Received: 4 April 2023 | Revised: 2 June 2023 | Accepted: 5 July 2023

CAAI Transactions on Intelligence Technology

DOI: 10.1049/cit2.12303

ORIGINAL RESEARCH

Improved organs at risk segmentation based on modified U-Net with self-attention and consistency regularisation

Maksym Manko[1,2] | Anton Popov[1] | Juan Manuel Gorriz[2] | Javier Ramirez[2]

[1]Electronic Engineering Department, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine

[2]Department of Signal Theory, Telematics and Communications, University of Granada, Granada, Spain

Correspondence

Maksym Manko.
Email: mman@correo.ugr.es

Abstract

Cancer is one of the leading causes of death in the world, with radiotherapy as one of the treatment options. Radiotherapy planning starts with delineating the affected area from healthy organs, called organs at risk (OAR). A new approach to automatic OAR segmentation in the chest cavity in Computed Tomography (CT) images is presented. The proposed approach is based on the modified U-Net architecture with the ResNet-34 encoder, which is the baseline adopted in this work. The new two-branch CS-SA U-Net architecture is proposed, which consists of two parallel U-Net models in which self-attention blocks with cosine similarity as query-key similarity function (CS-SA) blocks are inserted between the encoder and decoder, which enabled the use of consistency regularisation. The proposed solution demonstrates state-of-the-art performance for the problem of OAR segmentation in CT images on the publicly available SegTHOR benchmark dataset in terms of a Dice coefficient (oesophagus—0.8714, heart—0.9516, trachea—0.9286, aorta—0.9510) and Hausdorff distance (oesophagus—0.2541, heart—0.1514, trachea—0.1722, aorta—0.1114) and significantly outperforms the baseline. The current approach is demonstrated to be viable for improving the quality of OAR segmentation for radiotherapy planning.

KEYWORDS

3-D, COMPUTER VISION, DEEP LEARNING, DEEP NEURAL NETWORKS, IMAGE SEGMENTATION, MEDICAL IMAGE PROCESSING, OBJECT SEGMENTATION

1 | INTRODUCTION

Cancer ranks as one of the leading causes of death and an important barrier to increasing life expectancy in every country of the world. According to estimates from the World Health Organisation (WHO) in 2019, cancer is the first or second leading cause of death before the age of 70 years in 112 of 183 countries and ranks third or fourth in a further 23 countries. Lung cancer remains the leading cause of cancer death, with an estimated 1.8 million deaths in 2020 (18%), followed by colorectal (9.4%), liver (8.3%), stomach (7.7%), and female breast (6.9%) cancers. [1].

Radiation therapy is one of the standard treatments for lung and oesophageal cancer. In this procedure, the tumour is irradiated with ionising beams to destroy the target tumour, while protecting healthy tissue and surrounding organs, called organs at risk (OAR), from radiation. Thus, defining the boundaries of the target tumour and OAR in computed tomography (CT) images is the first step in treatment planning. As a rule, the segmentation of OAR is performed manually by an expert. Manual segmentation is time-consuming and can be a source of human error. In this regard, it becomes necessary to develop automatic methods of OAR segmentation, which would speed up the segmentation process and improve the quality of delineation of healthy organs from the affected area [2].

Over the past few years, many solutions based on convolutional neural networks have been proposed to the problem of OAR segmentation in the thorax [3–14] (see Related work section), and deep learning is proven to be the most promising approach to solving medical image segmentation problems.

One of the common difficulties in working with the databases of medical imaging dataset is the small size of the available training data, which does not provide enough variability of the anatomical structures that can be observed in medical practice. This is particularly the case for SegTHOR [2] dataset of 60 CT images introduced on a Codalab platform for comparing the solutions for the OAR segmentation in the chest cavity and since then recognised as the benchmark dataset for such a task. Such a small amount of the training data may lead to the fact that deep networks start to learn non-informative local features of training samples, which results in worse performance at the inference stage (overfitting). Working with comparatively small medical datasets requires specific actions to prevent overfitting and reduce the variance of the deep learning models [15–17].

To prevent overfitting, regularisation techniques are used, including weight decay [18, 19], batch normalisation [20], dropout [21], augmentations [22, 23], mixing input data and feature maps [24–27], additional noise [28–30], consistency regularisation [31–38], contrastive regularisation [39, 40] etc.

Consistency regularisation is a highly effective method of model regularisation that facilitates the training of more generalised models and prevents the learning of local non-informative features of training samples, which can lead to overfitting. This regularisation technique can be particularly useful when training models on small datasets, such as SegTHOR. The original version of consistency regularisation involves utilising unlabelled data by assuming that the model should generate similar predictions when presented with perturbed versions of the same image. [31, 34, 35, 37] Recently, new variations of consistency regularisation have been introduced, wherein the same data is fed to two different neural networks instead of feeding perturbed input data to one network. We have adopted this approach in our research work.

Typically, consistency regularisation is used during unsupervised training (for example, generative adversarial networks (GANs) [34, 35]) or semi-supervised training [36–38] to unlabelled data, including segmentation tasks [31–33]. While some works [41, 42] have used consistency learning in supervised tasks; these are multi-task learning problems where different neural network branches solve different tasks, and consistency loss functions are calculated for the intermediate representations of input data. In our research, we propose to explore the possibility of applying consistency regularisation to labelled data without the use of auxiliary unlabelled data. To this end, two neural networks (or two branches of a neural network) will solve the same problem, and consistency regularisation will be applied to the output feature maps.

To further improve model performance, we have also implemented attention mechanisms, which have been shown to increase the efficiency of neural networks [43, 44]. In recent years, attention mechanisms in neural networks have gained popularity and have become a crucial component of many state-of-the-art models. Attention mechanisms have been particularly important for the success of transformer-based architectures, such as BERT [45] and GPT-3 [46], which have outperformed recurrent neural networks for sequential modelling tasks. Moreover, attention-based models have also demonstrated competitive performance with fully convolutional networks (FCNs) [47–49] in the field of computer vision and have achieved state-of-the-art results on various tasks [50–52]. Self-attention is one such mechanism, which relates different positions of a single sequence to compute a representation of the sequence [43]. However, we found that the use of common self-attention blocks resulted in uninformative feature maps for SegTHOR data. To address this issue, we proposed a modified attention block where the inner product between pixel representations was replaced by scalable cosine similarity. The use of this modified attention block has shown a notable improvement in comparison to the commonly used self-attention block. This is due to the fact that the common block is highly sensitive to the distribution of feature values and can result in one pixel having significantly greater attention weight than others in the attention map.

This paper introduces a novel two-branch CS-SA U-Net architecture, which includes two parallel U-Net-like models with self-attention blocks. The inner products of the element representations in the self-attention blocks are replaced by cosine similarities (CS-SA block). The proposed model is trained using consistency regularisation, which optimises the consistency loss between the two branches of the neural network using hard pseudo-labels. Our approach to OAR segmentation outperforms the defined baseline and achieves a state-of-the-art performance in the trachea and aorta segmentation while demonstrating competitive performance for oesophagus and heart segmentation.

2 | RELATED WORK

In this section, several approaches to the design of segmentation systems based on deep neural networks for organ segmentation in CT images are presented.

Most of the existing solutions to the problem of OAR segmentation [3–14] are based on U-Net- [47] or V-Net-based [48] architectures. U-Net and V-Net architectures are similar conceptually and differ mainly in dimensionality (2D and 3D) and minor architectural details. Therefore, by U-Net- and V-Net-based architectures, we mean an encoder–decoder structure with a residual connection between the encoder and decoder blocks.

Han et al. [3] proposed a V-Net model with bottleneck blocks instead of the conventional convolutional layers. A feature of their approach is the use of a multi-resolution strategy, where one model predicts coarse organ masks, which are then used to localise regions of interest for a high-resolution model. The similar multi-resolution approaches were developed by Chen et al. [4] and Zhang et al. [5]. Zhang et al. use a common encoder for coarse and fine resolution and separate decoders. Kim et al. [6] also propose a two-stage approach with regions of interest extraction, but in 2D fashion, a fine-resolution segmentation model predicts masks along all three axes. Wang et al. [7] proposed a cascaded end-to-end multi-scale model.

Several papers have successfully applied dilated convolutional layers [8, 9]. Lachinov [10] applied pixel shuffling as a layer up sampling in 3D U-Net with ResNet Encoder. Feng et al. [11] proposed an axis-based denoise method as a post-processing technique for mask adjustment. He et al. [12] proposed a 2D U-Net with an auxiliary head, which was used to binary classify the presence of each organ in a CT slice. In another work [13], they proposed a similar system with an additional false positive filtering with a dynamic threshold selection.

Particularly noteworthy is the Isensee et al. work [14], which presents the current state-of-the-art approach to OAR segmentation and other medical image segmentation tasks. In this work, a framework called nnU-Net ('no-new-Net') is presented, which implements three variations of the U-Net architecture (2D, 3D, and 3D cascaded). The main feature of this framework is the complete automation of the training pipeline design process: architecture parameters, pre-processing, and hyperparameters of the training process completely depend on the features of the dataset and are determined automatically, demonstrating high performance on a huge set of semantic segmentation problems.

