Mask R-CNN Transfer Learning Variants for Multi-organ Medical Image Segmentation

Hongjian Lem
Department of Computer Science
Royal Holloway, University of London
Surrey, TW20 0EX, UK
hong.lem.2021@live.rhul.ac.uk

Li Zhang
Department of Computer Science
Royal Holloway, University of London
Surrey, TW20 0EX, UK
li.zhang@rhul.ac.uk

***Abstract*—Medical abdomen image segmentation is a challenging task owing to discernible characteristics of the tumour against other organs. As an effective image segmenter, Mask R-CNN has been employed in many medical imaging applications, e.g. for segmenting nucleus from cytoplasm for leukaemia diagnosis and skin lesion segmentation. Motivated by such existing studies, this research takes advantage of the strengths of Mask R-CNN in leveraging on pre-trained CNN architectures such as ResNet and proposes three variants of Mask R-CNN for multi-organ medical image segmentation. Specifically, we propose three variants of the Mask R-CNN transfer learning model successively, each with a set of configurations modified from the one preceding. To be specific, the three variants are (1) the traditional transfer learning with customized loss functions with comparatively more weightage on the segmentation performance, (2) transfer learning based on Mask R-CNN with deepened re-trained layers instead of only the last two/three layers as in traditional transfer learning, and (3) the fine-tuning of Mask R-CNN with expansion of the Region of Interest pooling sizes. Evaluating using Beyond-the-Cranial-Vault (BTCV) abdominal dataset, a well-established benchmark for multi-organ medical image segmentation, the three proposed variants of Mask R-CNN obtain promising performances. In particular, the empirical results indicate the effectiveness of the proposed adapted loss functions, the deepened transfer learning process, as well as the expansion of the RoI pooling sizes. Such variations account for the great efficiency of the proposed transfer learning variant schemes for undertaking multi-organ image segmentation tasks.**

Keywords—Mask R-CNN, Medical Image Segmentation, Customized Loss Function, Transfer learning

I. Introduction

Image segmentation refers to the problem of precisely segmenting an image such that objects of interest could be identified and distinguished from the background at the pixel-level [1-10]. One key use was in the medical field, e.g. multi-organ medical image segmentation, where it has been employed to automate the masking out of anatomy areas of interest on medical image data such as Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) scans. It would otherwise be time-consuming if performed manually.

Mask R-CNN, proposed by He et al. [11], is one of the most popular techniques for "everyday" image segmentation problems. In comparison with U-Net [12, 13] which has been deployed extensively in the medical image segmentation tasks [13], repurposing Mask R-CNN for the medical field has gained increasing attention owing to its superior performance for image segmentation and classification. Moreover, a key feature of Mask R-CNN is its ability to leverage on established pre-trained CNN models such as ResNet [14] for feature extraction, a technique which could potentially be useful for tumour/lesion detection, segmentation and classification.

This research therefore sets out to establish the viability of applying Mask R-CNN transfer learning for multi-organ medical image segmentation. For example, we aim to explore whether and to what extent Mask R-CNN pre-trained weights could be transfer-learned for the medical image domain, as well as the development of a preliminary study on what adaptations could be useful.

To this end, we propose three variations of transfer learning using Mask R-CNN, each built successively based on observations from the preceding iteration. For training and evaluation, we used the Beyond the Cranial Vault abdominal images (BTCV) dataset [15] which is a well-established benchmark for multi-organ medical image segmentation research comprising of 50 randomly selected abdomen CT scans collected under the supervision of the Institutional Review Board (IRB). We elected to use this dataset as it comprised a comparatively large number of classes (13 abdominal organs manually labelled). As a challenging dataset, it may enable us to draw more conclusive interpretations. We summarize our key contributions below.

- This research exploits the viability of Mask R-CNN transfer learning for multi-organ medical image segmentation. The three variants of transfer learning using Mask R-CNN are proposed for image segmentation, i.e. (1) the traditional transfer learning with customized loss functions which has a higher weight on the segmentation performance for the loss calculation, (2) transfer learning based on a Mask R-CNN with deepened number of re-trained layers in comparison with purely the last 2 or 3 layers as in traditional transfer learning, and (3) a Mask R-CNN-based transfer learning with expansion of the Region of Interest (RoI) pooling sizes.

- Preliminary experimentation surfaced adaptations for better performance. Specifically, we doubled the weightage for per-pixel Mask cross-entropy relative to other loss components for more effective loss function convergence. We doubled the RoI pool sizes for the Box RoI head (7x7→14x14) and Mask RoI head (14x14→28x28) which attain improved accuracy for this challenging multi-organ segmentation tasks.

The rest of the paper is organized as follows. We present existing studies and well-known segmentation models in Section 2. The proposed three variants of Mask R-CNN based transfer learning with revised loss functions, deepened re-trained layers and expansion of the RoI pooling sizes in Section 3. Sections 4 and 5 present the detailed evaluation performed and respective research findings. We conclude this research and identify future directions in Section 6.

II. Related Work

In this section, we present several state-of-the-art deep neural networks for image segmentation.

A. *Overview of Mask R-CNN Architecture*

Mask R-CNN extends the works of "Faster R-CNN" by Ren et al. [16] and "Fast R-CNN" by Ross Girshick [17]. It operates in two stages in a similar manner as those for Faster R-CNN. In the first stage, it identifies possible RoIs. Such anchor ROIs would then be forwarded to the downstream components in the second stage for bounding box detection, classification, and mask generation. Architecturally, Mask R-CNN consists of four key components (Backbone, RPN, Box RoI Head, and Mask RoI Head) which we briefly describe below.

Backbone Feature Pyramid Network (FPN): The backbone is a Convolutional Neural Network (CNN) for feature extraction over the entire image. Mask R-CNN implementation generally leverages on established pre-trained CNN architectures such as VGG [18] or ResNet [14]. In this research, we used the ResNet-50 architecture pre-trained on the COCO dataset as the backbone.

