf{b}_i\}} \left\{ f(\{\mathbf{W}_i\},\{\mathbf{b}_i\}) + \sum_{i=1}^{N} h_i(\mathbf{W}_i) \right\}. \tag{4}$$

The first term of Eq. 4 represents the nonlinear loss function, and the second represents the non-differentiable penalty term [62] that $h_i(.)$ is the indicator function shown in Eq. 5:

$$h_i(\mathbf{W}_i) = \begin{cases} 0 & \text{if} \text{card}(\mathbf{W}_i) \leqslant \alpha_i, \\ +\infty & \text{otherwise}. \end{cases} \tag{5}$$

Clearly, the problem cannot be solved only analytically or only using stochastic gradient descent because of the non-differentiable part. Duplicate variables [70] are introduced to enable the decomposition into smaller manageable subproblems and the problem is rewritten as Eq. 6:

$$\begin{aligned} &\text{minimize}_{\{\mathbf{W}_i\},\{\mathbf{b}_i\}} \quad f(\{\mathbf{W}_i\},\{\mathbf{b}_i\}) + \sum_{i=1}^{N} h_i(\mathbf{Z}_i), \\ &\text{subject to} \qquad\quad \mathbf{W}_i = \mathbf{Z}_i, \quad i = 1,\ldots,N \end{aligned} \tag{6}$$

To solve the problem, SLR leverages the Lagrangian multipliers Λ_i to relax the constraints $\mathbf{W}_i = \mathbf{Z}_i$ (where $\dim\Lambda_i = \dim\mathbf{W}_i$), and penalizes their violations using quadratic penalties with a positive scalar penalty coefficient ρ, as shown in Eq. 7 below.

$$\begin{aligned} L_\rho(\mathbf{W}_i,\mathbf{b}_i,\mathbf{Z}_i,\Lambda_i) &= f(\mathbf{W}_i,\mathbf{b}_i) + \sum_{i=1}^{N} h_i(\mathbf{Z}_i) \\ &+ \sum_{i=1}^{N} \mathrm{tr}\left[\Lambda_i^T(\mathbf{W}_i - \mathbf{Z}_i)\right] + \sum_{i=1}^{N} \frac{\rho}{2}\|\mathbf{W}_i - \mathbf{Z}_i\|_F^2, \end{aligned} \tag{7}$$

where $\|.\|_F^2$ denotes the Frobenius norm and $tr(.)$ denotes the trace. The relaxed problem can be decomposed into two subproblems and being solved iteratively until convergence.

SubProblem 1: Solve "loss-function" subproblem for $\mathbf{W}_i$ using Stochastic Gradient Decent. At iteration t, the objective of the "loss-function" subproblem is minimized by keeping $\mathbf{Z}_i$ at previously obtained values $\mathbf{Z}_i^{t-1}$, that is, the Lagrangian function for given values of multipliers Λ_i^t is minimized: $\min_{\mathbf{W}_i,\mathbf{b}_i} L_\rho\left(\mathbf{W}_i,\mathbf{b}_i,\mathbf{Z}_i^{t-1},\Lambda_i\right)$. Stochastic gradient decent (SGD) [71] can be leveraged to solve this subproblem as the loss function is differentiable. At this point, the following "surrogate" optimality condition in Eq. 8 needs to be satisfied to ensure that multipliers are updates along "proper" directions:

$$L_\rho\left(\mathbf{W}_i^t,\mathbf{b}_i^t,\mathbf{Z}_i^{t-1},\Lambda_i^t\right) < L_\rho\left(\mathbf{W}_i^{t-1},\mathbf{b}_i^{t-1},\mathbf{Z}_i^{t-1},\Lambda_i^t\right). \tag{8}$$

As demonstrated in [35], the above condition ensures that Lagrangian values approach dual values; consequently, multiplier-updating directions (violation levels of relaxed constraints $\mathbf{W}_i^t = \mathbf{Z}_i^{t-1}$) approach subgradient directions, which, in turn, form acute angles with directions toward optimal multipliers. As a result, when multipliers are updated along the directions $\mathbf{W}_i^t = \mathbf{Z}_i^{t-1}$:

$$\Lambda_i^{t+\frac{1}{2}} := \Lambda_i^t + s^{t-\frac{1}{2}}\left(\mathbf{W}_i^t - \mathbf{Z}_i^{t-1}\right) \tag{9}$$

by using "appropriate" stepsizes [35],

$$s^{t-\frac{1}{2}} = \alpha^t \frac{s^{t-1}\|\mathbf{W}^{t-1} - \mathbf{Z}^{t-1}\|}{\|\mathbf{W}^t - \mathbf{Z}^{t-1}\|} \tag{10}$$

the multipliers asymptotically approach optimal multipliers. If the "surrogate" optimality condition is not satisfied, previous step-sizes and multipliers are kept.

SubProblem 2: Solve "Cardinality" problem for $\mathbf{Z}_i$ through weight pruning using Projections onto Discrete Subspace. The cardinality subproblem can be written as $\min_{\mathbf{Z}_i} L_\rho\left(\mathbf{W}_i^t,\mathbf{b}_i^t,\mathbf{Z}_i,\Lambda_i^{t+1}\right)$, that being solved with respect to $\mathbf{Z}_i$ by fixing other variables at $\mathbf{W}_i$. As $h_i(.)$ is indicator function, the global optimal of this subproblem can be obtained analytically using Eq. 11 [70], where $\Pi_{\mathbf{S}_i}(.)$ is the Euclidean projection onto $\mathbf{S}_i = \{\mathbf{W}_i | \mathrm{card}(\mathbf{W}_i) \leqslant \alpha_i\}, i = 1,\ldots,N$.

$$\mathbf{Z}_i^t = \Pi_{\mathbf{S}_i}\left(\mathbf{W}_i^t + \frac{\Lambda_i^{t+1}}{\rho}\right) \tag{11}$$

Similar as subProblem 1, the second "surrogate" optimality condition, as shown in Eq. 12, needs to be satisfied at this point to ensure multipliers updating to "proper" directions.

$$L_\rho\left(\mathbf{W}_i^t,\mathbf{b}_i^t,\mathbf{Z}_i^t,\Lambda_i^{t+\frac{1}{2}}\right) < L_\rho\left(\mathbf{W}_i^t,\mathbf{b}_i^t,\mathbf{Z}_i^{t-1},\Lambda_i^{t+\frac{1}{2}}\right) \tag{12}$$

Step-sizes and multipliers are updated again as Eq. 13 when Eq. 12 is satisfied.

$$\begin{aligned} s^t &= \alpha^t \frac{s^{t-\frac{1}{2}}\|\mathbf{W}^{t-1} - \mathbf{Z}^{t-1}\|}{\|\mathbf{W}^t - \mathbf{Z}^t\|} \\ \Lambda_i^{t+1} &:= \Lambda_i^{t+\frac{1}{2}} + s^t\left(\mathbf{W}_i^t - \mathbf{Z}_i^t\right) \end{aligned} \tag{13}$$

Same as the first step, previous step-sizes and multipliers are kept if the condition is not satisfied. A fractional index $\frac{1}{2}$ was adopted within (9) to indicate that only a half of an iteration is complete, and the "full" update is complete after both subproblems are solved in (13). In both steps, parameter for the step-sizes are generically formalized as Eq. 14, where M and r are predefined hyper-parameters:

$$\alpha^t = 1 - \frac{1}{M \times t^{\left(1-\frac{1}{t^r}\right)}}, \quad M > 1,\ 0 < r < 1 \tag{14}$$

3.3. Multi-view information fusion

Because each model individually gives its prediction based on the data in that view, we expected different segmentation accuracy in different views. To deal with this, we fuse the prediction from each of the views by leveraging the majority vote strategy. As shown in Fig. 3, each single-view model would predict out a 3D matrix. We use transform and padding to ensure the three matrices are of the same dimension. We then use the majority vote method, setting the pixel being one when more than two single-view models segment that pixel as objects on that pixel, or set it to zero when less or equal than one single-view model segments it out. The final prediction result would be one 3D matrix that fuses the predictions from the three single-view models. It is shown in Section 4 that our ensemble method greatly improves the segmentation accuracy.

4. Experiments

In this section, we demonstrate the effectiveness of our method with experiments, including description of our dataset, experimental setting, and results.

4.1. Dataset

The adopted dataset [72] consists of 220 3D CT images captured by a Simens biograph 64 machines. The ages of the associated patients range from 1 month to 21 years, with the majority between 1 month and 2 years. The size of the images is $512 \times 512 \times$ (130–340), and the typical voxel size is $0.25 \times 0.25 \times 0.5$ mm^3. The dataset covers 14 types of CHD. All labeling were performed by experienced radiologists, and the time for labeling each image is 1–1.5 h. The labels include seven substructures: left ventricle (LV), right ventricle (RV), left atrium (LA), right atrium (RA), myocardium (Myo), aorta (Ao), and pulmonary artery (PA). Note that the area including RA, LA, LV, RV, PA, and Ao is defined as blood pool. For easy processing, venae cavae (VC) and pulmonary vein (PV) are also labeled as part of RA and LA, respectively, as they are connected, and their boundaries are relatively hard to define. Anomalous vessels are also labeled as one of the above seven substructures based on their connections.