Many research studies focus extensively on system architectures, yet comparatively little attention is devoted to regularisation techniques. When dealing with small datasets, the regularisation of models is a crucial question. Moreover, attention mechanisms, which can be highly beneficial, especially in tasks that involve the segmentation of small objects, are often overlooked in these studies.

3 | METHOD

In this section, we present a novel approach for OAR segmentation using a U-Net-like architecture that incorporates a newly proposed CS-SA block and consistency regularisation. The approach is described in detail below.

3.1 | Baseline model

Medical image segmentation is a crucial task in computer-aided diagnosis and treatment planning. Various segmentation neural network architectures have been developed, including 2D, 2.5D, 3D, or combined. In 2D convolutional networks, slices of CT images are fed into the input, and segmentation masks are obtained for each corresponding slice. In 3D convolutional networks, a whole CT image or a cropped CT image is fed into the input, and a mask is returned with the same size as the input image. The 2.5D approach provides a trade-off between the 2D and 3D models, where a three-dimensional crop of the image, consisting of several slices of a CT image, is fed into the input, and a mask is returned only for the middle slice.

Given the potential hardware and memory limitations, as well as the relatively small size of our dataset, we opted to utilise a 2D neural network approach that performs organ segmentation slice by slice.

U-Net architecture was chosen as a baseline model which is one of the basic and the most common models for semantic segmentation problems. The U-Net architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localisation. To avoid loss of context about small objects on large receptive fields, there are added skip connections between the encoder and decoder blocks. To enhance the performance of the baseline model, we replaced the original U-Net encoder with the ResNet-34 architecture [53], which is a deeper and more complex network that can handle more complicated image features. In addition, ResNet-34 was trained and validated on the large-scale classification dataset ImageNet, and the parameters of the model pretrained on this dataset are publicly available. In this regard, the encoder of the model was initialised with pretrained ImageNet parameters.

In the proposed approach, a combination of categorical cross-entropy and logarithmic Dice loss [54] is used as an optimised loss function $L(p,y)$:

$$L(p,y) = \frac{\mathrm{DICE}(p,y) + \mathrm{CE}(p,y)}{2}, \tag{1}$$

$$\mathrm{CE}(p,y) = -\frac{1}{\mathrm{N}} \sum_{i=1}^{N} \sum_{c=1}^{C} 1_{y_i \in C_c} \log\left(p\left[y_i \in C_c\right]\right), \tag{2}$$

$$\mathrm{DICE}(p,y) = -\log\left(2 \cdot \frac{\sum_{i=1}^{N} \sum_{c=1}^{C} y_{ci} \cdot p_{ci}}{\sum_{i=1}^{N} \sum_{c=1}^{C} y_{ci} + p_{ci}}\right), \tag{3}$$

where y—ground truth; p—softmax probability; N—number of samples (pixels); C–number of classes; and $1_{y_i \in C_c}$—indicator function.

3.2 | Self-attention block

The self-attention mechanism has emerged as a powerful architectural solution that can effectively capture global image information at different levels of input data representation by evaluating the interdependence of pixels. In this way, self-attention blocks can complement convolutional layers that extract local image information at various scales of input data representation. The utilisation of global image information can have a significant impact on the accuracy of organs-at-risk segmentation, especially in the thoracic cavity where the object distribution is homogeneous. The attention mechanism can help to more accurately locate and segment organs-at-risk by providing global information on the distribution of objects in the thoracic cavity. This global information can be effectively incorporated into the self-attention blocks, allowing the neural network to take into account both global and local features for better segmentation results.

In the proposed implementation of the CS-SA block, the traditional dot product operation of queries with keys is substituted by calculating the cosine similarity, followed by

multiplying the resulting values with a learnable parameter denoted as β. A similar method was previously utilised in Henry et al. [55], but in the context of machine translation. To initialise the β parameter, we apply a heuristic expression

$$\beta = \log_2\left(L^2 - L\right), \tag{4}$$

where L—a sequence length.

Figure 1 illustrates the self-attention block used in the proposed architecture. The input tensor x is processed by three independent convolutional layers with kernel 1 × 1 and outputs of these convolutional layers are flattened in space dimensions. Three representations of x referred to as queries, keys, and values ($f(x)$,$g(x)$,$v(x)$, respectively) are obtained. An attention function maps a query and a set of key-value pairs to an output. The output is obtained by computing a weighted sum of the values, where each weight corresponds to a compatibility function of the query with the corresponding key [43]. The calculated output is reshaped back to the input spatial resolution and fed to the convolution layer with kernel 1 × 1. In some self-attention blocks of the CS-SA U-Net architecture, the input tensor size is very large. To reduce memory usage, a max pooling operation is applied to the matrices of keys and values. The output of the CS-SA block is multiplied by the self-attention weight parameter α and added to the input tensor of the block.

The output of the proposed attention block can be calculated as follows:

$$\mathrm{attn}_{ij} = \mathrm{softmax}\left(\beta \cdot \frac{f(x)_i \cdot g(x)_j^T}{\left\|f(x)_i\right\| \cdot \left\|g(x)_j^T\right\|}\right), \tag{5}$$

$$o = \mathrm{attn} \cdot v(x), \tag{6}$$

$$y = x + \alpha \cdot \mathrm{conv}(o_r), \tag{7}$$

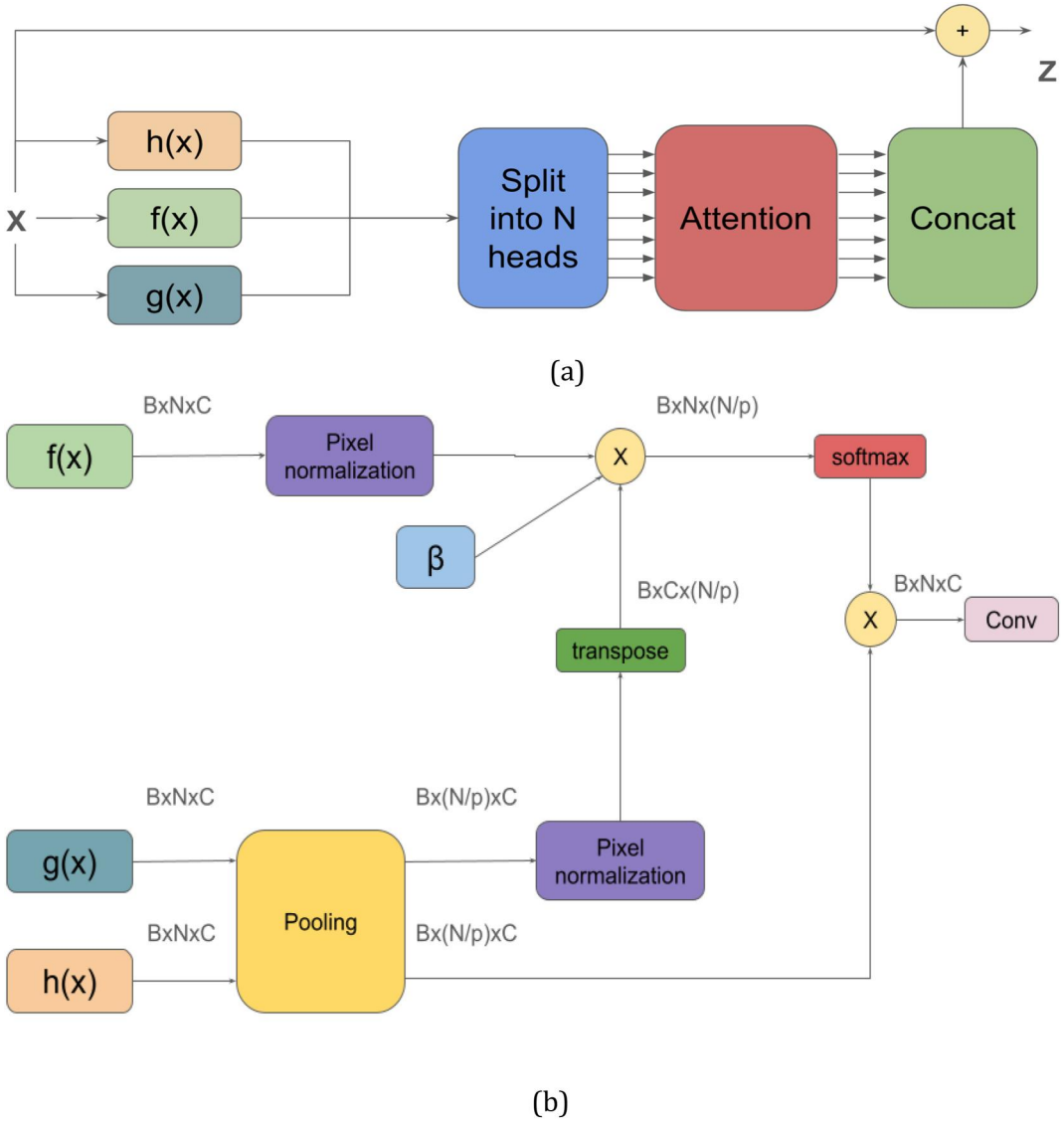

FIGURE 1 Self-attention block: (a) the proposed multi-head self-attention block; (b) the structure of the self-attention block.

where $f(x)$,$g(x)$,$v(x)$–a query, a key and a value tensors accordingly; $||\cdot||$—vector's L2-norm; softmax(·)—a softmax function applied over the last dimension; β—a scaling parameter of the attention block; α—a weight parameter of the attention block output; o_r—reshaped output tensor o; and conv(·) - a convolutional layer.