Region Proposal Network (RPN): The RPN receives the extracted feature maps from the backbone and proposes a set of bounding boxes that contain foreground objects. Internally, it is a convolutional network that generates a set of k anchor boxes (k is a Mask R-CNN parameter) at each pixel location. Overlapping anchor boxes are merged using Non-maximum suppression (NMS).

Box and Mask RoI Heads: The Box RoI head and the Mask RoI head predict the bounding boxes and segmentation masks respectively for a set of proposed RoIs. Key to both RoI Head components is a pre-processing step known as "RoI pooling". RoI pooling "cookie-cuts" the stack of feature maps corresponding to the RoIs, then resizes each "cut-out" as a uniform grid for processing.

B. *Overview of U-Net Architecture*

U-Net is a fully convolutional network configured as an encoder-decoder architecture first presented by Ronneberger et al. [13] for the purpose of biomedical image segmentation. In the encoding path, features are extracted by each successive layer downsampling feature maps via pooling operations but doubling the number of channels. The decoding path is a near mirror image of the encoding path, with the pooling operations replaced by upsampling operators, forming a "U-shaped" architecture. Additionally, each layer in the decoding path is also augmented with information from its corresponding encoder layer to preserve high resolution information.

One of the most well-known and best-performing U-Net-based model was the nnU-Net ("no new U-Net") developed by Isensee et al. [19]. nnU-Net is a framework residing on a set of three basic U-Net models (2D U-Net, 3D Unet, and UNet Cascade) that automatically adapts its architectures to the given image geometry, including the training and preprocessing pipelines. From the three basic models, the framework automatically selects the most appropriate model or ensemble of two models for each task. Their work claimed that the basic U-Net model was effective enough without any architectural tweaks such as residual connections. They also found that the design of the training and pre-processing pipelines was more impactful to performance than architectural tweaks.

C. *Other Related Work*

Shu et al. [20] applied Mask R-CNN for multi-organ medical image segmentation. In their work, Mask R-CNN was trained on a (non-publicly available) dataset comprising of CT scans of 44 esophageal cancer patients with a total of 4341 CT images, where each image was labelled with 5 organs (heart, left lung, right lung, PTV, and CTV). It achieved reasonable performances with an average Dice coefficient of 94.48% per organ. To the best of our knowledge, despite the employing of Mask R-CNN for a variety of medical segmentation tasks, there are no published studies on Mask R-CNN on standardized benchmark data sets such as BTCV [15] or Medical Segmentation Decathlon [21], which motivates this research.

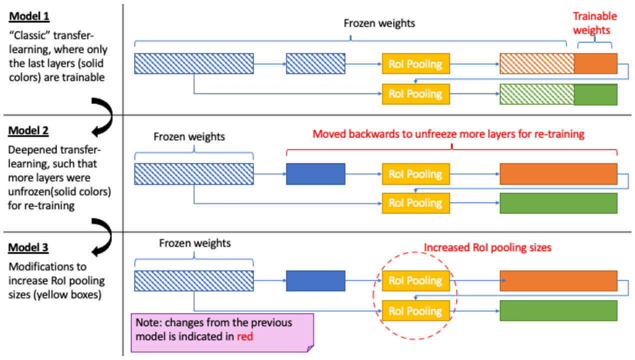
Model 1
"Classic" transfer-learning, where only the last layers (solid colors) are trainable
Frozen weights
Trainable weights
RoI Pooling
RoI Pooling
Model 2
Deepened transfer-learning, such that more layers were unfrozen(solid colors) for re-training
Frozen weights
Moved backwards to unfreeze more layers for re-training
RoI Pooling
RoI Pooling
Model 3
Modifications to increase RoI pooling sizes (yellow boxes)
Frozen weights
Increased RoI pooling sizes
RoI Pooling
RoI Pooling
Note: changes from the previous model is indicated in red

Fig. 1. Overview of three proposed variant models for image segmentation

III. The Proposed Mask R-CNN Variant Methods

To establish the viability and characteristics of Mask R-CNN transfer learning for medical image segmentation, we propose three Mask R-CNN variant models in succession, each trained via customised transfer learning with a different set of configurations. We adopted an iterative approach which make use of empirical observations from a preceding iteration to guide the implementation of the next proposed model. Our code leveraged on the PyTorch Mask R-CNN implementation [22, 23] pre-trained with COCO [3] with ResNet-50 [23] as its backbone.

As shown in Fig. 1, we provide a simplified overview of the system architectures which shows how each model evolved from the preceding method. We introduce each model in the respective subsections below (Subsections III-B, C, and D).

A. Dataset

BTCV consists of a total of 50 3D CT scans captured during portal venous contrast phase with variable volume sizes, with each CT scan consisting of multiple slices. The organs for segmentation include, (1) spleen, (2) right kidney, (3) left kidney, (4) gallbladder, (5) esophagus, (6) liver, (7) stomach, (8) aorta, (9) inferior vena cava, (10) portal vein and splenic vein, (11) pancreas, (12) right adrenal gland, and (13) left adrenal gland.

For this research, we used only the axial view for training and evaluation. In addition, while the dataset is split into training and test subfolders, we used data only from the training subfolder, which consisted of 30 CT scans containing 3778 (axial) slices in total. This was because the ground truth labels were not released for the test subfolder.

B. Model 1: "Classic" Transfer Learning

The first proposed model adopted the classic approach to transfer learning, where only the last layer was re-trained using the new BTCV dataset. In this case, the last layers for both the Box RoI head and Mask RoI heads were replaced and trained to reflect the required number of classes: 14 (13 organs + background) instead of the original 91 (COCO dataset). Weights for all other layers were frozen during training. This is illustrated in Fig. 2.

