Discriminative Image Enhancement for Robust Cascaded Segmentation of CT Images

Boonnatee Sakboonyarat[1] and Pinyo Taeprasartsit[2]

ABSTRACT: This work proposes to improve robustness of a cascaded segmentation neural network by adding discriminative image enhancement to its workflow. Unlike prior work, this image enhancement can also be applied as data augmentation and easily adapted for existing models. Its generalization can improve accuracy across multiple segmentation tasks and datasets. The method first localizes a target organ in a 2D fashion to obtain a tight neighborhood of the organ in each slice. Next, the method computes an HU histogram of a region combined from multiple 2D neighborhoods. This allows the method to adaptively handle HU-range difference among images. Then, HUs are nonlinearly stretched through a parameterized mapping function providing discriminative features for the neural network. Varying the function parameters creates different intensity distributions of the target region. This effectively enhances and augments image data at the same time. The HU-reassigned region is then fed to a segmentation model for training. Our experiments and ablation analysis on liver and kidney segmentation showed that even a simple cascaded 2D U-Net model with limited original training data could deliver competitive performance in a variety of datasets. Overall, contributions of this work include adaptive image enhancement and data augmentation that are specifically designed for CT image segmentation and cascaded networks. The method was shown to be generalizable and effective in improving robustness of existing networks in a way that enables a simple model to both save computing resources and be highly accurate.

Keywords: Automated Segmentation Method, CT Image, Data Augmentation, Image Enhancement, Image Segmentation, Neural Networks

DOI: 10.37936/ecti-cit.2021152.240112

Article history: received March 18, 2020; revised May 13, 2020; accepted June 29, 2020; available online April 20, 2021

1. INTRODUCTION

Analysis of 3D medical images has become a common tool in many clinical practices, including disease screening, diagnosis, and treatment. Tumor and polycyst analyses in the liver and kidney are among the most popular subjects in recent years. Also, total kidney and liver volume has been found to be a major risk factor for malnutrition in some patients. Volumetric analysis of the liver and kidney through computed tomography (CT) data is a convenient tool in a case study [1]. These analyses usually rely on region segmentation of involved organs and lesions. Although there have been some significant improvements in segmentation accuracy in the past few years, researchers are still looking for batter methods that are accurate across multiple tasks and datasets [2].

In recent years, many neural networks for medical image segmentation have been proposed to improve accuracy and efficiency. There are several noteworthy achievements for livers, kidneys, and their tumor segmentation (e.g. H-DenseUNet and a fined-tuned 3D Residual U-Net) [3][4]. This work, however, tackles the segmentation problem in another aspect. Instead of trying to find a better neural network architecture, we explored the challenge of the low contrast issue common to most medical image segmentation problems. To this end, we introduce a novel discriminative image enhancement and data augmentation, which can be easily employed by most cascaded image-segmentation networks.

[1,2]The authors are with Department of Computing, Faculty of Science, Silpakorn University, Nakhon Pathom, Thailand., E-mail: boonnatee.sak@mwit.ac.th and taeprasartsit_p@silpakorn.edu