The attention blocks in the proposed architecture are implemented as multi-headed attention. In the multi-head attention block, the input tensor, or query, key and value tensors as in our case are split into multiple tensors along the channel dimension. Thus, N sets of smaller-dimensional queries, keys and values are obtained and processed in parallel, each in its own head, and the outputs from all heads are concatenated.

In the CS-SA U-Net architecture (Figure 2), CS-SA blocks are incorporated between encoder blocks and decoder blocks of the baseline U-Net model with the ResNet-34 encoder. The input tensor resolution for the first block in the encoder and the last block in the decoder is too large to fit into the memory, so a max pooling layer with a filter size and step of two is utilised in these blocks. For the remaining layers, the filter size and step are set to 1. The α parameter is fixed to 1 and is not learnable, while the head size is set to 64 filters for all self-attention blocks.

With the exception of the incorporated CS-SA blocks, the architecture remains similar to the standard U-Net architecture with the ResNet-34 encoder. The encoder comprises five encoder blocks, and each has a different receptive field sizes. Conventional ResNet-34 architecture with cut out classification head is used as an encoder. The decoder follows the conventional U-Net design, consisting of four decoder blocks that incrementally expand the receptive field with each successive block. Hidden feature maps obtained from the encoder block outputs are propagated through short connections, where they are concatenated with the input feature maps of the decoder blocks.

3.3 | Consistency regularisation

Consistency regularisation provides more a efficient implementation of collaborative training of two neural networks, in our case, training of two branches of a neural network. The use of consistency regularisation introduces noise into the data

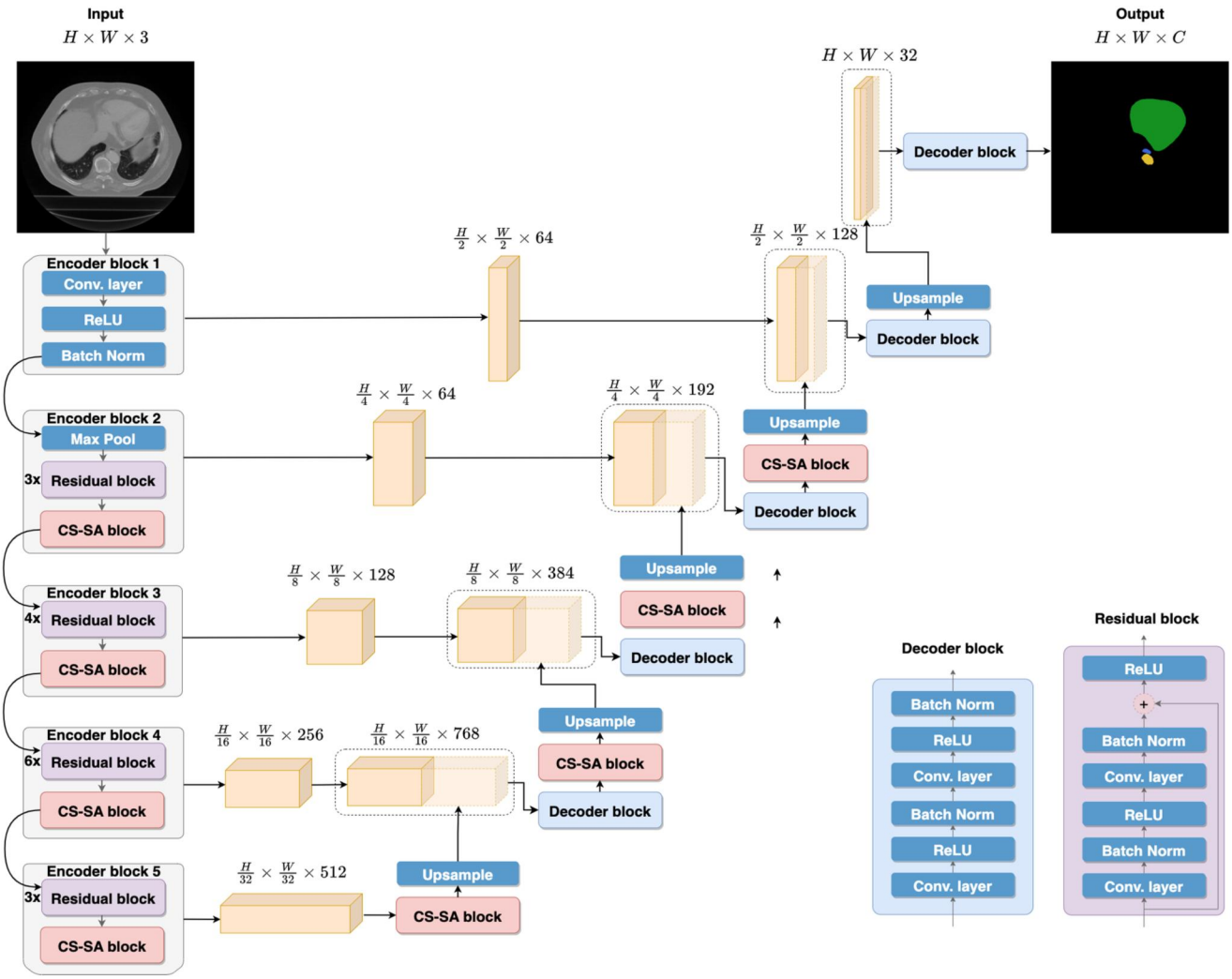

FIGURE 2 CS-SA U-Net architecture.

labels during the training stage, thereby regularising the model and reducing the overfitting effect.

In this work we propose to use a Cross Pseudo Supervision approach [33] as the strategy for consistency learning. Training pipeline with consistency regularisation consists of two parallel segmentation models with same architectures $f(\theta_1)$ and $f(\theta_2)$ initialised differently. The input batch for both networks is the same, and it means that the input data and augmentation of this data at each step are the same for both networks. At each step, both models return confidence maps P_1 and P_2, which are then one-hot encoded into Y_1 and Y_2. One-hot encoded predictions are then used as pseudo labels while calculating a consistency loss function $L_{\text{cons}}(p_1,p_2,y_1,y_2)$:

$$L_{\text{cons}}(p_1,p_2,y_1,y_2) = \frac{\text{CE}(p_1,y_2) + \text{CE}(p_2,y_1)}{2}, \quad (8)$$

where p_1,p_2—softmax probabilities of the two models, respectively; y_1,y_2—one-hot encoded prediction of the two models, respectively.

The supervised loss $L_{\text{sv}}(p_1,p_2,y)$ function is also calculated at each step for both models:

$$L_{\text{sv}}(p_1,p_2,y) = \frac{\text{DICE}(p_1,y) + \text{CE}(p_1,y) + \text{DICE}(p_2,y) + \text{CE}(p_2,y)}{2}. \quad (9)$$

The final loss function $L(p_1,p_2,y_1,y_2,y)$ is calculated by the following expression:

$$L(p_1,p_2,y_1,y_2,y) = L_{\text{sv}}(p_1,p_2,y) + \lambda \cdot L_{\text{cons}}(p_1,p_2,y_1,y_2), \quad (10)$$

where λ—a consistency learning a trade-off coefficient.

A schematic block diagram of the pipeline with consistency regularisation is shown in Figure 3. The input tensor X and its corresponding masks Y undergo preprocessing and transformation prior to being utilised. Subsequently, the transformed input tensor is provided as input to both models. An argmax operation is applied to the resulting output tensors P_1 and P_2, yielding hard pseudo labels Y_1 and Y_2. The models' outputs P_1 and P_2 are utilised to compute the loss function using the actual labels Y. Additionally, a consistency loss function is computed by leveraging the pseudo labels Y_1 and Y_2.

4 | EXPERIMENTS

4.1 | Dataset

In our experiments, the SegTHOR benchmark dataset [2] is used, which contains 60 thoracic CT scans with manual labelled oesophagus, heart, trachea, and aorta (Figure 4). CT scans in this dataset have resolutions 0.98 × 0.98 × (2–2.5) mm per voxel. The data and corresponding masks are divided into 40 images for training and 20 images for testing.

4.2 | Pre-processing

At the pre-processing step, voxel values are clipped between −1000 and 1000 Hounsfield units (HU). HU values lower than −1000 do not have any semantic meaning (−1000 HU corresponds to air) and are used for padding. Values higher than 1000 do not bring any relevant information since they represent bones or foreign bodies. [56] After clipping, the images were transformed from single-channel to three-channel by duplicating the image and normalising with ImageNet [57] statistics. Finally, CT image slices are cropped in the centre from 512 × 512 to 320 × 320.