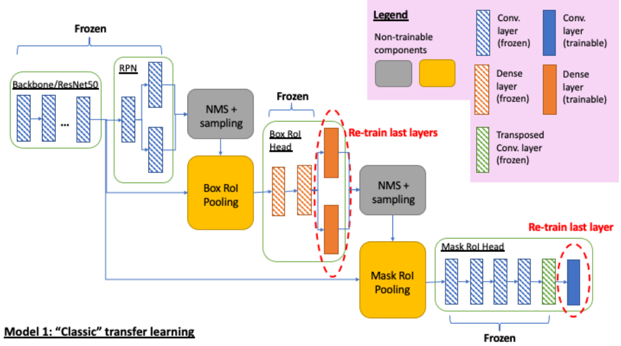

Fig. 2. Model 1 system architecture (with a modified loss function)

1) Loss functions

The standard Mask R-CNN loss function is a linear sum of five separate loss functions obtained from the RPN, Box RoI head and Mask RoI head. As the RPN component was not trained in this transfer learning model, only three of the five loss function constituents were relevant for backpropagation, as described below.

1. $\mathcal{L}_{boxCls}$ (Box RoI Head): Cross-entropy (softmax) classification loss against the true label [17].

2. $\mathcal{L}_{boxReg}$ (Box RoI Head): Regression loss of the predicted bounding box t^u for the true class u, where t^u is represented by its x and y coordinates, as well as its width and height [17].

3. $\mathcal{L}_{mask}$ (Mask RoI Head): Average per-pixel cross-entropy loss against the ground-truth mask.

Additionally, we introduced one modification to the loss function. While in the standard Mask R-CNN training loss, all loss constituents had equal weightage, early observations suggested that $\mathcal{L}_{mask}$ generally starts off with considerably higher loss values and would therefore benefit from a comparatively higher training gradient during backpropagation. For this reason, we experimented with assigning higher weightages to the $\mathcal{L}_{mask}$ loss constituent and in our study, we found that doubling the weightage of $\mathcal{L}_{mask}$ performed comparatively better. Formally, the new customized loss function is defined as follows.

$$\mathcal{L}_{model1} = (\mathcal{L}_{boxCls} + \mathcal{L}_{boxReg} + 2 \times \mathcal{L}_{mask})/4 \quad (1)$$

Such a loss function is able to give more emphasis on the mask prediction performance to better inform the backpropagation and weight adjustment during training.

2) Training procedure

Taking reference from the original study of Mask R-CNN, an image-centric training was adopted such that each mini-batch has two images. Stochastic Gradient Descent (SGD) was used with the following hyper-parameters for network training, i.e. learning rate=0.02, Momentum=0.9 and Weight decay= 0.005. Additionally, a three-step learning rate schedule was applied, which reduced the learning rate from 0.02 to 0.002 and finally 0.0002 over the course of the training epochs.

3) Training results and analysis

The training loss exhibited expected behaviour with sharply descending loss results that approached a plateau at epoch 5. However, its eventual loss value was not fully satisfactory. A closer analysis of the training loss constituents revealed the following: $\mathcal{L}_{obj}$ was abnormally high with a loss value of 0.419, which we indicated in orange shading in Table I below.

TABLE I. BREAKDOWN BY LOSS CONSTITUENTS AT THE END OF TRAINING, AVERAGED OVER 5 FOLDS (ABNORMAL HIGH $\mathcal{L}_{obj}$ J HIGHLIGHTED IN ORANGE)

	$\mathcal{L}_{boxCls}$	$\mathcal{L}_{boxReg}$	$\mathcal{L}_{mask}$	$\mathcal{L}_{obj}$	$\mathcal{L}_{rpnReg}$
Model 1	0.10417	0.07631	0.27171	0.419	0.02772

This was significant because while $\mathcal{L}_{obj}$ (which is described in a later section) was not used in the loss function calculation, its high loss value was indicative of the RPN component's inability to clearly distinguish the foreground objects from the background to propose suitable regions of interest for downstream processing. This likely caused poor overall performance of Model 1 based on traditional transfer learning.

Owing to the vast variations between daily objects in COCO dataset and the BTCV medical organ data, it is was not unexpected that Mask R-CNN pre-trained on the COCO dataset shows limitations in tackling the segmentation tasks in the new domain under the traditional transfer learning scheme. We subsequently propose Model 2 with deepened customised transfer learning to tackle the above challenges.

C. *Model 2: Moving transfer learning inwards to unfreeze more layers*

To follow up on our aforementioned hypothesis, we then unfreeze the RPN component weights for re-training in Model 2. Additionally, because the Box RoI Head and the Mask RoI Head components receive input from the RPN component, their pre-trained weights would no longer be valid and therefore needed to be re-trained as well.

This architecture for Model 2 is illustrated in Fig. 3. In this model, we unfroze all layers down to the RPN component for re-training, effectively "moving" transfer learning inwards, freezing only the backbone component weights. In total, 13 layers in Mask R-CNN across three components would be unfrozen and re-trained. The layers are listed in Table II.