[2]Corresponding author: taeprasartsit_p@silpakorn.edu

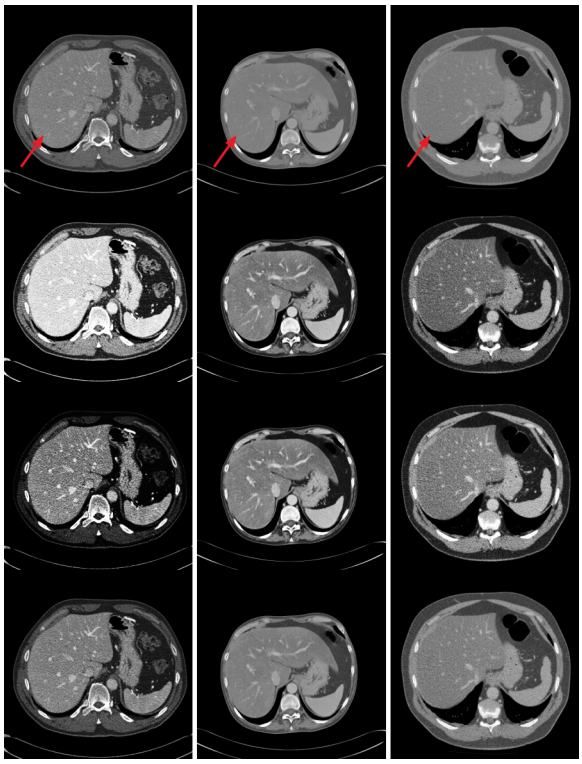

Fig.1: *Examples of the proposed discriminative image enhancement and data augmentation for liver segmentation. The method both boosts contrast and provides a variety of images for data augmentation. From left to right: example images from the SLiver07, 3DIRCADb, and LiTS datasets, respectively [7]–[9]. Top row: images with HU window [-450, +550] (red arrows point at the livers). Second row: images when enhanced by a method that assumes a fixed HU range of the liver. It is obvious that the liver from SLiver07 looks much different from others due to its different HU range. This is not a major issue by itself, as long as a model and training process can address such HU differences. However, that can be challenging in many contexts. Third row: images after enhancement by the proposed adaptive method with the default parameters and with focus on the liver. Bottom row: similar to the third row, but are enhanced by different parameters. This produces additional variation of enhanced images for training data.*

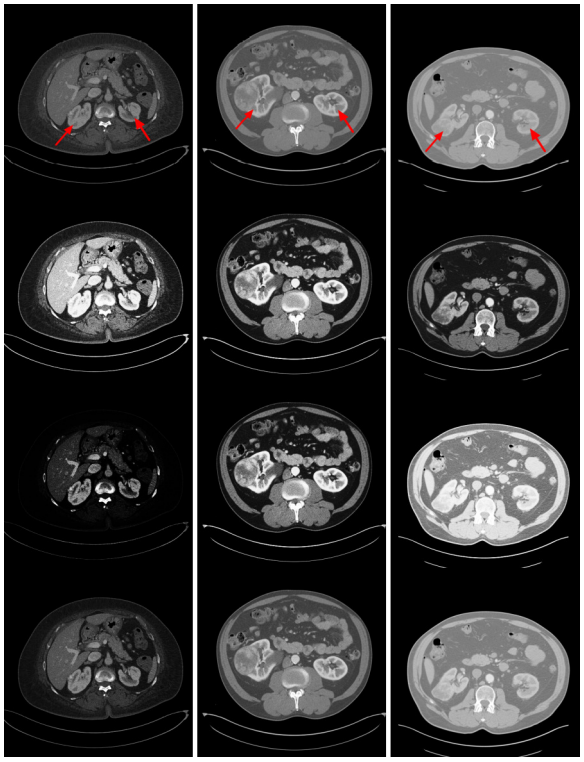

Fig.2: *Examples of the proposed adaptive image enhancement and data augmentation for kidney segmentation. The arrangement of rows and columns are the same as Fig.1. The red arrows point to the kidneys. All images in this figure are from the KiTS dataset [17]. In the third and fourth rows, the proposed method does not make enhanced images from the three cases look similar, but contrast between the kidneys and their neighborhood becomes pronounced in different ways.*

In our ablation analysis and cross validation, we show that accuracy and robustness gained from the proposed discriminative image enhancement and data augmentation is particularly significant when applied to a cascaded segmentation network. Without the proposed image enhancement, variances of accuracy in liver and kidney segmentation were much higher, and there were several cases that were poorly segmented.

1.1 Study Overview

The low contrast issue can usually be ameliorated by image enhancement, but there may be considerable Housfield-Unit (HU) variation in a target region among patient cases. It is hard to specify a fixed, tight HU range on which an image enhancement will focus and achieve robustness in region segmentation throughout a large dataset. If we, however, know the HU profile of a target region of a patient case, it is possible to select better image-enhancement parameters resulting in improvement of segmentation accuracy for that patient case. In addition, if we employ different sets of image-enhancement parameters, a variety of images can be generated and utilized as augmented data for neural network training. This has potential to improve segmentation accuracy in many tasks.

For the sake of concreteness, Fig.1 and Fig.2 are provided to illustrate one of the main ideas of the proposed image enhancement and data augmentation techniques. The figures depict how HU reas-

signment can enhance the image and ameliorate the low-contrast issue, and how we can adapt it to augment a dataset. Furthermore, the underlying HU-reassignment function is partially based on a sigmoid function. It can provide discriminative advantages for region segmentation as well.

To achieve our goal, the method first needs to focus on a target region. It adaptively creates new data from HU-value profiles in the target neighborhood. The method employs a RetinaNet-based model for region localization to obtain a bounding box [5][6]. Then, a histogram of HU values in the bounding box is computed and redistribution of the HU values is performed to enhance and augment image data.