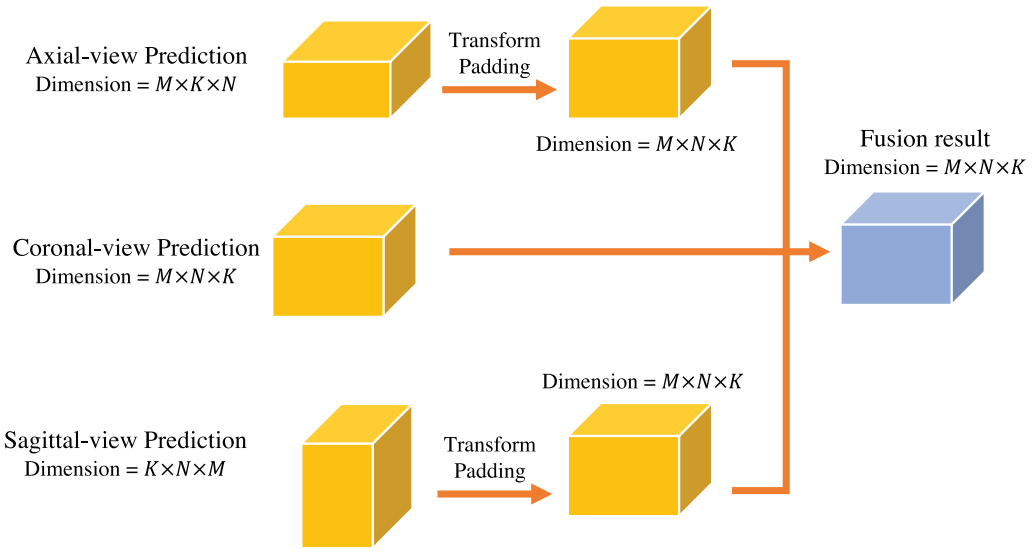

Fig. 3. Details of the fusion method.

In all the experiments, we leave one 3D image for testing. Then among the remaining 219 images, 90% of the images are used for training while 10% are used for validation.

4.2. Implementation detail

For evaluation purpose, we use the Dice Index as our evaluation matrix. The equation is similar as Eq. 2 and is shown in Eq. 15.

$$Dice\ Index = \frac{2\sum_{pixels} y_{true} \times y_{pred}}{\sum_{pixels} y_{true}^2 + \sum_{pixels} y_{pred}^2} \tag{15}$$

In the optimizing process, images and segmentation masks from all three views are resized to 256×256. For all the three models, we use Adam [73] with cosine decay learning rate strategy [74] with the learning rate initialized as $1e-3$, and batch size as 64.

SLR pruning process is then applied to all three models that trained from the three views' data. During SLR training, SLR parameters are set as $M = 300, r = 0.1, s0 = 10^{-2}$ and $\rho = 0.1$. We also use learning rate as 0.001, batch size as 32 and Adam optimizer in the pruning process. All the experiments are conducted on Ubuntu 18.04, Python 3.7 and PyTorch v1.6.0 software version. Furthermore, we use Nvidia Quadro RTX 6000 GPU with 24 GB GPU memory for the training. Inference and performance testing are conducted on Nvidia Jetson TX2, which will be introduced later in Section 4.3.

4.3. Experimental results and discussions

In this part, we first show the performance of the three single-view models under different compression rates, and compare the running speed between models with compression and models without compression. The three single-view models are built on axial view, sagittal view, and coronal view, respectively. Nvidia Jetson TX2 is used to test the inference running speed. Jetson TX2 is a fast and power-efficient embedded AI computing device. It is a 7.5-watt platform that built around an NVIDIA Pascal-family GPU and loaded with 32 GB storage, 8 GB memory and 59.7GB/s memory bandwidth [75]. On TX2, we use Python 3.6.9 with Torch v1.3.0.

Table 1
SLR training results on the three view of dataset through different compression rates.

View	(%) Baseline	Compression Rate	(%) After Training	(%) After Hardpruning	(%) After Retraining
Axial	98.15	1.944×	97.64	97.63	97.82
		7.947×	91.81	91.86	95.49
		20.934×	90.50	90.54	91.50
Coronal	96.89	1.944×	96.72	96.72	96.72
		7.947×	90.55	90.48	94.26
		20.934×	88.81	88.60	89.77
Sagittal	96.53	1.944×	96.05	96.01	96.01
		7.947×	88.78	86.55	93.57
		20.934×	85.91	84.70	86.58

Table 2
Running speed (ms/img) on TX2 under different compression rates.

Compression Rate	Axial	Coronal	Sagittal
1×	73.46	65.02	58.09
1.944×	66.09	58.48	44.28
7.947×	58.39	48.53	39.46
20.934×	53.59	36.22	35.86

4.3.1. Results of single-view models with SLR pruning

Table 1 shows the prediction accuracy of the three single-view models with and without model compression, and the accuracy of each model under different compression ratios. "Baseline" means the accuracy of models without applying compression. All three models have a baseline greater than 96%. The axial-view model has a prediction accuracy being 98.15%, which means this view can predict more accurately than the other two views, providing more information when ensemble the three results together. For SLR weight pruning, results for all compressed models are reported after 100 epochs of training and 10 epochs of masked retraining. In the table, the compression rate is defined as the division of the size of the uncompressed model by the size of the compressed model. It measures the relative reduction in the size of the model after performing a model compression algorithm on it [76].

Table 3
Accuracy (%) of the single-view models and ensemble multi-view models under different compression rate.

Compression Rate	Axial	Coronal	Sagittal	Ensemble
1×	95.76	94.08	93.98	96.58
1.944×	95.70	93.88	93.72	96.54
7.947×	93.07	90.65	90.83	94.81
20.934×	79.94	83.48	80.37	88.39

As shown in the table, when the compression ratio is 1.944×, there is less than 0.5% accuracy drop for all three models. Especially for the coronal-view model, there is only 0.17% accuracy difference, which can be ignored. When the compression ratio is 7.947×, there is 2.66% accuracy drop for the axial-view model, 2.63% accuracy drop for the coronal-view model, and 2.96% accuracy drop for the sagittal-view model. And when the compression ratios are up to 20.934×, there are 6.65%, 7.12% and 9.95% accuracy drop for the axial, coronal, and sagittal-view models, respectively. Comparing all three compression ratios, the axial-view model is the most robust one compared with the other two as it not only provide the highest prediction accuracy, but also has minimal accuracy loss under different compression ratios. In contrast, the sagittal-view model is more vulnerable compared with the other two as it has the most accuracy loss and provides the lowest prediction accuracy under all three compression ratios.

Such phenomenon may due to two reasons. First, the pixel size of the 2D images in the axial view is usually isotropic in the two directions (e.g., $0.25mm \times 0.25mm$). However, the pixel size of that in the coronal view and the sagittal view is anisotropic, which introduces difficulty in the segmentation. Second, the features in the sagittal view are more difficult than those in the other two views. There are two main parts of the blood pool: large blood pool islands including the heart chambers and large great vessels, and small blood pool islands including small great vessels and other anomalous small vessels. In the axial view and the coronal view,

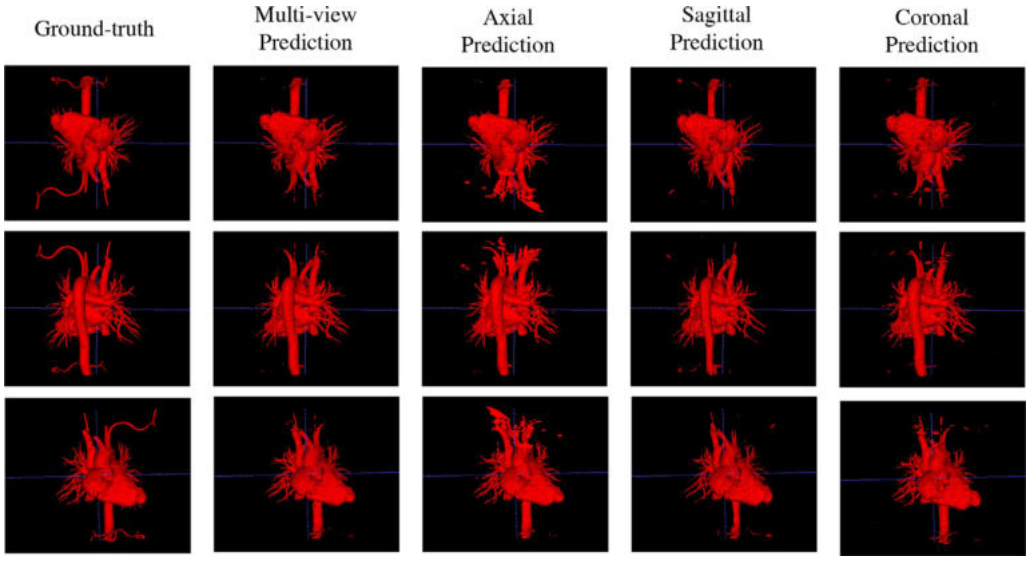

Fig. 4. 3D surface visualization for the ground-truth and the output generated by the proposed method (without pruning) and also from only S, C, and A views.