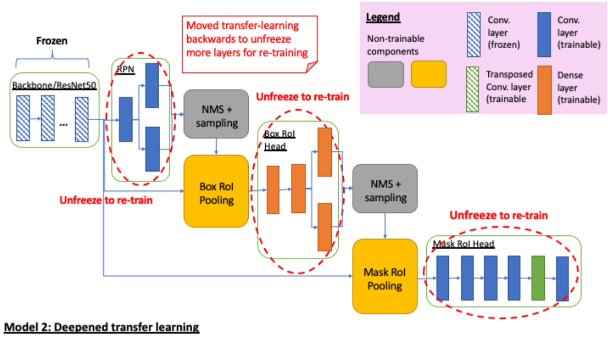

Fig. 3. Model 2 System architecture with deepened and customised transfer learning

TABLE II. THE LAST 13 MASK R-CNN LAYERS THAT HAVE BEEN RE-TRAINED

Layer description (PyTorch convention)
RPN
Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
Conv2d(256, 3, kernel_size=(1, 1), stride=(1, 1))
Conv2d(256, 12, kernel_size=(1, 1), stride=(1, 1))
Box RoI Head
Linear(in_features=12544, out_features=1024, bias=True)
Linear(in_features=1024, out_features=1024, bias=True)
Linear(in_features=1024, out_features=14, bias=True)
Linear(in_features=1024, out_features=56, bias=True)
Mask RoI Head
Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
ConvTranspose2d(256, 256, kernel_size=(2, 2), stride=(2, 2))
Conv2d(256, 14, kernel_size=(1, 1), stride=(1, 1))

1) *Loss Functions*

Since the RPN component would be re-trained, all five loss function constituents would now be incorporated into the training loss calculation for backpropagation. In addition to the three ($\mathcal{L}_{boxCls}$, $\mathcal{L}_{boxReg}$, $\mathcal{L}_{mask}$) described in the previous section, the "new" additional loss components are the following obtained from RPN.

4. $\mathcal{L}_{obj}$ (RPN) denotes the binary cross-entropy classification loss on the anchor (object or background), also commonly known as the "objectness" loss according to [16]. As a recap, in Equation (2), although they seemed similar, $\mathcal{L}_{boxCls}$ refers to a multi-class classification, while $\mathcal{L}_{obj}$ is strictly binary and reflects whether a region (proposed by the RPN) contains an object or not.

5. $\mathcal{L}_{rpnReg}$ (RPN) refers to the regression loss on the predicted coordinates for positive anchors. Each coordinate is represented by a vector containing four elements, which were its top-left location of the predicted bounding box as well as its width and height [16]. Likewise, although $\mathcal{L}_{boxReg}$ and $\mathcal{L}_{rpnReg}$ seemed similar, the difference is that $\mathcal{L}_{boxReg}$ receives the predicted bounding box coordinates predicted for all classes per anchor, while $\mathcal{L}_{rpnReg}$ receives only one set of bounding box coordinates per anchor.

The loss function is fundamentally identical to that of Model 1 shown in Equation (1) albeit with the addition of $\mathcal{L}_{obj}$ and $\mathcal{L}_{rpnReg}$ loss constituents that were not previously relevant in Model 1. Similar to Model 1, the weightage for $\mathcal{L}_{mask}$ was doubled compared to that of other loss constituents. Formally, the loss function is defined as follows.

$$\mathcal{L}_{model2} = (\mathcal{L}_{boxCls} + \mathcal{L}_{boxReg} + 2 \times \mathcal{L}_{mask} + \mathcal{L}_{obj} + \mathcal{L}_{rpnReg})/6 \quad (2)$$

2) *Training results and analysis*

Overall training loss for Model 2 improved substantially. More significantly, as shown in Table III, we see a vast improvement in the $\mathcal{L}_{obj}$ loss constituent of 0.03338 for Model 2, an order of magnitude smaller compared to Model 1's loss of 0.419.

TABLE III. THE LOSS RESULTS FOR MODEL 1 VS MODEL 2 (THE IMPROVEMENT IN $\mathcal{L}_{OBJ}$ LOSS HIGHLIGHTED IN ORANGE FOR MODEL 1 AND GREEN FOR MODEL 2 RESPECTIVELY)

	$\mathcal{L}_{boxCls}$	$\mathcal{L}_{boxReg}$	$\mathcal{L}_{mask}$	$\mathcal{L}_{obj}$	$\mathcal{L}_{rpnReg}$
Model1	0.10417	0.07631	0.27171	0.419	0.02772
Model2	0.06739	0.10084	0.13726	0.03338	0.01392

This confirmed our hypothesis that the high $\mathcal{L}_{obj}$ loss observed in the previous model was not inherent in the Mask R-CNN architecture but only due to incompatible pre-trained RPN component weights, which was not surprising given the vast difference between the "everyday" images of the COCO dataset and the BTCV medical images.

Additionally, the re-training seemed to have a "knock-on" effect on the other loss constituents as well, all of which had improved considerably from Model 1.

D. Model 3: Experimentation with increasing RoI pool size

With Model 2 establishing the necessity of moving transfer learning inwards to unfreeze and re-train the RPN component and retain only the backbone weights, it presented an opportunity to experiment with architectural tweaks to the re-trained layers so that the model is able to better capture knowledge in the new medial domain.

To this end, we chose to experiment with expanding the RoI pool sizes in Model 3. Our hypothesis was that because the RoI pool size affects the resolution and therefore amount of information contained in the "cut-outs", increasing the RoI pool sizes should help the Box RoI and Mask RoI Heads make more accurate predictions.

The default pool sizes for Box RoI Head and Mask RoI Head are 7x7 and 14x14 respectively. In Model 3, we double the pool sizes for both heads. Fig. 4 illustrates the architecture for Model 3. Other than the doubling of RoI pool sizes, its architecture is identical to that of Model 2.

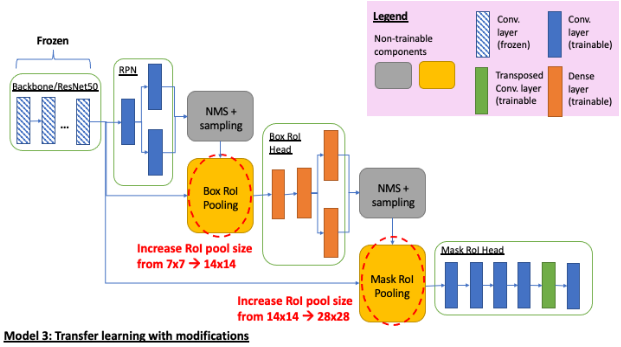

Fig. 4. Model 3 system architecture with increased RoI pooling sizes

1) Training results and analysis

Table IV below showed the loss constituents for Model 3 alongside the previous two models for comparison. Of interest is the comparison between Model 3 and Model 2, where the better loss constituent was highlighted in green while the less performant counterpart in orange.