4.3 | Training setup

During the training phase, five-fold cross-validation with stratification by patients was used for all experiments. Data is

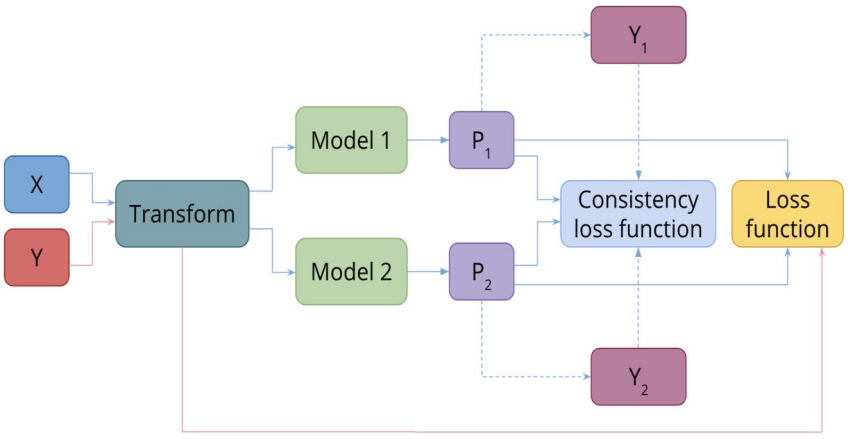

FIGURE 3 Training pipeline with consistency regularisation.

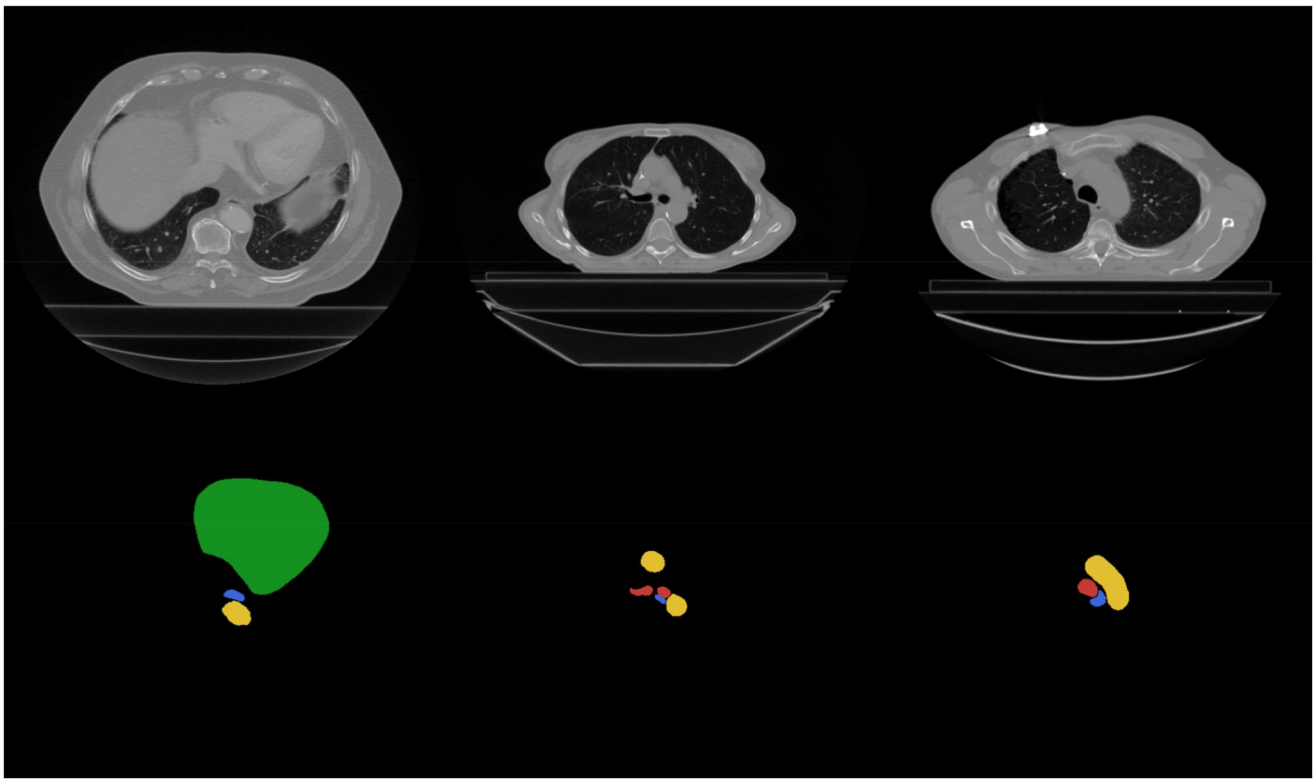

FIGURE 4 Example of the SegTHOR training sample: top—computed tomography slices; bottom—corresponding masks with manually labelled organs (green—heart, blue - oesophagus, yellow—aorta, red—trachea).

randomly sampled as 2D transverse slices of CT images. To prevent overfitting, data is augmented with affine transformations (horizontal flip, rotation, shift, and scaling) and randomly cropped to a 160 × 160 patch. In addition, the consistency regularisation method was used to prevent overfitting.

All models have been trained with a batch size of 32 for 1000 epochs with no early stopping. As an optimiser, we use Lookahead [58] with RAdam [59] with an initial learning rate equal to 5e-3 and cosine annealing learning rate scheduler [60] without warmup and restarts. Weight decay was not used. The clipping gradient value is set to 1. During the validation process and the best model selection, the average Dice coefficient (see Evaluation) over all organs was monitored.

The consistency learning trade-off coefficient was tuned by running experiments with five-fold cross-validation. The consistency learning trade-off coefficient from the experiment with the highest average Dice coefficient (see Evaluation) over all organs was selected (Figure 5) and equal to 5. Since at the beginning of the training process, the model predictions are inaccurate and cannot be used as adequate pseudo-labels, and the consistency loss function is optimised starting from epoch 500.

4.4 | Post-processing

At the validation and inference stages, the model predicts one-hot encoded organ masks that are combined into 3D masks for each patient. Subsequently, a post-processing stage is performed in which connected 3D components are clustered, followed by filtering out components that consist of less than 5% of the entire predicted mask. This step is necessary to remove outliers and ensure that only the relevant components are retained.

4.5 | Evaluation

The resulting three-dimensional predicted masks and annotated masks are used to calculate the quality metrics of the model. Dice coefficient and Hausdorff distance are used as quality metrics. Dice coefficient is calculated by the following expression:

$$\mathrm{DICE}(p,y) = \frac{2 \cdot \sum_{i=1}^{N} \sum_{c=1}^{C} y_{ci} \cdot \hat{y}_{ci}}{\sum_{i=1}^{N} \sum_{c=1}^{C} y_{ci} + \hat{y}_{ci}}, \tag{11}$$

where y—ground truth; $\hat{y}$—one-hot encoded predictions; N - number of samples; C - number of classes.

Hausdorff distance is calculated by the following expression:

$$d_H\left(\hat{Y}, Y\right) = \max\left\{\sup_{\hat{y}\in\hat{Y}} \inf_{y\in Y} d(\hat{y}, y), \sup_{y\in Y} \inf_{\hat{y}\in\hat{Y}} d(\hat{y}, y)\right\}, \tag{12}$$

where $\hat{Y}$—set of predicted voxels of a specific class in image; Y—set of annotated voxels of a specific class in image; sup—the supremum; inf—the infimum; and $d(\hat{y}, y)$—the distance between points $\hat{y}$ and y in a specified metric space.

4.6 | Results

Table 1 presents the evaluation results for the OAR segmentation task on the test set. The predictions of all models obtained through five-fold cross-validation are averaged and post-processed using the method described in the Post-processing subsection. Table 1 shows the results for the baseline U-Net model, the two-branch baseline model, the two-branch CS-SA U-Net model, and the two-branch CS-SA U-Net model with consistency regularisation.

All methods outlined above in Table 1 underwent evaluation using the identical test set provided by the SegTHOR dataset authors. No annotations for the test set are available, and the evaluation of the approach is performed on the

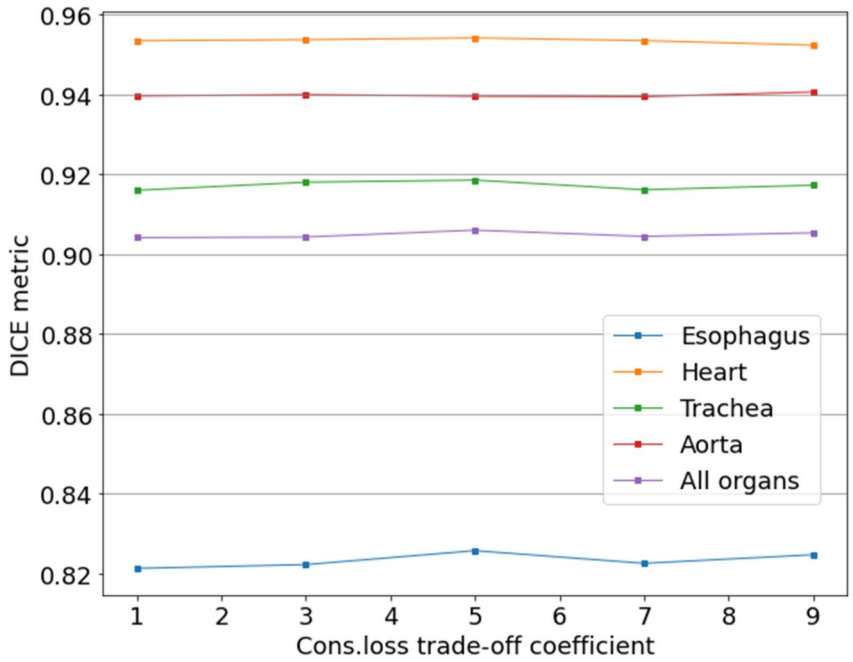

FIGURE 5 Dependence of the validation DICE coefficients on the Cross Pseudo Supervision trade-off coefficient.