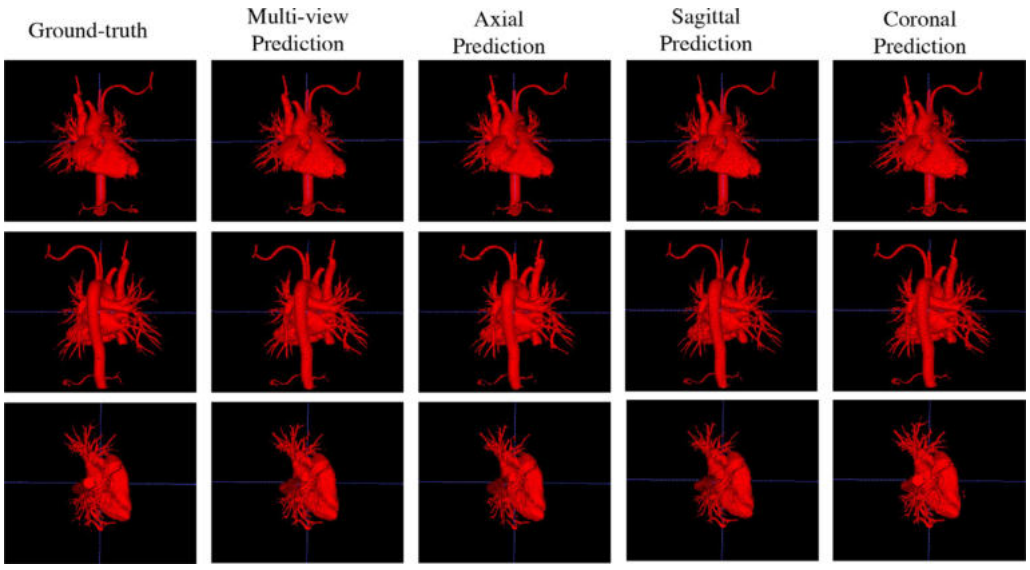

Fig. 5. 3D surface visualization for the ground-truth and the output generated by the proposed method with compression rate being 1.944× and also from only S, C, and A views.

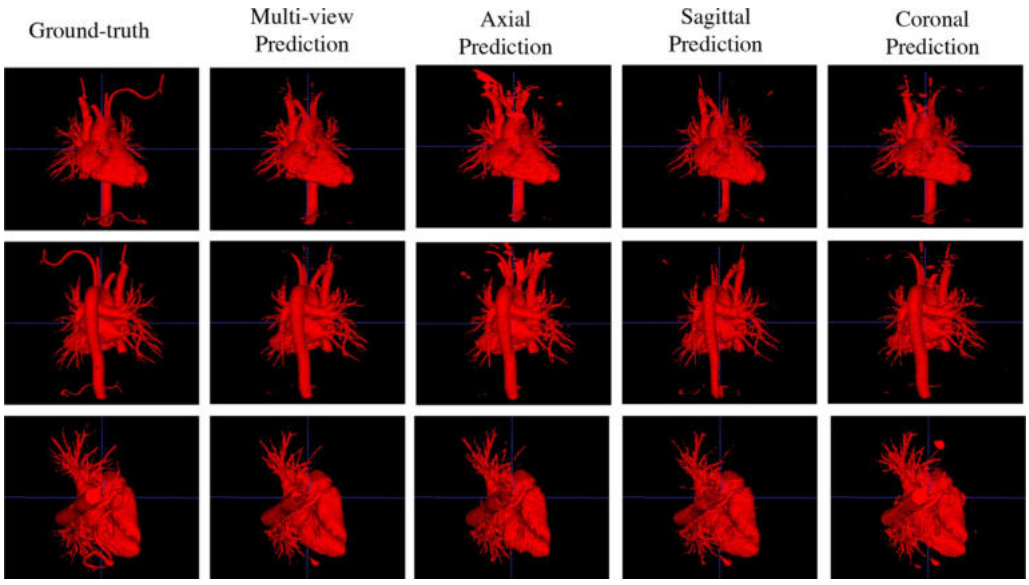

Fig. 6. 3D surface visualization for the ground-truth and the output generated by the proposed method with compression rate being 7.947× and also from only S, C, and A views.

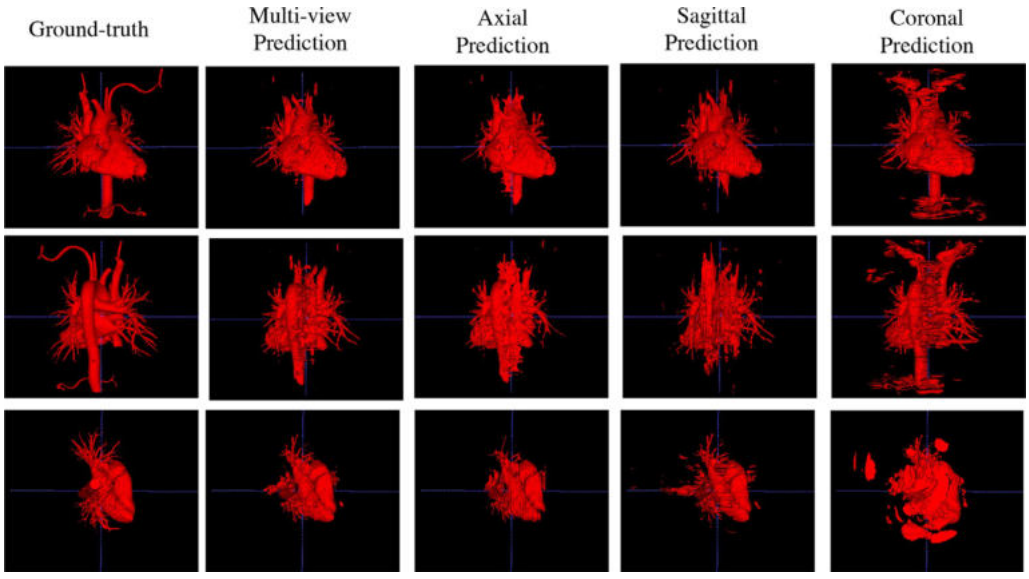

Fig. 7. 3D surface visualization for the ground-truth and the output generated by the proposed method with compression rate being 20.934× and also from only S, C, and A views.

most of the 2D images include both the large blood pool islands and the small blood pool islands. However, in the sagittal view, almost half of the 2D images only contain the small blood pool islands (mainly the small pulmonary vessels), while the other half only contains large blood pool islands (mainly the heart chambers). In general, with the SLR-based pruning method applied, there are very small accuracy losses in all three single-view models, which shows the effectiveness of adding this compression technique.

Table 4
Comparison of of the single-view models, ensemble multi-view models and 3D-UNet under different experimental metrics when image size is 128.

	Axial	Coronal	Sagittal	Ensemble	3D-UNet [6]
Dice score	0.7843	0.8191	0.7321	0.8956	0.9067
Precision	0.7843	0.8399	0.7533	0.8957	0.9297
Recall	0.6903	0.9046	0.7532	0.9502	0.9306
F1-score	0.9080	0.7839	0.7534	0.8471	0.9301
#Operations (GMAC)	23.78	23.78	23.78	71.59	1894.42

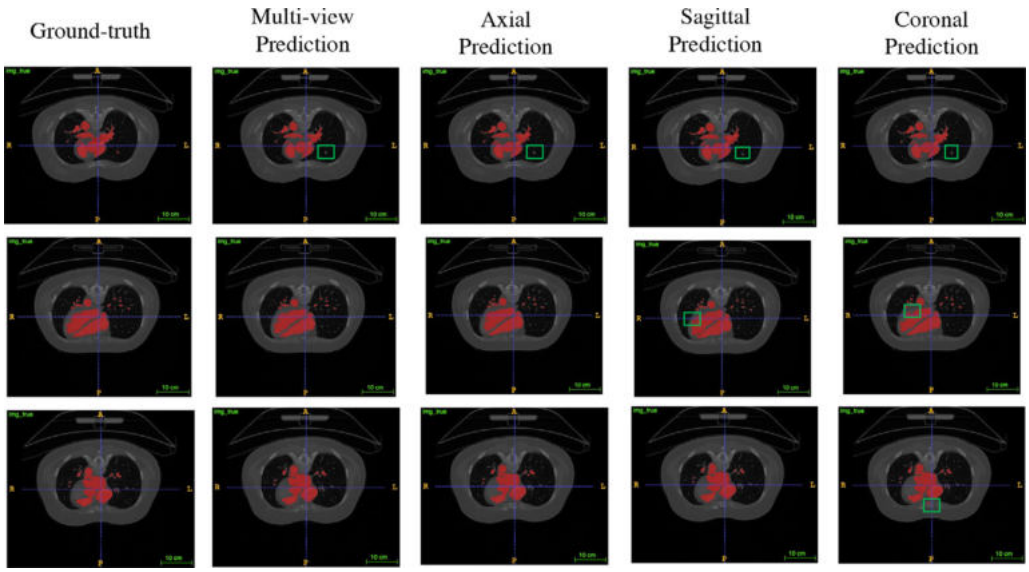

Fig. 8. Axial side visualization for the ground-truth and the output generated by the proposed method (without pruning) and also from only S, C, and A views.

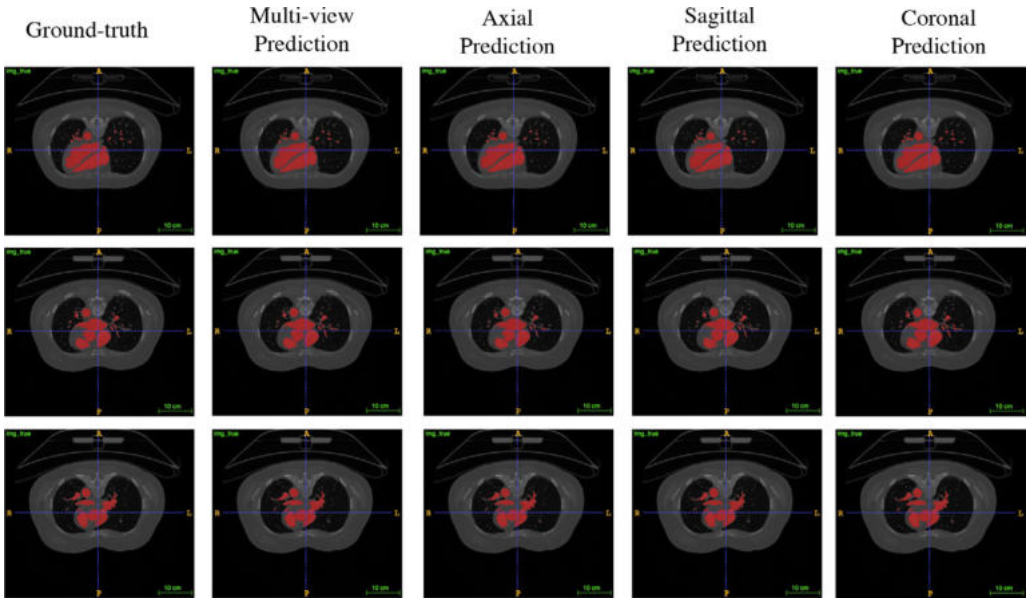

Fig. 9. Axial side visualization for the ground-truth and the output generated by the proposed method with compression rate being 1.944× and also from only S, C, and A views.