This suggested that despite its higher overall loss, Model 3 should actually perform better than Model 2 since it obtained better results in four out of five loss constituents. In addition, we conjectured that Model 3's higher $\mathcal{L}_{mask}$ constituent loss was only a side-effect of its higher resolution and not suggestive of worse performance. The following section on Evaluation would provide further empirical results to support our conjecture.

TABLE IV. LOSS CONSTITUENTS FOR ALL 3 MODELS (BETWEEN MODEL 2 AND MODEL 3, THE BETTER SCORE IS HIGHLIGHTED IN GREEN, WHILE THE POORER SCORE IS INDICATED IN ORANGE WITH THE WORST IN RED)

	$\mathcal{L}_{boxCls}$	$\mathcal{L}_{boxReg}$	$\mathcal{L}_{mask}$	$\mathcal{L}_{obj}$	$\mathcal{L}_{rpnReg}$
Model 1	0.10417	0.07631	0.2717	0.419	0.02772
Model 2	0.06739	0.10084	0.1373	0.03338	0.01392
Model 3	0.05808	0.09712	0.1477	0.03313	0.0139

IV. EVALUATION

For evaluation, we used 5-fold cross-validation using the BTCV training dataset (comprising of 30 CT scan images) on each model. In addition to the three Mask R-CNN models as described in the previous section, we also implemented a basic U-Net model using open-source code [23] to provide a baseline for comparison, which we re-trained the U-Net from scratch to ensure relevancy to the BTCV dataset.

For the evaluation metric, we used a two-class (Background/Foreground) Dice Coefficient which is described in the subsection below.

A. Evaluation Metric

We adopted Dice coefficient, the de-facto standard for evaluating image segmentation performance. It is defined as follows, where X refers to the predicted mask, while Y refers to the ground-truth mask:

$$Dice = (2 \times |X \cap Y|)/(|X| + |Y|) \qquad (3)$$

Instead of computing the Dice coefficient for each organ, we flattened masks for all organs into a single class to transfer our evaluation into a two-class (background vs foreground) segmentation task. We elected not to perform evaluation on a per-organ basis as the relatively large number of classes (13) with varied shapes and sizes in the BTCV dataset would introduce additional variability which would confound analysis and detract from our aim of establishing viability.

Finally, in-line with the convention for medical image segmentation, evaluation was conducted on a volumetric basis, i.e. the Dice coefficient was computed on the entire 3D scan dataset based on 5-fold cross-validation.

B. Evaluation Results

Table V below shows the background-foreground Dice scores of each model, averaged across 5-folds. As expected, Model 1 performed under-par due to its pre-trained RPN component weights being incompatible with the BTCV dataset which affected its downstream predictions. On the other hand, both Models 2 and 3 achieved superior performance and outperformed the U-Net implementation, with Model 3 being the best performing model among all four models. This supported our earlier hypothesis that Model 3's higher $\mathcal{L}_{mask}$ training loss was only a side-effect of its higher RoI pool resolution and not indicative of poorer performance compared to Model 2.

TABLE V. DICE SCORES OF EACH MODEL FOR BACKGROUND-FOREGROUND SEGMENTATION, AVERAGED ACROSS 5-FOLDS

	Model 1	Model 2	Model 3	U-Net
Background	0.9846	0.99343	0.99366	0.99326
Foreground	0.64406	0.85742	0.86158	0.84893

To help us visualize the quality of the predictions, we produced two example mask predictions using our best performing model, i.e. the customised Mask R-CNN Model 3. For both examples as shown in Figs. 5 and 6, moving clockwise from the top-left are the original CT scan slice, (top-right) ground-truth masks overlaid and colour-coded to differentiate between different organs, (bottom-right) predicted per-organ masks, (bottom-left) predicted (flattened) foreground mask in turquoise.

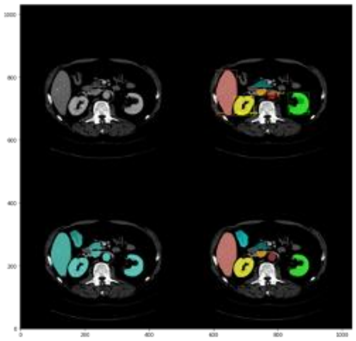

Fig. 5. Predicted mask example 1. Clockwise from top-left: Original CT scan, Ground-truth mask overlaid, Predicted per-organ masks, and Predicted flattened (fg) mask.

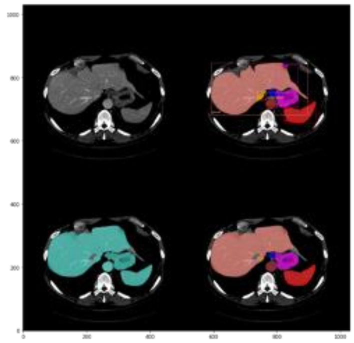

Fig. 6. Predicted mask example 2. Clockwise from top-left: Original CT scan, Ground-truth mask overlaid, Predicted per-organ masks, and Predicted flattened (fg) mask.

V. DISCUSSION

The evaluation results confirmed the viability of the three proposed Mask R-CNN transfer learning models for multi-organ medical image segmentation. Specifically, the empirical results indicate that instead of unfreezing only the last layers as per the "classic" transfer learning approach, we found it more effective to move transfer learning inwards and unfreeze more layers (13 layers) for re-training, retaining only the pre-trained backbone weights. Both Model 2 and Model 3 adopted this transfer learning approach and achieved improved performance that outperformed Model 1 and U-Net implementation.