TABLE 1 Comparison of evaluation performance on the test set obtained by different methods (red—the top result, blue—the second result).

	DICE				Hausdorff distance			
Experiment	**Eso**	**Heart**	**Trachea**	**Aorta**	**Eso**	**Heart**	**Trachea**	**Aorta**
Chen et al. [4]	0.8166	0.9329	0.8910	0.9232	0.4914	0.2417	0.2746	0.3081
Lachinov [10]	0.8303	0.9381	0.9088	0.9353	-	-	-	-
Vesal et al. [8]	0.8580	0.9410	0.9260	0.9380	0.3310	0.2260	0.1930	0.2970
Wang et al. [7]	0.8597	0.9459	0.9217	0.9433	0.2883	0.1594	0.2045	0.1551
He et al. [13]	0.8594	0.9500	0.9201	0.9484	0.2743	0.1383	0.1824	0.1129
Han et al. [3]	0.8651	0.9536	0.9276	0.9464	0.2590	0.1272	0.1453	0.1209
Isensee et al. [14]	0.8890	0.9570	0.9228	0.9509 (7)	0.1937	0.1216	0.1938	0.1219
Ours								
One-branch baseline	0.8617	0.9524	0.9263	0.9491	0.2635	0.1444	0.1897	0.1236
Two-branch baseline	0.8620	0.9514	0.9258	0.9483	0.2659	0.1509	0.1883	0.1221
Two-branch CS-SA	0.8694	0.9504	0.9283	0.9510 (0)	0.2602	0.1554	0.1812	0.1153
Two-branch CS-SA U-Net + Cons. Reg.	0.8714	0.9516	0.9286	0.9510 (2)	0.2541	0.1514	0.1722	0.1114

SegTHOR contest website. The approaches to organs-at-risk segmentation outlined above diverge from our proposed approach not only in terms of architectural variances but also in preprocessing methods, augmentation techniques, model regularisation methods, optimiser selection, and other factors.

Table 1 primarily showcases a noteworthy enhancement in the performance achieved by the proposed method when compared to the chosen baseline. Additionally, the results of other approaches presented in the table demonstrate the competitiveness of the proposed method in addressing the task of organs-at-risk segmentation.

Figure 6 illustrates an example of the prediction made by the two-branch CS-SA U-Net model on a test sample.

5 | DISCUSSION

This paper proposes a novel architecture, named the two-branch CS-SA U-Net, comprising of two U-Net models with cosine similarity self-attention blocks. The primary distinguishing feature of this block is the use of cosine similarity as a metric for measuring the similarity between pixel representations. Additionally, we hypothesise that consistency learning can serve as an effective regularisation technique for supervised segmentation tasks, and this hypothesis is supported by experimental results.

We demonstrate the efficacy and utility of CS-SA blocks and consistency regularisation in the context of OAR segmentation. Our experimental results show significant improvements in quality metrics compared to the baseline model, and our approach achieves the level of state-of-the-art results.

5.1 | Model dimensionality

The usage of a 2D approach for medical image segmentation can be justified in situations where the dataset size is relatively small or when there are hardware and memory constraints that limit the usage of 3D convolutional neural networks.

When working with small datasets, it may be difficult to obtain sufficient training examples to fully exploit the potential of 3D convolutional neural networks. This may result in overfitting and poor generalisation performance, and hence a 2D approach may be more appropriate. Additionally, some medical imaging applications may only require the segmentation of specific anatomical structures or regions of interest, which may be well represented by 2D slices.

On the other hand, 3D CNNs require more computational resources and memory as compared to 2D CNNs, which can be a limiting factor when working with large datasets or limited hardware. This can lead to longer training times and increased hardware requirements, which may not be feasible in some scenarios. Moreover, complex datasets with irregular structures may pose challenges for 3D CNNs, as they require additional preprocessing steps such as resampling or normalisation to achieve uniform voxel sizes and isotropic resolution.

Therefore, the use of 2D CNNs can be a practical and efficient solution in certain scenarios where hardware and memory limitations or small dataset sizes make the use of 3D CNNs less feasible. However, it is important to note that 2D CNNs may not fully consider the spatial information from 3D volumes and may result in suboptimal performance for certain tasks.

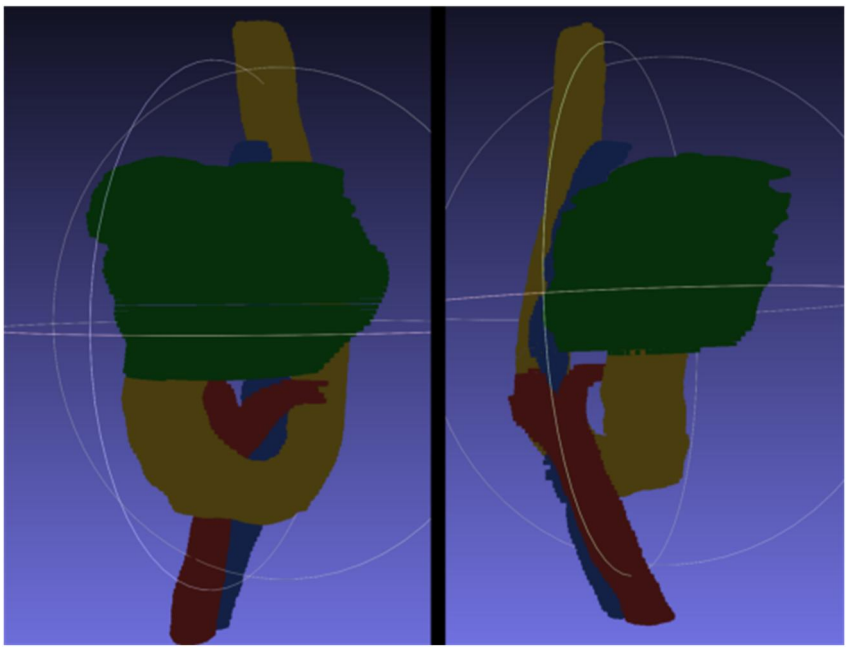

FIGURE 6 Model prediction on a test sample (green—heart, blue—oesophagus, yellow—aorta, red—trachea).

5.2 | Consistency regularisation

In order to determine the most suitable approach for our task, we conducted experiments with several configurations of the consistency learning pipeline. Specifically, we evaluated an approach with soft pseudo-labels and regression loss functions (MSE and KL-divergence), following the Cross Confidence Consistency approach [61], as well as an approach with hard pseudo-labels and a cross-entropy loss function, as in the Cross Pseudo Supervision approach [33]. Given the notable performance demonstrated by these approaches in semi-supervised learning tasks, we adopted them as a framework for our consistency regularisation approach. Furthermore, it is noteworthy that other conventional consistency learning approaches share conceptual similarities with these two approaches. For instance, the Mean Teacher approach [62] and PseudoSeg approach [63] can be regarded as one-sided versions (only one model learning from the pseudo-labels of the other model) of Cross Confidence Consistency and Cross Pseudo Supervision, respectively.

The Mean Teacher approach, characterised by the utilisation of soft pseudo-labels, introduces a lower degree of noise compared to the Cross Pseudo Supervision approach. However, it is likely that this reduced noise level diminishes the regularisation effect of the Mean Teacher approach, leading to a minimal impact on the overall performance. Our results showed that the Cross Pseudo Supervision-based approach had a more significant effect on the learning process and resulted in higher performance. Therefore, we used it as an inspiration for our proposed solution.

During the training process, each branch of the model learns from both the annotator labels and the predictions of the opposite model branch. However, consistency learning can be unstable at an early stage of training, when the model still does not predict accurately enough. To address this issue, we enabled the consistency loss function after a certain number of epochs. We conducted several experiments with different numbers of waiting epochs and found that the most reliable and effective option was to include the consistency loss function when the metric reached a plateau. As a result, we enabled the consistency loss function at the 500-th epoch out of 1000.

It may initially appear counterintuitive that intentionally training a model on partially incorrect pseudo-masks in the presence of correct labels labelled by an expert would not result in a decrease in performance. However, in fact, consistency learning has a regularising effect on the model by introducing noise into the annotations. The use of pseudo-labels also transforms the decision space, making it dynamic as the pseudo-labels change over time due to the training of the model. This process reduces the chance of getting stuck in a local minimum. However, the effectiveness of the consistency loss function cannot be solely attributed to these reasons. In order to better understand the workings of the consistency loss function in this approach, we will consider its behaviour in different scenarios.