4.3.2. Comparison on performance of models with and without SLR pruning

Table 2 lists the inference running speed of each model on TX2. The speed is tested when models are under different compression ratios. In the table, 1× means the models are not applied with SLR compression. From the result, we can see that when there is no SLR compression applied, the axial-view model has the largest running speed as 73.46*ms*/*img*. Combining with the information from Table 1, the axial-view model has the highest accuracy when there is no compression, and also has the smallest accuracy loss when the compression ratio is 20.934×, these together prove that the axial-view model is the most robust one among all the three single-view models. Then for all the three single-view models, models with SLR compression technique applied can achieve approximately 1.5× faster than models without compression under compression ratio being 20.934×. This means that the SLR compression technique significantly improves the model efficiency. Moreover, we can see that under compression ratio being 20.934×, both the coronal-view model and the sagittal-view model can achieve a running speed of around 30*ms*/*img*. This leads to real-time segmentation and further verifies the importance of adding model compression to our segmentation model – this accuracy-speed trade-off proves the effectiveness of applying the SLR weight pruning. The difference in running speeds in the three views is because of the different image sizes. The size of a CT image is usually $512 \times 512 \times Z_0$ (Z_0 is usually between 200 and 300). In this case, the image size in the axial-view is 512×512, while in the coronal-view and sagittal-view are both $512 \times Z_0$, which are much smaller than that in the axial-view.

Table 3 compares the inference accuracy of the three single-view models with our multi-view models under different compression rates. 1× means no compression technique applied. As the

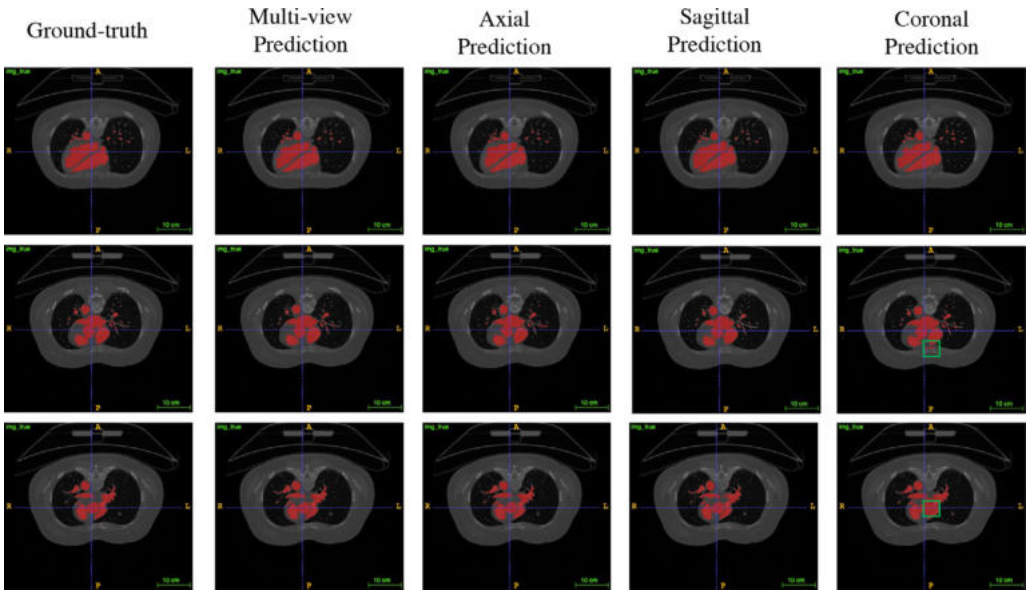

Fig. 10. Axial side visualization for the ground-truth and the output generated by the proposed method with compression rate being 7.947× and also from only S, C, and A views.

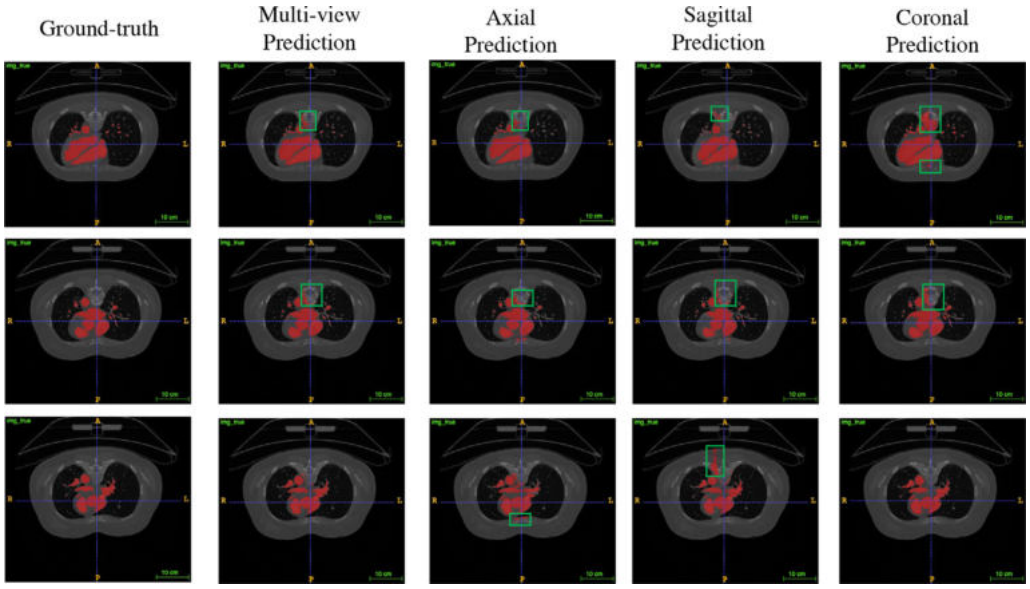

Fig. 11. Axial side visualization for the ground-truth and the output generated by the proposed method with compression rate being 20.934× and also from only S, C, and A views.

compression ratio increases for each single-view model, each model has a different degree of accuracy loss. The coronal-view model is more stable than the other two models as the accuracy dropped by around 10% points when the compression ratio increases to 20.934×, which is smaller compared with the axial-view model drops around 15% and the sagittal-view model drops around 13%. The ensemble multi-view models outperform all the three single-view models under all compression rates. Firstly, it achieves much higher accuracy than all the three single-view models regardless of the compression rate. When no compression technique is applied, the ensemble model improves the accuracy by around 2% on average. As the compression ratio increases to 20.934×, the accuracy improves by 7% to 88.39%. Secondly, as the compression ratio goes up from 1× to 20.934×, there is 8% decrease in the accuracy of the ensemble model, which is smaller than all the three single-view models.

Combining Table 2 and Table 3, we can see that for all the three single-view models, the running time does not decrease much when the compression ratio changes from 7.947× to 20.934×, and is accompanied by a significant drop in accuracy of almost 10%, however, the running time is reduced by a factor of 1.5 when the compression rate changes from 1× to 7.947×, and the accuracy drop only around 3%. This shows the effectiveness of applying the SLR pruning technique as it not only balances the relationship between accuracy and speed very well, but also achieves the so-called "real-time" medical image segmentation.

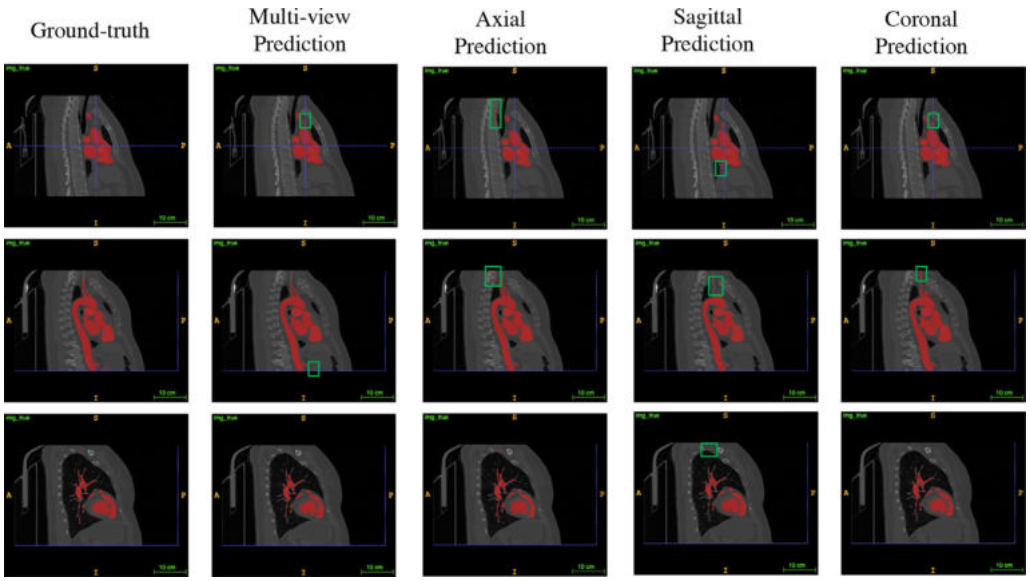

Fig. 12. Sagittal side visualization for the ground-truth and the output generated by the proposed method (without pruning) and also from only S, C, and A views.