Finally, Model 3's improved performance over Model 2 indicated that increasing the RoI pool size was beneficial. This was despite its slightly higher $\mathcal{L}_{mask}$ loss value during training, which we hypothesized is only a side effect of its higher resolution outputs.

A. *"Sneak peek" of Mask R-CNN per-organ performance*

While we had not set out to conduct a thorough evaluation of Mask R-CNN on a per-organ basis, we wanted to take a "sneak peek" of per-organ Dice coefficient for the Mask R-CNN models to round out our assessment.

Table VI below shows the results for Model 2 and Model 3. We omitted Model 1 to focus on Model 2 and Model 3 since the two-class (background/foreground) Dice score results already demonstrated that Model 1 performance was suboptimal. Likewise, per-organ classification for the basic U-Net model was not implemented for this "sneak peek" since we are primarily interested only in Mask R-CNN here.

TABLE VI. PER-ORGAN DICE COEFFICIENT COLLECTED ON MASK R-CNN MODELS 2 AND 3

Organ	Model 2	Model 3
spleen	0.877	0.859
right kidney	0.846	0.862
left kidney	0.853	0.86
gallbladder	0.458	0.508
esophagus	0.621	0.62
liver	0.906	0.909
stomach	0.641	0.644
aorta	0.837	0.849
inf. vena cava	0.708	0.718
portal and splenic vein	0.065	0.121
pancreas	0.381	0.407
right adrenal gland	0247	0.339
left adrenal gland	0.227	0.285

The per-organ Dice Coefficient results looked promising with generally promising scores. For example, the liver obtained a score of > 0.9 in both models. This was remarkable given that we had not implemented any optimization techniques such as data augmentation [24-30], fine-tuning of training parameters (e.g. activation functions) [31-45], or ensemble techniques [46-56], which had been shown to be critical for obtaining good performance for medical image segmentation tasks.

In addition, we noted that Model 3 outperformed Model 2 for most of the organs, further lending support to our hypothesis that increasing the RoI pool size would improve performance for medical image segmentation.

However, we also observed less performant results for the portal and splenic veins, pancreas, and adrenal glands in this set of preliminary comparison, giving us a glimpse of potential challenges ahead, some of which could be inherent in medical image datasets.

One plausible hypothesis was that because Mask R-CNN works by identifying centres of mass ("anchor") for each object, it would not work as well for detecting objects that are virtually enclosed by another larger object such that its centre of mass could not be clearly distinguished from the larger object. Fig. 7 illustrates one such example where the portal and splenic veins (in green) was enclosed entirely within the liver in the ground-truth mask (middle image). Mask R-CNN's prediction (right image) was not able to detect this although it predicted other organs very well in this instance.

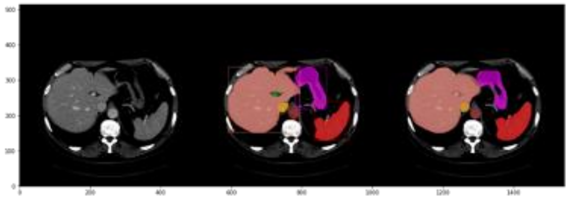

Fig. 7. Original image (left), the organ of Portal and Splenic veins (green) which is entirely enclosed within the liver in the ground-truth (middle), but not captured in the prediction (right) result.

Another hypothesis was that Mask R-CNN relied on being able to first learn distinguishing features to detect individual organs in the images. This puts smaller organs such as the adrenal glands at a disadvantage, as they appear in much fewer slices, thus giving Mask R-CNN less opportunities to learn their features. A follow-up research could investigate augmenting the dataset such that the smaller organs would be presented more frequently during training. Overall, the preliminary per-organ results are promising and lent further support to the viability of applying Mask R-CNN transfer learning to the medical image segmentation domain.

VI. Conclusion and Future Work

In this research, we established three Mask R-CNN variant models with customised loss functions, deepened transfer learning as well as the expansion of RoI pooling sizes, for undertaking medical image segmentation. Results from this research were promising and confirmed the viability of applying these proposed Mask R-CNN transfer learning models for multi-organ medical image segmentation. For future research, we aim to incorporate data augmentation [57], ensemble methods [58-60] and hyper-parameter optimization [61, 62] with the proposed variant methods to further enhance network performance.

References

[1] Long, J., Shelhamer, E. and Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 3431-3440).

[2] Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N. and Terzopoulos, D., 2021. Image segmentation using deep learning: A survey. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 44(7), pp.3523-3542.

[3] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. and Zitnick, C.L., 2014. Microsoft coco: Common objects in context. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13 (pp. 740-755). Springer International Publishing.

[4] Everingham, M., Van Gool, L., Williams, C.K., Winn, J. and Zisserman, A., 2010. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88, pp.303-338.

[5] S. Slade, L. Zhang, Y. Yu and C.P. Lim, 2022. An evolving ensemble model of multi-stream convolutional neural networks for human action recognition in still images. *Neural Computing and Applications*, pp.1-27.

[6] Tan, T.Y., Zhang, L. and Lim, C.P., 2020. Adaptive melanoma diagnosis using evolving clustering, ensemble and deep neural networks. *Knowledge-Based Systems*, *187*, p.104807.

[7] S.C. Neoh, W. Srisukkham, L. Zhang, S. Todryk, B. Greystoke, C.P. Lim, A. Hossain and N. Aslam. An Intelligent Decision Support System for Leukaemia Diagnosis using Microscopic Blood Images, *Scientific Reports*, 5 (14938). 2015.