- Both branches of the model correctly predict the pixel class—the case where the label and pseudo-label are identical for both branches, and the confidence scores increase; these examples demonstrate a relatively simple pattern that both branches of the model were able to effectively learn from. This suggests the existence of prominent and shared local features that are worth paying attention to;
- Both branches of the model incorrectly predict the pixel class—in cases where the label and pseudo-label do not match for both branches and the confidence scores decrease with a sufficiently large consistency learning trade-off coefficient λ, the examples are considered complex. This complexity may be due to atypical local features for the dataset, erroneous expert labelling, or other reasons. Continuously attempting to train both branches to correctly detect such pixels may result in overfitting of the model. Sacrificing such pixels may potentially lead to an increase in the precision of classifying simpler pixels.
- One branch correctly predicts the pixel class, and the second one does not—this is the most unpredictable case, since this case is essentially an adversarial example—one branch tries to convince the opposite one that this pixel belongs to the annotated class and the second one vice versa.

The relationships between the branches as described above are considered in a one-pixel approximation. Due to the complexity of the model containing millions of parameters, it is challenging to confidently assert the precise nature of the interactions between the branches. However, simplified models of such interactions can be proposed based on assumptions and observations from the experiments conducted.

The consistency regularisation technique not only provided an improvement in the performance of OAR segmentation but it also has the potential to enhance the precision of segmenting objects with poorly defined or blurry boundaries, such as the oesophagus in the SegTHOR dataset.

5.3 | Self-attention

When designing the attention block, various configurations were tested, including multiple heads (multiple heads and single head), the weight coefficient of the attention block output (constant or learnable, different initialisation values), the pixel similarity function (inner product and cosine similarity), the presence of normalisation of attention maps before the softmax function.

Comparing experiments with multi-head and single-head attention blocks, the multi-head approach demonstrated slightly better performance.

Different weight coefficients for the attention maps were also tested, including constant and learnable weights. However, learnable weights tend to approach zero during training, which can lead to the vanishing of the attention block's effect. Additionally, learnable weight coefficients can also have negative values, which is undesirable.

The primary challenge in designing the self-attention block was to ensure an appropriate distribution of softmax values while predicting attention maps. Initially, our model showed nearly the same performance with and without the self-attention blocks described in ref. [64]. Upon closer examination of the internal representations of input data and attention maps, we discovered that the inner product of sequence and key pixel values could generate large values, leading to a wide range of values. This resulted in a distribution of softmax function output values where almost all pixel values tended towards zero, particularly for feature maps with large receptive fields.

As an example, in Figure 7a input image with segmentation mask is presented. For this input image, attention maps were calculated and presented (Figure 7b) for three residual blocks outputs in the encoder. Bottom three images correspond to the attention maps obtained using the cosine similarity function, and top images correspond to the attention maps with the inner product. Every row of the attention map corresponds to attention scores of one query pixel with all key pixels. Each row represents an attention mask for each specific pixel in the input image and can be reshaped to fit the input of the attention block. Figure 7c shows the attention masks for the pixel marked with a pink dot in Figure 7a (top row—vanilla attention blocks, bottom row—CS-SA blocks).

When using the inner product in attention layers, the values of the elements fed to the softmax function can range from hundreds to millions, leading to a few elements in the key matrix with such large feature values regardless of the query element considered, and the inner product is significantly larger for this element of the key matrix than for others. This can result in an attention map with only one or a few columns filled

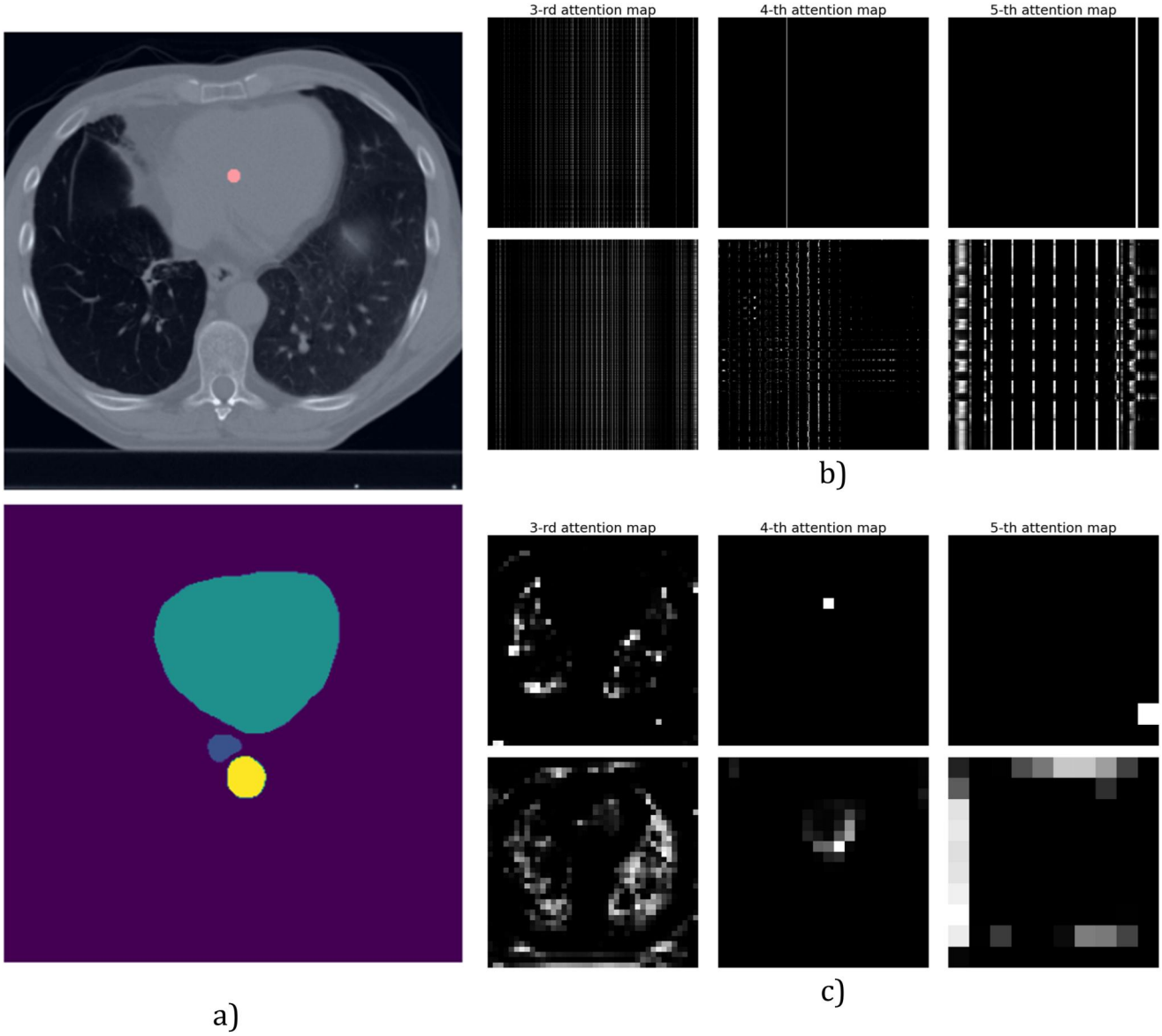

FIGURE 7 Self-attention maps: (a) Input image and mask; (b) Attention maps for one head of intermediate CS-SA blocks (bottom) and vanilla self-attention blocks (top); (c) Attention mask for the pixel marked pink in the input image (bottom—CS-SA blocks, top—vanilla self-attention blocks).

with ones, and the remaining elements are equal to zero. Although this behaviour does not occur in all attention blocks, it can render some of the attention blocks in the architecture useless.

This behaviour of self-attention blocks is observed from the beginning of network training. It is expected that the neural network would learn to normalise the distributions of the inner product values of keys and queries, but there is no guarantee of this. It is likely that the neural network does not fully comprehend the benefits of learning the self-attention block and instead continues to set the softmax values of almost all attention map pixels to zero (Figures 7b,c). However, by replacing the inner product with a cosine similarity function and introducing a trainable scaling parameter beta, some control over the distribution of softmax values is achieved, which encourages the neural network to learn to predict appropriate attention maps (Figure 7b,c).

5.4 | Performance metrics precision and overfitting

In this work, we report segmentation performance using the Dice score metric. The precision of this metric is given to 4 decimal places, which is highly relevant for the three-dimensional object segmentation task. Specifically, when the Dice score reaches values of 0.85–0.9, accurate delineation of the object's boundary becomes more critical, especially for large objects with a significant ratio of volume to surface. For the OAR segmentation task, precise boundary delineation is crucial as it allows more accurate targeting of malignant tissues. We note that the scatter of values among the best solutions in Table 1 is in the range of a third-second decimal digit after the comma, confirming the importance of precision in this task.

To avoid overfitting given the small size of the SegTHOR dataset, we employed data augmentation and cross-validation. We also tested models with an independent control test set using an ensemble of all models obtained by cross-validation. The use of model ensembling ensures good generalisation of the results by averaging predictions and prevents overfitting. For the aorta segmentation, we report results up to 5 decimal places for ranking purposes only, as our results coincide with those of the nnU-Net model up to 4 decimal places.