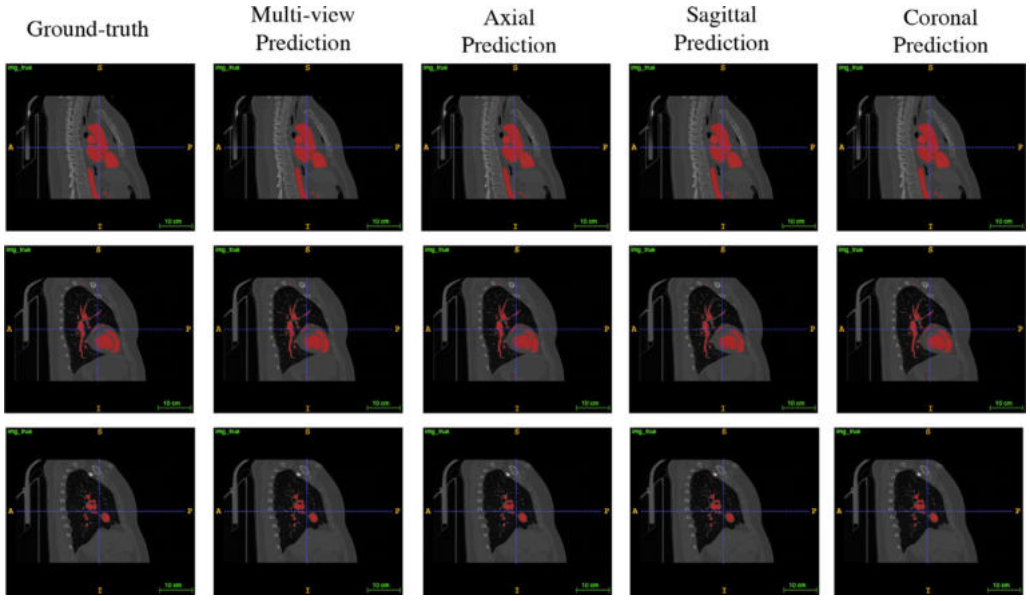

Fig. 13. Sagittal side visualization for the ground-truth and the output generated by the proposed method with compression rate being 1.944× and also from only S, C, and A views.

4.3.3. Visualization and comparison on segmentation results from models with and without SLR pruning

Example images and predicted segmentation results are presented for qualitative evaluation and comparison. Fig. 4–7 compare the ground truth 3D surface visualization output and results that generated by the proposed method and the three single-view models under different compression rates. For the result from models without weight pruning applied (Fig. 4), all the three single-view models cannot precisely segment every detail. However, the output from multi-view prediction, which fuses the three models' prediction, is more accurate than the ground truth. In Fig. 5 with the compression rate being 1.944×, coronal-view prediction in the first row does not precisely predict the blood vessel at the lower-left corner, but as both axial and sagittal-view give the right prediction, our multi-view prediction combines the information and segment it out. As the compression rate gets greater, as in Fig. 7, with the compression rate being 20.934×, it is difficult for the single-view models to predict very accurately under that high compression rate, then the advantages of the multi-view are becoming increasingly clear.

Single-view 2D segmentation results from the three single-view models under different compression rates are presented in A. We can see that predictions can be made only on the single side of images, but high accuracy can be maintained using the multi-view models, especially when the compression rate is large.

From the comparisons in these 2D and 3D figures, we can notice that the segmentation error comes from two sources: imprecise boundary segmentation and extra small islands. There are mainly

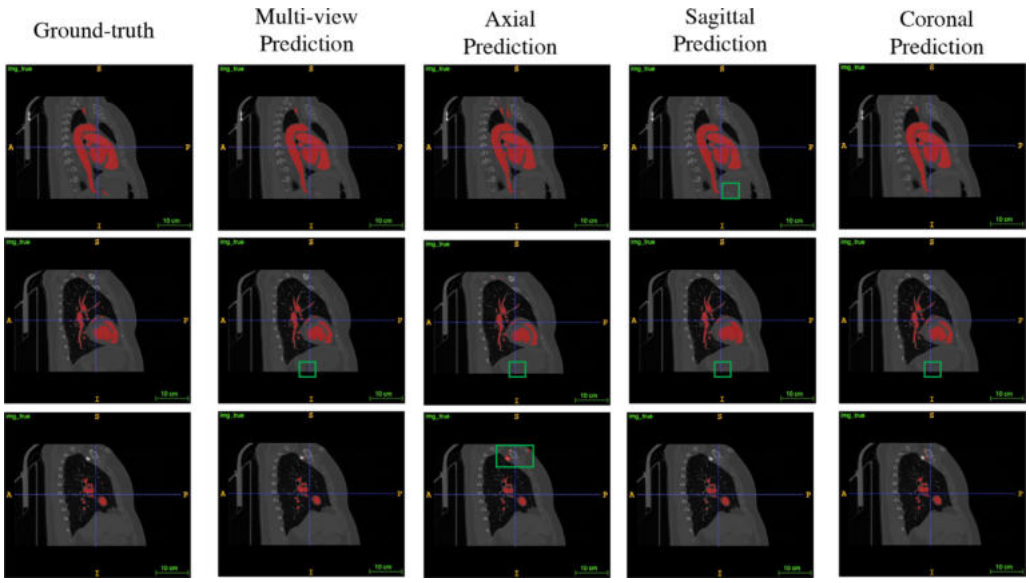

Fig. 14. Sagittal side visualization for the ground-truth and the output generated by the proposed method with compression rate being 7.947× and also from only S, C, and A views.

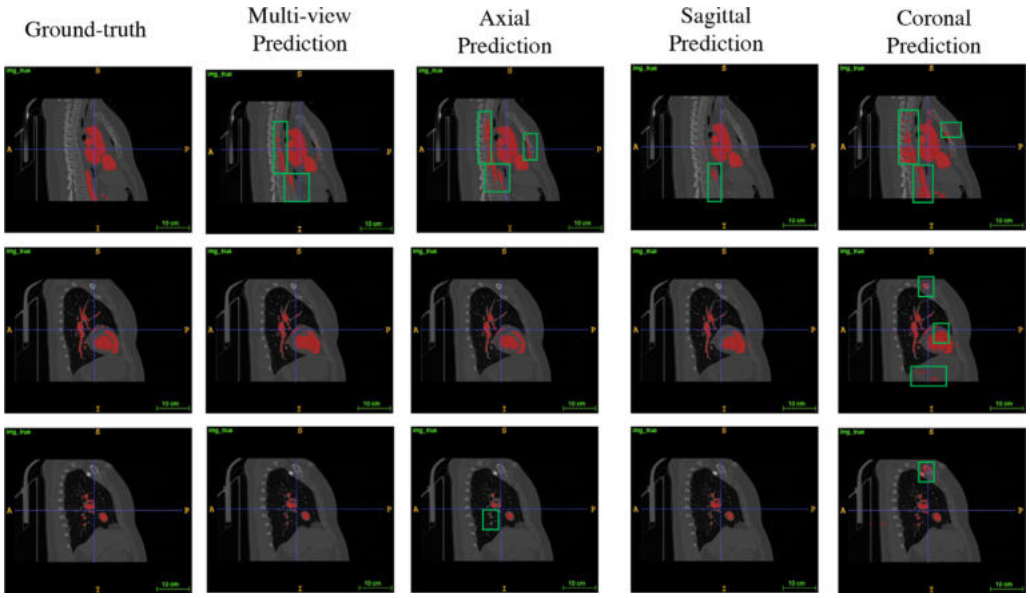

Fig. 15. Sagittal side visualization for the ground-truth and the output generated by the proposed method with compression rate being 20.934× and also from only S, C, and A views.

two reasons. First, the contrast of the CT images in the boundary part is rather low, making it hard to recognize even for radiologists. Note that this is also a widely recognized problem in the biomedical domain [1]. Second, 3D segmentation learning that only using 2D images may not fully exploit their correlation.

Moreover, CNN-based structures have some disadvantages in biomedical image tasks such as losing spatial relationships of learned features. To overcome this, capsule networks have been used recently in medical image classification tasks [77]. A combination of our proposed architecture with the capsule network will be left as our future work.

4.3.4. Comparison of multi-view ensemble model with 3D model

In this part, we evaluate our 2D multi-view ensemble method under segmentation metrics, which includes precision, recall, and F-1 score. We also compare the performance of the ensemble method with the 3D image segmentation method.