[8] Bandara, I., Zhang, L. and Mistry, K., 2022, July. Deep Learning Based Short-Term Total Cloud Cover Forecasting. In *2022 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE.

[9] Zhang, L. and Lim, C.P., 2020. Intelligent optic disc segmentation using improved particle swarm optimization and evolving ensemble models. *Applied Soft Computing*, *92*, p.106328.

[10] W. Srisukkham, L. Zhang, S.C. Neoh, S. Todryk and C.P. Lim. Intelligent Leukaemia Diagnosis with Bare-Bones PSO based Feature Optimization, *Applied Soft Computing*, 56. pp. 405-419. 2017.

[11] He, K., Gkioxari, G., Dollár, P. and Girshick, R., 2020. Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 386-397.

[12] Paperswithcode.com. https://paperswithcode.com/sota/medical-image-segmentation-on-synapse-multi (accessed on 31 July 2022)

[13] Ronneberger, O., Fischer, P. and Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Germany, Part III 18 (pp. 234-241).

[14] He, K., Zhang, X., Ren, S. and Sun, J., 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[15] Landman, B., Xu, Z., Igelsias, J.E., Styner, M., Langerak, T.R. and Klein, A., 2015. *Multi-atlas labeling beyond the cranial vault workshop and challenge*.

[16] Ren, S., He, K., Girshick, R. and Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28.

[17] Girshick, R. Fast R-CNN. 2015. In *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 1440-1448.

[18] Simonyan, K. and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. *The 3rd International Conference on Learning Representations (ICLR)*.

[19] Isensee, F., Jaeger, P.F., Kohl, S.A., Petersen, J. and Maier-Hein, K.H., 2021. nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, 18(2), pp.203-211.

[20] Shu, J.H., Nian, F.D., Yu, M.H. and Li, X., 2020. An improved mask R-CNN model for multi-organ segmentation. *Mathematical Problems in Engineering*, 2020, pp.1-11.

[21] Antonelli, M., Reinke, A., Bakas, S., Farahani, K., Kopp-Schneider, A., Landman, B.A., Litjens, G., Menze, B., Ronneberger, O., Summers, R.M. and van Ginneken, B., 2022. The medical segmentation decathlon. *Nature communications*, 13(1), p.4128.

[22] TorchVision, Mask R-CNN. Available online: https://pytorch.org/vision/main/models/mask_rcnn.html (accessed on 31 July 2022)

[23] TorchVision, U-Net for Brain MRI. Available online: https://pytorch.org/hub/mateuszbuda_brain-segmentation-pytorch_unet/ (accessed on 31 July 2022).

[24] Wall, C., Zhang, L., Yu, Y. and Mistry, K., 2021, July. Deep recurrent neural networks with attention mechanisms for respiratory anomaly classification. In *IJCNN* (pp. 1-8). IEEE.

[25] Shen, Y., Zhang, L. and Shao, L. Semi-Supervised Vision-Language Mapping via Variational Learning. 2017. *IEEE International Conference on Robotics and Automation*, May 29 - June 3, 2017, Marina Bay Sands Convention Centre, Singapore.

[26] Tan, T.Y., Zhang, L., Lim, C.P., Fielding, B., Yu, Y. and Anderson, E., 2019. Evolving ensemble models for image segmentation using enhanced particle swarm optimization. *IEEE Access*, *7*, pp.34004-34019.

[27] Dasari, P., Zhang, L., Yu, Y., Huang, H. and Gao, R., 2022, July. Human Action Recognition Using Hybrid Deep Evolving Neural Networks. In *2022 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-8). IEEE.

[28] Yu, Y., Chen, X., Zhang, L., Gao, R. and Gao, H., 2020. Neural graph for personalized tag recommendation. *IEEE Intelligent Systems*, *37*(1), pp.51-59.

[29] Zhang, L., Lim, C.P., Yu, Y. and Jiang, M., 2022. Sound classification using evolving ensemble models and Particle Swarm Optimization. *Applied Soft Computing*, *116*, p.108322.

[30] Zhang, L., Lim, C.P. and Yu, Y., 2021. Intelligent human action recognition using an ensemble model of evolving deep networks with swarm-based optimization. *Knowledge-Based Systems*, *220*, p.106918.

[31] Mistry, K., Zhang, L., Neoh, S.C., Lim, C.P. and Fielding, B., 2016. A micro-GA embedded PSO feature selection approach to intelligent facial emotion recognition. *IEEE transactions on cybernetics*, *47*(6), pp.1496-1509.

[32] Tan, T.Y., Zhang, L. and Lim, C.P., 2019. Intelligent skin cancer diagnosis using improved particle swarm optimization and deep learning models. *Applied Soft Computing*, *84*, p.105725.

[33] L. Zhang, K. Mistry, S.C. Neoh. and C.P. Lim. Intelligent facial emotion recognition using moth-firefly optimization, *Knowledge-Based Systems*. Volume 111, Nov. 2016, 248–267.

[34] B. Fielding, T. Lawrence and L. Zhang. Evolving and Ensembling Deep CNN Architectures for Image Classification, In *Proceedings of International Joint Conference on Neural Networks (IJCNN)*. 2019.

[35] P. Kinghorn, L. Zhang and L. Shao. A Hierarchical and Regional Deep Learning Architecture for Image Description Generation, *Pattern Recognition Letters*. 2019.

[36] Zhang, L., Srisukkham, W., Neoh, S.C., Lim, C.P. and Pandit, D., 2018. Classifier ensemble reduction using a modified firefly algorithm: An empirical evaluation. *Expert Systems with Applications*, *93*, pp.395-422.

[37] Kinghorn, P., Zhang, L. and Shao, L., 2017, May. Deep learning based image description generation. In *2017 International Joint Conference on Neural Networks (IJCNN)* (pp. 919-926). IEEE.