5.5 | Statistical tests

Although we see an improvement for certain organs in comparison with the baseline according to the average values of the metrics, the question arises about the statistical significance of the obtained estimates. Statistical tests are used to determine the statistical significance of the obtained results. They are used to determine if there is a significant difference between two or more groups or if a relationship exists between two variables.

In this study, we perform pairwise statistical tests for Dice metric scores. As a group, we use a set of Dice scores for each patient obtained in one experiment. Since we do not know the law of distribution of Dice metric scores in each experiment, a non-parametric test will be used. Considering the small sample size and the use of the same test sets in different experiments, the most suitable option is a non-parametric analogue of the paired t-test, which is the Wilcoxon signed-rank test [65].

In our case, a one-tailed test is used, since we are interested in segmentation improvement. Tests are conducted to test two hypotheses that the CS-SA U-Net model with consistency regularisation segments certain organs better than the one-branch baseline model and nnU-Net. p-values are given in Table 2. If we use the typical significance level ($p \leq 0.05$), we can conclude that the CS-SA U-Net model reliably outperforms the baseline model for two organs (oesophagus and aorta) and performs at about the same level for the heart. As for the trachea, although the result does not reach the specified level of significance, we can still assume with a high degree of confidence that the model outperforms the baseline in the trachea segmentation task. As for nnU-Net, nnU-Net strongly outperforms our model in segmenting the oesophagus and heart and segments the aorta at about the same level. For the trachea, the results are similar to those of the baseline model.

5.6 | Limitations

The proposed consistency learning approach in the given section shows potential for applications in various segmentation tasks. However, there are some limitations and considerations that need to be taken into account when utilising this method.

Firstly, the instability of the method is a key limitation. The proposed consistency regularisation approach involves training a model using both real labels and pseudo labels to enforce consistency between predictions of different models. While this can improve generalisation and robustness, it can also introduce instability during training. The selection of the consistency learning trade-off coefficient, which determines the balance between the supervised and consistency losses, requires careful consideration. If this coefficient is set too high, it may lead to overemphasising the consistency loss and result in over-smoothed predictions. On the other hand, setting it too low may undermine the benefits of consistency learning. Finding the right balance is crucial and may require some trial and error or experimentation.

Additionally, the number of waiting epochs before enabling the consistency loss function is another factor that needs careful consideration. Enabling the consistency loss too early in the training process can lead to unstable and unreliable predictions. On the other hand, delaying the introduction of the consistency loss for too long may hinder the model's ability to learn consistent representations. The waiting epochs should be chosen based on the dataset, model architecture, and the convergence characteristics observed during training.

To enhance the proposed consistency learning approach, one possible idea is to investigate adaptive approaches for determining the consistency learning trade-off coefficient. Instead of manually selecting a fixed value, an adaptive

mechanism could dynamically adjust the coefficient based on the progress of training or other relevant metrics. This would allow the model to self-regulate and find the optimal balance between supervised and consistency losses.

Moving on to the self-attention block, it is a versatile component that can be used without any specific restrictions. However, as the size of the input image increases, the computational operations required by the self-attention mechanism also increase significantly. This poses a limitation in terms of computational efficiency, especially when dealing with high-resolution images or large-scale datasets.

To address this limitation, several strategies can be considered. One approach is to reduce the size of the key and value matrices through pooling operations. Pooling can effectively downsample the feature maps and decrease the computational demands of the self-attention mechanism. However, it is important to strike a balance between downsampling and preserving the necessary spatial information for accurate segmentation.

Another strategy is to decrease the number of attention heads in the self-attention block. Attention heads allow the model to capture different types of relationships and dependencies within the input, but reducing their number can help mitigate computational complexity. However, this reduction should be carefully evaluated to ensure that sufficient attention capacity is retained for effective representation learning.

Lastly, reducing the number of channels in the input tensor of the attention block can also help alleviate computational demands. By reducing the dimensionality of the input, fewer computations are required within the self-attention mechanism. However, similar to the previous strategies, a careful analysis of the trade-off between computational efficiency and model performance should be conducted to ensure that the reduction in channels does not adversely affect segmentation accuracy.

In summary, the proposed consistency learning approach has the potential for broader applications but requires caution in selecting appropriate configurations to mitigate instability. For the self-attention block, computational efficiency can be improved by employing strategies such as pooling, reducing attention heads, or decreasing the number of input channels, while considering the impact on segmentation performance.

5.7 | Open questions and further steps

The efficacy of the proposed consistency regularisation method needs further investigation for its application to other tasks.

The impact of including the consistency loss function was observed only in some of the validation folds, while for others, there was no significant positive or negative effect. In the experiment presented in the Results section, the inclusion of the consistency loss function was clearly observed to have a positive impact on three out of five validation folds (Figure 8). However, even when the effect was not clearly observable, the metrics were consistently higher when the model was trained with the regularisation. Further research will be conducted to identify the reasons for the presence or absence of the effect of the consistency loss function.

The effectiveness of the self-attention block has been demonstrated in our experiments, but there are concerns regarding the optimality of the heuristic expression used for calculating the beta parameter. In future research, we plan to derive a more theoretically rigorous formula for initialising the beta parameter specifically for the task of image segmentation.

TABLE 2 *p*-values obtained from the Wilcoxon signed-rank test.

Pair of experiments	Eso	Heart	Trachea	Aorta
One-branch BL - > two branch CS-SA U-Net + Cons. Reg.	0.001	0.608	0.101	0.024
nnU-Net - > two branch CS-SA U-Net + Cons. Reg.	0.999	0.988	0.095	0.622

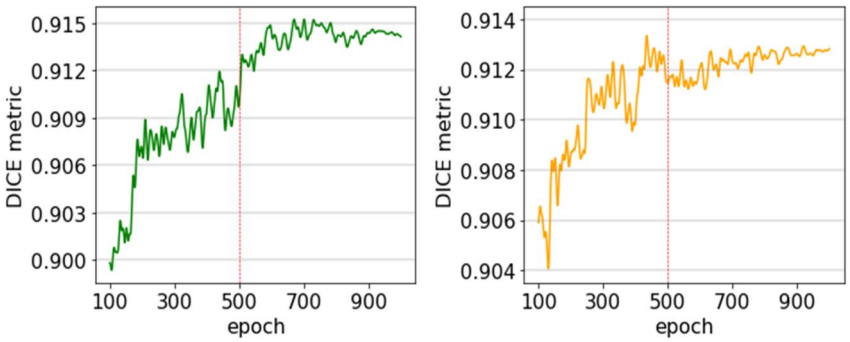

FIGURE 8 Dice metric validation curves for two folds trained with the consistency loss function (for better visibility plots were smoothed with a Gaussian filter and cropped to range from 100-th epoch to 1000-th epoch; the red vertical line indicates the moment of enabling the consistency loss function).

6 | CONCLUSIONS

This paper introduces a novel approach for organ segmentation in the thoracic cavity using a 2D U-Net-like architecture called CS-SA U-Net, which incorporates a custom cosine similarity self-attention block (CS-SA block) and a consistency regularisation method. Unlike traditional consistency regularisation methods, the proposed method applies consistency regularisation to the labelled data in the supervised task, which was shown to enhance the performance of the model significantly compared to the model without regularisation.

Our method has been demonstrated to be effective for segmenting OAR, achieving state-of-the-art results for the trachea and aorta segmentation and competitive results for the oesophagus and heart segmentation. However, it requires careful selection of hyperparameters, specifically, the consistency learning trade-off coefficient λ and the number of waiting epochs before enabling the consistency loss function.

The key feature of the architecture is the use of self-attention blocks with cosine similarity as the sequence-key similarity function (CS-SA block). Our proposed method replaces the inner product in the self-attention block with a cosine similarity function that incorporates a learnable scaling parameter β. This modification improves the effectiveness of self-attention blocks and has resulted in a significant performance improvement in organ segmentation tasks.

In addition, our approach presents a new perspective on consistency regularisation, which can be applied to other supervised learning tasks. However, further research is required to determine the generalisability of this method to other tasks.

Overall, our proposed CS-SA U-Net architecture with consistency regularisation and CS-SA blocks shows promising results in the field of organ segmentation and provides a new avenue for improving the performance of deep learning models for medical image analysis.

ACKNOWLEDGEMENTS

This research is part of the PID2022-137451OB-I00 and PID2022-137629OA-I00 projects funded by the MICIU/AEIAEI/10.13039/501100011033 and by ERDF/EU.

CONFLICT OF INTEREST STATEMENT

All authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The software that supports the findings of this study are openly available in Gitlab repository at https://gitlab.com/Max2510/cs-sa-unet. The SegTHOR dataset is openly available on the Codalab platform at https://competitions.codalab.org/competitions/21145#participate-get_starting_kit.