Table 4 compares the single-view models and the ensemble model with the 3D-UNet segmentation method under different evaluation metrics. Here, we do not apply the pruning technique to the three single-view models. The three evaluation metrics are calculated as in Eq. 16.

$$\begin{aligned} \text{Dice Score} &= \frac{2\times TP}{2\times TP+FP+FN} \\ \text{Precision} = \frac{TP}{TP+FP};\quad \text{Recall} &= \frac{TP}{TP+FN} \\ \text{F}-1 &= 2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \end{aligned} \tag{16}$$

where TP, TN, FT, and FN represent the number of true positive, true negative, false positive, and false negative, respectively. The threshold for precision, recall, and F-1 is 0.5 in all experiments. Note that

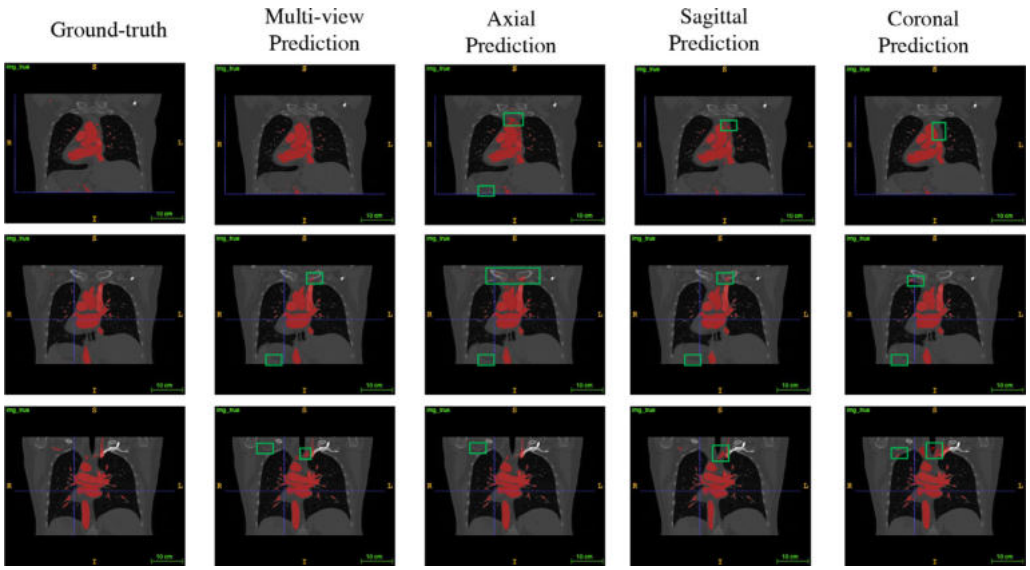

Fig. 16. Coronal side visualization for the ground-truth and the output generated by the proposed method (without pruning) and also from only S, C, and A views.

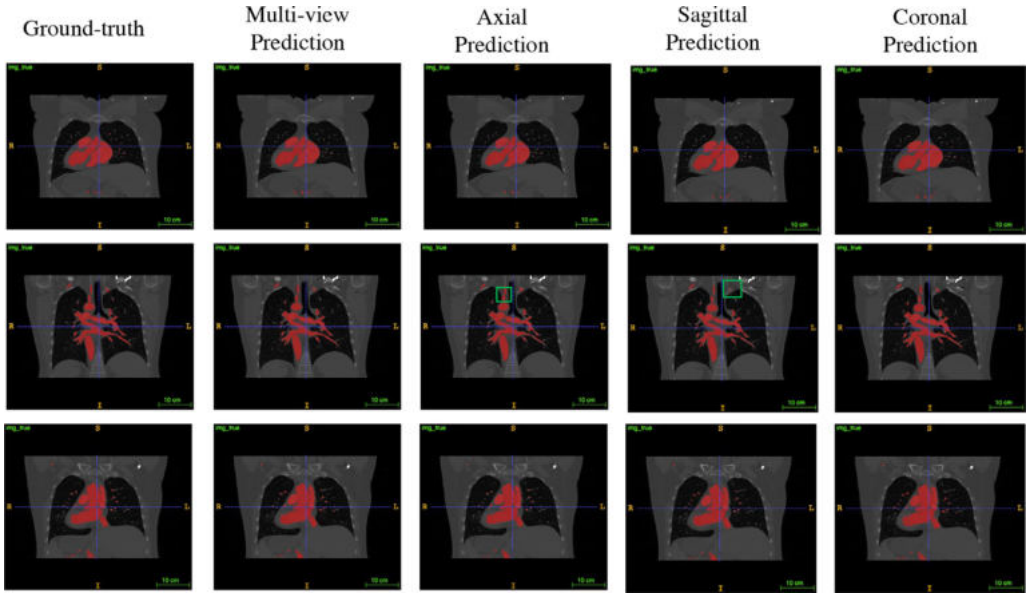

Fig. 17. Coronal side visualization for the ground-truth and the output generated by the proposed method with compression rate being 1.944× and also from only S, C, and A views.

because of the resource constraints, the 3D-UNet can only be trained when the 3D images are resized to $128 \times 128 \times 128$ under batch-size equals to 1. This not only lengthens the training time, but also affects the accuracy of the model. Even so, the model is still too large to be deployed on the Jetson TX2. In this case, we show the Giga multiply-accumulation operations (GMAC) of the models to show the performance. For a fair comparison, the single-view models are also trained under the image size being 128.

From the table, we can see that there is not much difference between the 3D model and the ensemble model under the four evaluation metrics. This indicates that our proposed method retains the accuracy very well. From the performance perspective, the 3D model is 26.5× more computationally intensive than the ensemble model (13.25× on running speed by estimation using the number of operations). Here, the three single-view models have the same GMAC because they share the same model structure but are trained under different data. The GMAC of the ensemble model is calculated as the sum of the three single-view models plus the GMAC of the majority vote process. This indicates that our ensemble model saves a lot of computational effort compared to the 3D model and would lead to much shorter running time while doing the inference. Similar to the result in Table 3, the ensemble model achieves a higher value than all the three single-view models in all the evaluation metrics except precision. For the three single-view models, the axial-view model gets the highest dice score and recall score, but gets the lowest precision and F-1 score among all the three models. This shows the axial-view model has higher *FP* compared with the other two models.

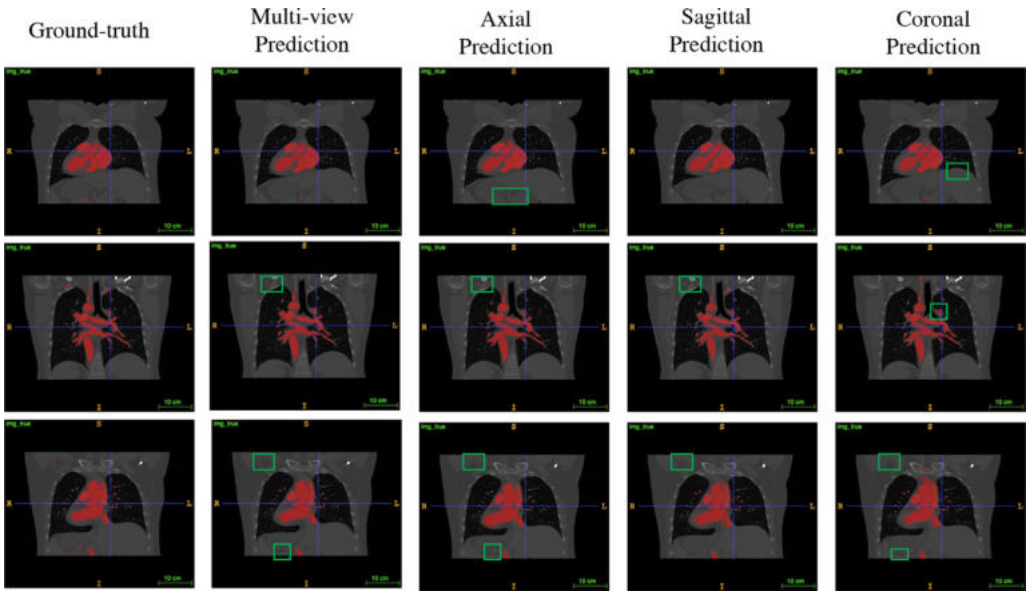

Fig. 18. Coronal side visualization for the ground-truth and the output generated by the proposed method with compression rate being 7.947× and also from only S, C, and A views.

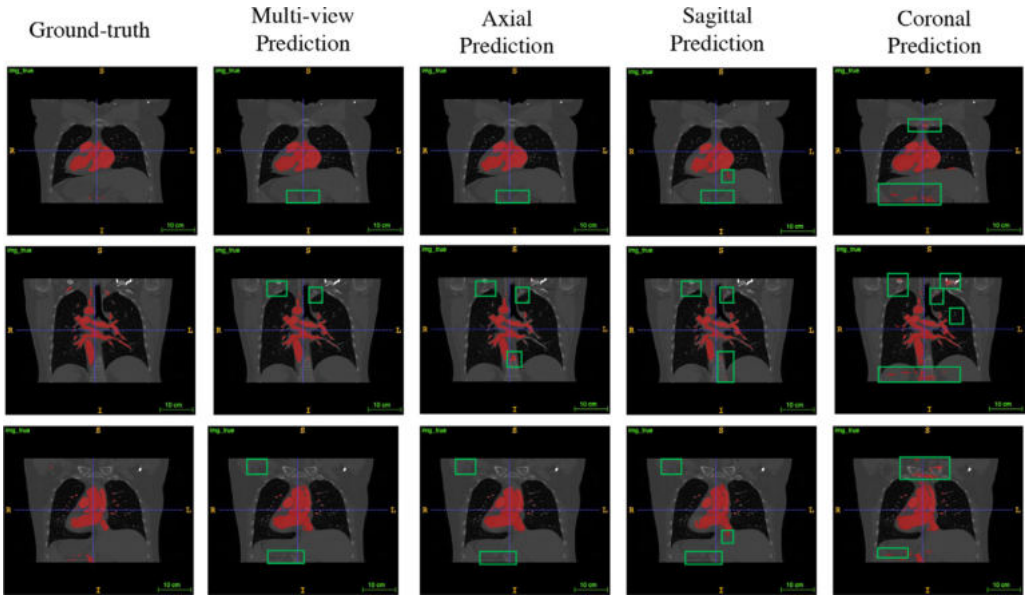

Fig. 19. Coronal side visualization for the ground-truth and the output generated by the proposed method with compression rate being 20.934× and also from only S, C, and A views.