[38] B. Fielding and L. Zhang. 2020. Evolving deep DenseBlock architecture ensembles for image classification. *Electronics*, *9*(11), p.1880.

[39] Lawrence, T. and Zhang, L., 2019. IoTNet: An efficient and accurate convolutional neural network for IoT devices. *Sensors*, *19*(24), p.5541.

[40] T. Tan, L. Zhang, S.C. Neoh, and C.P. Lim, C.P. Intelligent Skin Cancer Detection Using Enhanced Particle Swarm Optimization, *Knowledge-Based Systems*. 2018.

[41] Wall, C., Zhang, L., Yu, Y., Kumar, A. and Gao, R., 2022. A deep ensemble neural network with attention mechanisms for lung abnormality classification using audio inputs. *Sensors*, *22*(15), p.5566.

[42] Y. Zhang, L. Zhang, S.C. Neoh, K. Mistry and A. Hossain. Intelligent affect regression for bodily expressions using hybrid particle swarm optimization and adaptive ensembles, *Expert Systems with Applications*, 42 (22). pp. 8678-8697. 2015.

[43] L. Zhang, K. Mistry, C.P. Lim and S.C. Neoh. Feature selection using firefly optimization for classification and regression models, *Decision Support Systems*. 106 (2018) 64–85.

[44] Xie, H., Zhang, L., Lim, C.P., Yu, Y. and Liu, H., 2021. Feature selection using enhanced particle swarm optimisation for classification models. *Sensors*, *21*(5), p.1816.

[45] D. Pandit, L. Zhang, S. Chattopadhyay, C.P. Lim, and C. Liu. A Scattering and Repulsive Swarm Intelligence Algorithm for Solving Global Optimization Problems, *Knowledge-Based Systems*. 2018.

[46] S.C. Neoh, L. Zhang, K. Mistry, M.A. Hossain, C.P. Lim, N. Aslam and P. Kinghorn. Intelligent Facial Emotion Recognition Using a Layered Encoding Cascade Optimization Model, *Applied Soft Computing*. 2015.

[47] Y. Zhang, L. Zhang and M.A. Hossain. Adaptive 3D facial action intensity estimation and emotion recognition, *Expert Systems with Applications*, 42 (2015) 1446-1464.

[48] Lawrence, T., Zhang, L., Rogage, K. and Lim, C.P., 2021. Evolving Deep Architecture Generation with Residual Connections for Image Classification Using Particle Swarm Optimization. *Sensors*, *21*(23), p.7936.

[49] Lawrence, T., Zhang, L., Lim, C.P. and Phillips, E.J., 2021. Particle swarm optimization for automatically evolving convolutional neural networks for image classification. *IEEE Access*, *9*, pp.14369-14386.

[50] P. Kinghorn, L. Zhang and L. Shao. A region-based image caption generator with refined descriptions, *Neurocomputing*. 272 (2018) 416-424.

[51] B. Fielding, T. Lawrence and L. Zhang. Evolving and Ensembling Deep CNN Architectures for Image Classification, In *IJCNN*. 2019.

[52] B. Fielding and L. Zhang. Evolving Image Classification Architectures with Enhanced Particle Swarm Optimisation, *IEEE Access*, 6. pp. 68560-68575. 2018.

[53] Gao, R., Hu, S., Yan, L., Zhang, L., Ruan, H., Yu, Y. and Ye, Z., 2023. High-order deep infomax-guided deformable transformer network for efficient lane detection. *Signal, Image and Video Processing*, pp.1-8.

[54] H. Xie, L. Zhang, C.P. Lim, Y. Yu, C. Liu, H. Liu and J. Walters. Improving K-means clustering with enhanced Firefly Algorithms, *Applied Soft Computing*, 84, p.105763. 2019.

[55] Sowan, B., Eshtay, M., Dahal, K., Qattous, H. and Zhang, L., 2023. Hybrid PSO feature selection-based association classification approach for breast cancer detection. *Neural Computing and Applications*, *35*(7), pp.5291-5317.

[56] Qazani, M.R.C., Asadi, H., Zhang, L., Tabarsinezhad, F., Mohamed, S., Lim, C.P. and Nahavandi, S., 2022. A New Prepositioning Technique of a Motion Simulator Platform Using Nonlinear Model Predictive Control and Recurrent Neural Network. *IEEE Transactions on Intelligent Transportation Systems*, *23*(12), pp.23268-23277.

[57] Chen, W., Zhang, L. and Jiang, M., 2022, July. Failure Mode Identification of Elastomer for Well Completion Systems using Mask R-CNN. In *IJCNN*. (pp. 1-8). IEEE.

[58] Huang, H., Wei, J., Zhang, L., Wang, B. and Wang, S., 2022. A Coarse Alignment Method Based on Vector Observation and Truncated Vectorized κ-matrix for Underwater Vehicle. *IEEE Transactions on Vehicular Technology*.

[59] Lu, W., Zhao, D., Premebida, C., Zhang, L., Zhao, W. and Tian, D., 2023. Improving 3d vulnerable road user detection with point augmentation. *IEEE Transactions on Intelligent Vehicles*.

[60] Zhang, L., Lim, C.P. and Liu, C., 2023. Enhanced Bare-Bones Particle Swarm Optimization based Evolving Deep Neural Networks. *Expert Systems with Applications*, p.120642.

[61] Xie, H., Zhang, L. and Lim, C.P., 2020. Evolving CNN-LSTM models for time series prediction using enhanced grey wolf optimizer. *IEEE Access*, *8*, pp.161519-161541.

[62] Slade, S., Zhang, L., Huang, H., Asadi, H., Lim, C.P., Yu, Y., Zhao, D., Lin, H., and Gao, R., (In Press). Neural Inference Search for Multiloss Segmentation Models. *IEEE Transactions on Neural Networks and Learning Systems*.