ORCID

Maksym Manko https://orcid.org/0000-0001-6851-943X
Anton Popov https://orcid.org/0000-0002-1194-4424
Juan Manuel Gorriz https://orcid.org/0000-0001-7069-1714
Javier Ramirez https://orcid.org/0000-0002-6229-2921

REFERENCES

1. Sung, H., et al.: Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA A Cancer J. Clin. 71(3), 209–249 (2021). https://doi.org/10.3322/caac.21660
2. Lambert, Z., et al.: SegTHOR: segmentation of thoracic organs at risk in CT images. In: 2020 Tenth International Conference on Image Processing Theory, Tools and Applications (IPTA), pp. 1–6. IEEE (2020)
3. Han, M., et al.: Segmentation of CT thoracic organs by multi-resolution VB-nets. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
4. Chen, P., et al.: Two-stage network for OAR segmentation. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
5. Zhang, L., et al.: Segmentation of thoracic organs at risk in CT images combining coarse and fine network. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
6. Kim, S., et al.: A cascaded two-step approach for segmentation of thoracic organs. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
7. Wang, Q., et al.: 3D enhanced multi-scale network for thoracic organs segmentation. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
8. Vesal, S., Ravikumar, N., Maier, A.: A 2D dilated residual U-Net for multi-organ segmentation in thoracic CT. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
9. Gali, M.S.K., Garg, N., Vasamsetti, S.: Dilated U-net based segmentation of organs at risk in thoracic CT images. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
10. Lachinov, D.: Segmentation of thoracic organs using pixel shuffle. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
11. Feng, M., et al.: Multi-organ segmentation using simplified dense V-net with post-processing. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
12. He, T., et al.: Multi-task learning for the segmentation of thoracic organs at risk in CT images. In: Proceedings of the 2019 Challenge on Segmentation of THoracic Organs at Risk in CT Images, SegTHOR@ISBI 2019, April 8, 2019, vol. 2349. CEUR-WS (2019). CEUR Workshop Proceedings
13. He, T., et al.: Multi-task learning for the segmentation of organs at risk with label dependence. Med. Image Anal. 61, 101666 (2020). https://doi.org/10.1016/j.media.2020.101666
14. Isensee, F., et al.: nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. Nat. Methods 18(2), 203–211 (2021). https://doi.org/10.1038/s41592-020-01008-z
15. Gavrilov, A.D., et al.: Preventing model overfitting and under fitting in convolutional neural networks. Int. J. Software Sci. Comput. Intell. 10(4), 19–28 (2018). https://doi.org/10.4018/ijssci.2018100102
16. Xie, Z., et al.: Artificial neural variability for deep learning: on overfitting, noise memorization, and catastrophic forgetting. Neural Comput. 33(8), 2163–2192 (2021). https://doi.org/10.1162/neco_a_01403
17. Bejani, M.M., Ghatee, M.: A systematic review on overfitting control in shallow and deep neural networks. Artif. Intell. Rev. 54(8), 6391–6438 (2021). https://doi.org/10.1007/s10462-021-09975-1

18. Krogh, A., Hertz, J.A.: A simple weight decay can improve generalization. In: Advances in Neural Information Processing Systems, pp. 950–957 (1992)
19. Ng, A.Y.: Feature selection, L 1 vs. L 2 regularization, and rotational invariance. In: Proceedings of the Twenty-First International Conference on Machine Learning, pp. 78 (2004)
20. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International Conference on Machine Learning, pp. 448–456. PMLR (2015)
21. Srivastava, N., et al.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15(1), 1929–1958 (2014)
22. Cubuk, E.D., et al.: Auto augment: learning augmentation strategies from data. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 113–123 (2019)
23. Buslaev, A., et al.: Albumentations: fast and flexible image augmentations. Information 11(2), 125 (2020). https://doi.org/10.3390/info11020125
24. Zhang, H., et al.: mixup: beyond empirical risk minimization. In: 6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018. Conference Track Proceedings (2018)
25. Yun, S., et al.: Cutmix: regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 6023–6032 (2019)
26. Harris, E., et al.: Fmix: enhancing mixed sample data augmentation. arXiv preprint arXiv:2002.12047 (2020)
27. Verma, V., et al.: Manifold mixup: better representations by interpolating hidden states. In: International Conference on Machine Learning, pp. 6438–6447. PMLR (2019)
28. An, G.: The effects of adding noise during backpropagation training on a generalization performance. Neural Comput. 8(3), 643–674 (1996). https://doi.org/10.1162/neco.1996.8.3.643
29. Neelakantan, A., et al.: Adding gradient noise improves learning for very deep networks. arXiv preprint arXiv:1511.06807 (2015)
30. You, Z., et al.: Adversarial noise layer: regularize neural network by adding noise. In: 2019 IEEE International Conference on Image Processing (ICIP), pp. 909–913. IEEE (2019)
31. French, G., et al.: Semi-supervised semantic segmentation needs strong, varied perturbations. In: 31st British Machine Vision Conference 2020, BMVC 2020, Virtual Event, UK, September 7-10, 2020. BMVA Press (2020)
32. Zou, Y., et al.: Pseudoseg: designing pseudo labels for semantic segmentation. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021 (2021)
33. Chen, X., et al.: Semi-supervised semantic segmentation with cross pseudo supervision. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2613–2622 (2021)
34. Zhang, H., et al.: Consistency regularization for generative adversarial networks. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020 (2020)
35. Tack, J., et al.: Consistency regularization for adversarial robustness. arXiv preprint arXiv:2103.04623 (2021)
36. Pham, H., et al.: Meta pseudo labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11557–11568 (2021)
37. Jeong, J., et al.: Consistency-based semi-supervised learning for object detection. Adv. Neural Inf. Process. Syst. 32, 10759–10768 (2019)
38. Verma, V., et al.: Interpolation consistency training for semi-supervised learning. Neural Network. 145, 90–106 (2022). https://doi.org/10.1016/j.neunet.2021.10.008
39. Kim, D., et al.: Selfreg: self-supervised contrastive regularization for domain generalization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9619–9628 (2021)
40. Li, J., Xiong, C., Hoi, S.C.: Comatch: semi-supervised learning with contrastive graph regularization. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9475–9484 (2021)
41. He, K., et al.: HF-UNet: learning hierarchically inter-task relevance in multi-task U-net for accurate prostate segmentation in CT images. IEEE Trans. Med. Imag. 40(8), 2118–2128 (2021). https://doi.org/10.1109/tmi.2021.3072956
42. Shi, F., Zhang, T.: A multi-task network with distance–mask–boundary consistency constraints for building extraction from aerial images. Rem. Sens. 13(14), 2656 (2021). https://doi.org/10.3390/rs13142656
43. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)
44. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. In: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015. Conference Track Proceedings (2015)
45. Devlin, J., et al.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL (2019)
46. Brown, T., et al.: Language models are few-shot learners. In: Larochelle, H., et al. (eds.) Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901. Curran Associates, Inc. (2020). https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf
47. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 234–241. Springer (2015)
48. Milletari, F., et al.: Fully convolutional neural networks for volumetric medical image segmentation. In: 2016 Fourth International Conference on 3D Vision (3DV), pp. 565–571. IEEE (2016)
49. Zhou, Z., et al.: UNet++: redesigning skip connections to exploit multiscale features in image segmentation. IEEE Trans. Med. Imag., 1–1 (2019)
50. Dosovitskiy, A., et al.: An image is worth 16x16 words: transformers for image recognition at scale. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021 (2021)
51. Liu, Z., et al.: Swin transformer: hierarchical vision transformer using shifted windows. In: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021, pp. 9992–10002. IEEE (2021)
52. Dai, Z., et al.: Coatnet: marrying convolution and attention for all data sizes. Adv. Neural Inf. Process. Syst. 34, 3965–3977 (2021)
53. He, K., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
54. Jadon, S.: A survey of loss functions for semantic segmentation. In: 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), pp. 1–7. IEEE (2020)
55. Henry, A., et al.: Query-key normalization for transformers. In: Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020. Vol. EMNLP 2020 of Findings of ACL, pp. 4246–4253. Association for Computational Linguistics (2020)
56. Tseng, H.J., et al.: Imaging foreign bodies: ingested, aspirated, and inserted. Ann. Emerg. Med. 66(6), 570–582 (2015). https://doi.org/10.1016/j.annemergmed.2015.07.499
57. Deng, J., et al.: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. Ieee (2009)
58. Zhang, M.R., et al.: Lookahead optimizer: k steps forward, 1 step back. In: Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pp. 9593–9604 (2019)
59. Liu, L., et al.: On the variance of the adaptive learning rate and beyond. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020 (2020)
60. Loshchilov, I., Hutter, F.SGDR: Stochastic gradient descent with warm restarts. In: 5th International Conference on Learning Representations,

ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings (2017)
61. Ke, Z., et al.: Guided collaborative training for pixel-wise semi-supervised learning, 429–445 (2020)
62. Tarvainen, A., Valpola, H.: Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In: Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pp. 1195–1204 (2017)
63. Zou, Y., et al.: Pseudoseg: designing pseudo labels for semantic segmentation. arXiv preprint arXiv:2010.09713 (2020)
64. Zhang, H., et al.: Self-attention generative adversarial networks. In: International Conference on Machine Learning, pp. 7354–7363. PMLR (2019)
65. Wilcoxon, F.: Individual comparisons by ranking methods. Biometrics Bull. 6, 80–83 (1945). https://doi.org/10.2307/3001968

How to cite this article: Manko, M., et al.: Improved organs at risk segmentation based on modified U-Net with self-attention and consistency regularisation. CAAI Trans. Intell. Technol. 1–16 (2024). https://doi.org/10.1049/cit2.12303