5. Conclusion

In this paper, we propose to combine multi-view ensemble and Surrogate Lagrangian relaxation for real-time 3D biomedical image segmentation. In order to achieve real-time and balance the relationship between segmentation speed and accuracy, we split the three-dimensional medical images into a series of two-dimensional images in three different planes, i.e., axial planes, sagittal planes, and coronal planes, and apply 2D segmentation model, respectively. We further apply SLR-based weight pruning technique to reduce the model size and improve the running speed while keeping the performance of the model. Ensemble method is also applied for improvement of the segmentation accuracy. Experiments show that our ensemble model achieves 9% accuracy improves compared with single-view segmentation models. It also saves 26× computational resources and 6× memory resources compared to 3D segmentation model. With SLR weight pruning applied, the sagittal-view model can achieve real-time segmentation, while the axial and coronal-view models can achieve nearly real-time. This leads our ensemble model 1.5× faster with a very small accuracy loss compared with single-view models without compression.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was supported by the National Science Foundation of China (No. 62006050).

Appendix A. Single-view 2D segmentation results

In this appendix, we show the 2D segmentation result from the three single-view models under different compression rates, and compare them with the ground truth to visualize the performance of each single-view model. Fig. 8–11 show the prediction on axial view, Fig. 12–15 show the prediction on sagittal view, and Fig. 16–19 show the prediction on coronal view. They are all under different compression rates. The green boxes in these figures label the segmentation error between the current figure and the corresponding ground truth.

References

[1] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer, 2015, pp. 234–241.
[2] J. Chen, L. Yang, Y. Zhang, M. Alber, D.Z. Chen, Combining fully convolutional and recurrent neural networks for 3d biomedical image segmentation, in: Advances in neural information processing systems, 2016, pp. 3036–3044.
[3] I.D. Dinov, Volume and value of big healthcare data, Journal of medical statistics and informatics 4.
[4] J. Gerrity, Health networks – delivering the future of healthcare,https://www.buildingbetterhealthcare.co.uk/technical/article_page/Comment_Health_networks__delivering_the_future_of_healthcare/94931, 2014.
[5] A.A. Taha, A. Hanbury, Metrics for evaluating 3d medical image segmentation: analysis, selection, and tool, BMC medical imaging 15 (1) (2015) 1–28.
[6] Ö. Çiçek, A. Abdulkadir, S.S. Lienkamp, T. Brox, O. Ronneberger, 3d u-net: learning dense volumetric segmentation from sparse annotation, in: International conference on medical image computing and computer-assisted intervention, Springer, 2016, pp. 424–432.
[7] X. Xu, Q. Lu, L. Yang, S. Hu, D. Chen, Y. Hu, Y. Shi, Quantization of fully convolutional networks for accurate biomedical image segmentation, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 8300–8308.
[8] H. Chen, X. Qi, J.-Z. Cheng, P.-A. Heng, et al., Deep contextual networks for neuronal structure segmentation, AAAI (2016) 1167–1173.
[9] F. Milletari, N. Navab, S.-A. Ahmadi, V-net: Fully convolutional neural networks for volumetric medical image segmentation, in, fourth international conference on 3D vision (3DV), IEEE 2016 (2016) 565–571.
[10] H. Chen, Q. Dou, L. Yu, J. Qin, P.-A. Heng, Voxresnet: Deep voxelwise residual networks for brain segmentation from 3d mr images, NeuroImage.
[11] Q. Dou, H. Chen, Y. Jin, L. Yu, J. Qin, P.-A. Heng, 3d deeply supervised network for automatic liver segmentation from ct volumes, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2016, pp. 149–157.
[12] D.-P. Fan, T. Zhou, G.-P. Ji, Y. Zhou, G. Chen, H. Fu, J. Shen, L. Shao, Inf-net: Automatic covid-19 lung infection segmentation from ct images, IEEE Trans. Med. Imaging 39 (8) (2020) 2626–2637.
[13] D.-P. Fan, G.-P. Ji, T. Zhou, G. Chen, H. Fu, J. Shen, L. Shao, Pranet: Parallel reverse attention network for polyp segmentation, in: International conference on medical image computing and computer-assisted intervention, Springer, 2020, pp. 263–273.
[14] Z. Zhang, H. Fu, H. Dai, J. Shen, Y. Pang, L. Shao, Et-net: A generic edge-attention guidance network for medical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention Springer, 2019, pp. 442–450.
[15] X. Xu, Y. Ding, S.X. Hu, M. Niemier, J. Cong, Y. Hu, Y. Shi, Scaling for edge inference of deep neural networks, Nature Electron. 1 (4) (2018) 216.
[16] Z. Liu, X. Xu, T. Liu, Q. Liu, Y. Wang, Y. Shi, W. Wen, M. Huang, H. Yuan, J. Zhuang, Machine vision guided 3d medical image compression for efficient transmission and accurate segmentation in the clouds, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 12687–12696.
[17] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, Springer International Publishing, 2015.
[18] V. Badrinarayanan, A. Kendall, R. Cipolla, Segnet: A deep convolutional encoder-decoder architecture for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (12) (2017) 2481–2495.
[19] M.Z. Alom, C. Yakopcic, M. Hasan, T.M. Taha, V.K. Asari, Recurrent residual u-net for medical image segmentation, J. Med. Imaging 6 (1) (2019) 014006.
[20] O. Oktay, J. Schlemper, L.L. Folgoc, M. Lee, M. Heinrich, K. Misawa, K. Mori, S. McDonagh, N.Y. Hammerla, B. Kainz, et al., Attention u-net: Learning where to look for the pancreas, arXiv preprint arXiv:1804.03999.
[21] P.P. Ray, D. Dash, N. Kumar, Sensors for internet of medical things: State-of-the-art, security and privacy issues, challenges and future directions, Comput. Commun. 160 (2020) 111–131.
[22] M.S. Jalali, A. Landman, W.J. Gordon, Telemedicine, privacy, and information security in the age of covid-19, J. Am. Med. Inform. Assoc. 28 (3) (2021) 671–672.
[23] M. Fradi, E.-H. Zahzah, M. Machhout, Real-time application based cnn architecture for automatic usct bone image segmentation, Biomed. Signal Process. Control 71 (2022) 103123.
[24] J. Dong, S. Liu, Y. Liao, H. Wen, B. Lei, S. Li, T. Wang, A generic quality control framework for fetal ultrasound cardiac four-chamber planes, IEEE J. Biomed. Health Inform. 24 (4) (2019) 931–942.
[25] X. Li, D. Yang, Y. Wang, S. Yang, L. Qi, F. Li, Z. Gan, W. Zhang, Automatic tongue image segmentation for real-time remote diagnosis, 2019 IEEE international conference on bioinformatics and biomedicine (BIBM), IEEE 2019 (2019) 409–414.
[26] D. Hu, Y. Jiang, E. Belykh, Y. Gong, M.C. Preul, B. Hannaford, E.J. Seibel, Toward real-time tumor margin identification in image-guided robotic brain tumor resection, in: Medical Imaging 2017: Image-Guided Procedures, Robotic Interventions, and Modeling, Vol. 10135, SPIE, 2017, pp. 105–114.
[27] M. Islam, D.A. Atputharuban, R. Ramesh, H. Ren, Real-time instrument segmentation in robotic surgery using auxiliary supervised deep adversarial learning, IEEE Robot. Automat. Lett. 4 (2) (2019) 2188–2195.
[28] Z. Xie, D. Gillies, Near real-time hippocampus segmentation using patch-based canonical neural network, arXiv preprint arXiv:1807.05482.
[29] D. Jha, N.K. Tomar, S. Ali, M.A. Riegler, H.D. Johansen, D. Johansen, T. de Lange, P. Halvorsen, Nanonet: Real-time polyp segmentation in video capsule endoscopy and colonoscopy, 2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS), IEEE 2021 (2021) 37–43.
[30] E.M.A. Anas, P. Mousavi, P. Abolmaesumi, A deep learning approach for real time prostate segmentation in freehand ultrasound guided biopsy, Med. Image Anal. 48 (2018) 107–116.
[31] U. Jamil, A. Sajid, M. Hussain, O. Aldabbas, A. Alam, M.U. Shafiq, Melanoma segmentation using bio-medical image analysis for smarter mobile healthcare, J. Ambient Intell. Humaniz. Comput. 10 (10) (2019) 4099–4120.
[32] X. Xu, Q. Lu, T. Wang, J. Liu, C. Zhuo, X.S. Hu, Y. Shi, Edge segmentation: Empowering mobile telemedicine with compressed cellular neural networks, in: 2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD) IEEE, 2017, pp. 880–887.
[33] Z.-L. Ni, G.-B. Bian, Z.-G. Hou, X.-H. Zhou, X.-L. Xie, Z. Li, Attention-guided lightweight network for real-time segmentation of robotic surgical instruments, 2020 IEEE international conference on robotics and automation (ICRA), IEEE 2020 (2020) 9939–9945.
[34] Q. Zhou, Q. Wang, Y. Bao, L. Kong, X. Jin, W. Ou, Laednet: A lightweight attention encoder–decoder network for ultrasound medical image segmentation, Comput. Electr. Eng. 99 (2022) 107777.
[35] M.A. Bragin, P.B. Luh, J.H. Yan, N. Yu, G.A. Stern, Convergence of the surrogate lagrangian relaxation method, J. Optimiz. Theory Appl. 164 (1) (2015) 173–201.
[36] D. Gurevin, S. Zhou, L. Pepin, B. Li, M. Bragin, C. Ding, F. Miao, Enabling retrain-free deep neural network pruning using surrogate lagrangian relaxation, arXiv preprint arXiv:2012.10079.
[37] H. Chen, X. Qi, L. Yu, P.-A. Heng, Dcan: deep contour-aware networks for accurate gland segmentation, in: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2016, pp. 2487–2496.
[38] L. Yang, Y. Zhang, J. Chen, S. Zhang, D.Z. Chen, Suggestive annotation: A deep active learning framework for biomedical image segmentation, in: International conference on medical image computing and computer-assisted intervention, Springer, 2017, pp. 399–407.
[39] S. Graham, H. Chen, J. Gamper, Q. Dou, P.-A. Heng, D. Snead, Y.W. Tsang, N. Rajpoot, Mild-net: Minimal information loss dilated network for gland instance segmentation in colon histology images, Med. Image Anal. 52 (2019) 199–211.
[40] D. Jin, D. Guo, T.-Y. Ho, A.P. Harrison, J. Xiao, C.-K. Tseng, L. Lu, Deep esophageal clinical target volume delineation using encoded 3d spatial context of tumors, lymph nodes and organs at risk, in: International Conference on Medical Image Computing and Computer-Assisted Intervention Springer, 2019, pp. 603–612.
[41] S. Li, J. Zhang, C. Ruan, Y. Zhang, Multi-stage attention-unet for wireless capsule endoscopy image bleeding area segmentation, in: 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) IEEE, 2019, pp. 818–825.
[42] E. Goceri, Diagnosis of alzheimer's disease with sobolev gradient-based optimization and 3d convolutional neural network, Int. J. Numer. Methods Biomed. Eng. 35 (7) (2019) e3225.
[43] J. Zhang, Y. Zhang, X. Xu, Pyramid u-net for retinal vessel segmentation, arXiv preprint arXiv:2104.02333.
[44] Z. Yan, X. Yang, K.T. Cheng, A three-stage deep learning model for accurate retinal vessel segmentation, IEEE J. Biomed. Health Inform. 23 (4) (2019) 1427–1436.
[45] C. Chen, C. Biffi, G. Tarroni, S. Petersen, W. Bai, D. Rueckert, Learning shape priors for robust cardiac mr segmentation from multi-view images, in: International Conference on Medical Image Computing and Computer-Assisted Intervention Springer, 2019, pp. 523–531.
[46] A. Mortazi, R. Karim, K. Rhode, J. Burt, U. Bagci, Cardiacnet: Segmentation of left atrium and proximal pulmonary veins from mri using multi-view cnn, in:

International Conference on Medical Image Computing and Computer-Assisted Intervention Springer, pp. 377–385.
[47] Y.-X. Zhao, Y.-M. Zhang, M. Song, C.-L. Liu, Multi-view semi-supervised 3d whole brain segmentation with a self-ensemble network, in: International Conference on Medical Image Computing and Computer-Assisted Intervention Springer, 2019, pp. 256–265.
[48] S. Wang, M. Zhou, O. Gevaert, Z. Tang, D. Dong, Z. Liu, T. Jie, A multi-view deep convolutional neural networks for lung nodule segmentation, 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), IEEE 2017 (2017) 1752–1755.
[49] A. Birenbaum, H. Greenspan, Multi-view longitudinal cnn for multiple sclerosis lesion segmentation, Eng. Appl. Artif. Intell. 65 (2017) 111–118.
[50] V.A. Zimmer, A. Gomez, E. Skelton, N. Toussaint, T. Zhang, B. Khanal, R. Wright, Y. Noh, A. Ho, J. Matthew, et al., Towards whole placenta segmentation at late gestation using multi-view ultrasound images, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer, 2019, pp. 628–636.
[51] S. Liang, K.-H. Thung, D. Nie, Y. Zhang, D. Shen, Multi-view spatial aggregation framework for joint localization and segmentation of organs at risk in head and neck ct images, IEEE Trans. Med. Imaging 39 (9) (2020) 2794–2805.
[52] P. Kennel, L. Teyssedre, J. Colombelli, F. Plouraboué, Toward quantitative three-dimensional microvascular networks segmentation with multiview light-sheet fluorescence microscopy, J. Biomed. Optics 23 (8) (2018) 086002.
[53] A. Dai, M. Nießner, 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 452–468.
[54] A. Dai, A.X. Chang, M. Savva, M. Halber, T. Funkhouser, M. Nießner, Scannet: Richly-annotated 3d reconstructions of indoor scenes, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 5828–5839.
[55] S. Han, J. Pool, J. Tran, W. Dally, Learning both weights and connections for efficient neural network, Adv. Neural Inform. Process. Syst. 28 (2015) 1135–1143.
[56] J.-H. Luo, J. Wu, W. Lin, Thinet: A filter level pruning method for deep neural network compression, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 5058–5066.
[57] Y. Wang, C. Wang, Z. Wang, et al., Mcmia: Model compression against membership inference attack in deep neural networks, arXiv preprint arXiv:2008.13578.
[58] T. Zhang, X. Ma, Z. Zhan, et al., A unified dnn weight compression framework using reweighted optimization methods, arXiv preprint arXiv:2004.05531.
[59] T.-J. Yang, Y.-H. Chen, V. Sze, Designing energy-efficient convolutional neural networks using energy-aware pruning, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5687–5695.
[60] T. Zhang, S. Ye, K. Zhang, X. Ma, N. Liu, L. Zhang, J. Tang, K. Ma, X. Lin, M. Fardad, et al., Structadmm: A systematic, high-efficiency framework of structured weight pruning for dnns, arXiv preprint arXiv:1807.11091.
[61] X. Dai, H. Yin, N.K. Jha, Nest: A neural network synthesis tool based on a grow-and-prune paradigm, IEEE Trans. Comput. 68 (10) (2019) 1487–1497.
[62] T. Zhang, S. Ye, K. Zhang, J. Tang, W. Wen, M. Fardad, Y. Wang, A systematic dnn weight pruning framework using alternating direction method of multipliers, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 184–199.
[63] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778.
[64] N. Usuyama, K. Chahal, Unet/fcn pytorch,https://github.com/usuyama/pytorch-unet, 2020.
[65] E. Goceri, Diagnosis of skin diseases in the era of deep learning and mobile technology, Comput. Biol. Med. 134 (2021) 104458.
[66] E. GÖÇERİ, An application for automated diagnosis of facial dermatological diseases, İzmir Katip Çelebi Üniversitesi Sağlık Bilimleri Fakültesi Dergisi 6 (3) (2021) 91–99.
[67] T.A. Sorensen, 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 (1948) 1–34.
[68] L.R. Dice, Measures of the amount of ecologic association between species, Ecology 26 (3) (1945) 297–302.
[69] S. Jadon, A survey of loss functions for semantic segmentation, in: 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), IEEE, 2020, pp. 1–7.
[70] S. Boyd, N. Parikh, E. Chu, Distributed optimization and statistical learning via the alternating direction method of multipliers, Now Publishers Inc, 2011.
[71] L. Bottou, Large-scale machine learning with stochastic gradient descent, in: Proceedings of COMPSTAT'2010, Springer, 2010, pp. 177–186.
[72] X. Xu, T. Wang, Y. Shi, H. Yuan, Q. Jia, M. Huang, J. Zhuang, Whole heart and great vessel segmentation in congenital heart disease using deep neural networks and graph matching, in: International Conference on Medical Image Computing and Computer-Assisted Intervention Springer, 2019, pp. 477–485.
[73] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980.
[74] I. Loshchilov, F. Hutter, Sgdr: Stochastic gradient descent with warm restarts, arXiv preprint arXiv:1608.03983.
[75] N. Developer, Jetson tx2 module,https://developer.nvidia.com/embedded/jetson-tx2, 2020.
[76] C. Poynton, Digital video and HD: Algorithms and Interfaces, Elsevier, 2012.
[77] E. Goceri, Capsnet topology to classify tumours from brain images and comparative evaluation, IET Image Proc. 14 (5) (2020) 882–889.

Shanglin Zhou received the B.S. degree in statistics from Minzu University of China, Beijing, China, in 2015, and the M.S. degree in statistics from the University of Connecticut, Storrs, CT, USA, in 2017. She is currently working toward the Ph.D. degree in computer science and engineering at the University of Connecticut, Storrs, CT, USA. Her current research interests include deep learning model compression and the application of computer vision.

Xiaowei Xu is currently an associate professor at Guangdong Cardiovascular Institute, Guangdong Provincial People's Hospital, Guangzhou, China. He received the BS and PhD degrees in electronic science and technology from Huazhong University of Science and Technology, Wuhan, China, in 2011 and 2016 respectively. He worked as a post-doc researcher at University of Notre Dame, IN, USA from 2016 to 2019. His research interests include deep learning, and medical image segmentation. He was a recipient of DAC system design contest special service recognition reward in 2018 and outstanding contribution in reviewing, Integration, the VLSI journal in 2017. He has served as TPC members in ICCD, ICCAD, ISVLSI and ISQED.

Jun Bai is a 3rd year PhD student in Dr. Sheida Nabavi's lab at University of Connecticut. With a multidisciplinary background in computer science and medical imaging, she has been actively researching and developing deep learning models for medical image. She is author of six peer-reviewed publications.

Mikhail A. Bragin (Senior Member: IEEE; Member: INFORMS, IISE, PES) is an Assistant Research Professor with the Department of Electrical and Computer Engineering, University of Connecticut. His research work has been supported by the U.S. National Science Foundation, BNL, MISO, ISO-NE, ABB, CESMII, and AFRL. His research is geared toward solving complex technical and societal challenges within smart grids, manufacturing, and healthcare. Accordingly, his research interests include operations research, mathematical optimization, artificial intelligence, machine learning, quantum computing with applications to power systems optimization, grid integration of renewables, energy-based operation optimization of distributed energy systems, stochastic scheduling within manufacturing systems, and pharmaceutical scheduling. His research has appeared in top journals such as Journal of Optimization Theory and Applications, IEEE Transactions on Power Systems, IEEE Transactions on Automation Science and Engineering, Decision Support Systems, and IEEE Robotics and Automation Letters as well as in top conferences such as the PES General Meeting, the INFORMS Annual Meeting, IISE Annual Conference and Expo, and the International Joint Conference on Artificial Intelligence.