It is important to note that region localization becomes a typical step before actual segmentation in a variety of tasks, especially when a target region in a cross-sectional image is relatively small. This is especially true for the spine and tumors [10]–[14]. This segmentation workflow is referred to as a cascaded segmentation or a 'hard-attention' model. Many other methods employ a soft-attention mechanism that allows a segmentation model to focus better on the neighborhood of a target region to locally enhance an input image [14]–[16].

The major differences of our work from other cascaded models are (1) our region localization method is a 2D-3D hybrid which enables it to accumulate HU data more relevant to a target region across slices, (2) our image enhancement is not a common histogram equalization; it specifically aims at providing discriminative features for segmentation, (3) its parameterizable nature allows it to work as a data augmentation technique; more interestingly, our data augmentation is executed in the middle of a segmentation workflow, not at the beginning, and (4) its modular design is 'pluggable' to existing cascaded models.

1.2 Contributions of the Proposed Method

Considering common preprocessing steps and data-augmentation techniques, we see that neural network models for medical image segmentation in CT/MRI data rarely make use of characteristics of target regions or the nature of the CT/MRI image itself, at least not explicitly. Specifically, the image is generally similar to a grayscale image. A target region, such as the liver or kidney, has a typical HU range. Yet, the range may be significantly different for some cases. If we can utilize these characteristics beyond the common preprocessing steps and data-augmentation techniques, a seemingly less powerful model can probably deliver comparable accuracy with much less inference time and computing resources. Also, segmentation accuracy of a state-of-the-art technique can be further improved.

This work proposes discriminative image enhancement and data augmentation that can be applied in general CT image segmentation tasks. This is achieved via an effort to exploit knowledge of actual HU values of a target region in an input image, rather than relying on a fixed HU range as is done in most other work. For example, a common HU range of the liver is 45 to 65HU, while that of the kidney is 20 to 40HU. However, these ranges may not hold true due to various conditions such as use of a contrast agent and lesion involvement. Therefore, it is important to find good approximations for each case, since HU values may greatly vary among patients' cases.

Through that effort, this work does not need to set a tight HU window for a specific organ. Instead, we set HU windows to [-450, 550] for both liver and kidney segmentation in our experiments. The method, then, adaptively enhances and normalizes data according to HU values in the neighborhood of a target organ located by a detector.

It is worth noting that the HU window in our experiments is mainly utilized only for region localization, and it is not important during a segmentation step. This generalized method offers improved robustness in segmentation and enables a basic U-Net model to deliver competitive accuracy which we have demonstrated in liver and kidney segmentation.

The design of the method can also be readily utilized by other cascaded models where a neighborhood of a target region is extracted by the models. In detail, once the neighborhood is located, we can feed it to the proposed method to generate discriminative features for either prediction or data augmentation. Fig.3 illustrates how our work can collaborate with other cascaded models. We expect that the proposed method will provide similar benefits in other tasks and that it can be generalized to a greater extent. Evaluation of its impacts on other models, however, is beyond the scope of this work.

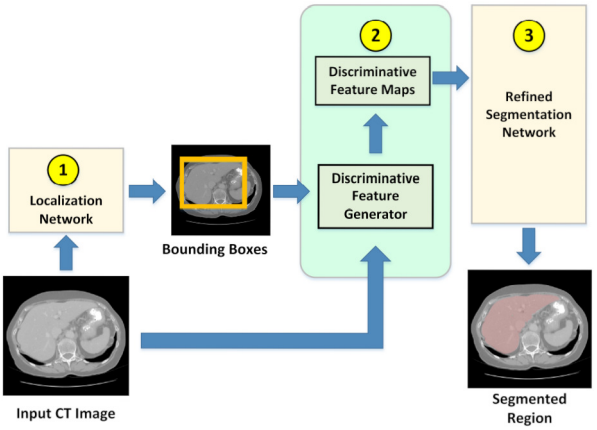

Fig.3: *Block diagram illustrating how the proposed method can work with other cascaded models. In a typical scenario, Block 1 is a localization network. The block creates bounding boxes for its cascaded mechanism. Block 2 is the proposed method for generation of additional discriminative features and data augmentation.*

The rest of this article is organized as follows: Section 2 discusses studies related to CT-image segmentation, especially those involving liver and kidney segmentation. Section 3 explains the configuration of our neural network models and the proposed image enhancement and data augmentation. Section 4 presents the experiment setups and results of our method. Finally, Section 5 concludes the article.

2. RELATED WORK

Many recent studies of CT-image segmentation focus on proposing neural network models. The U-Net model is arguably one of the most successful models. It has inspired many other competent segmentation models. There are a few recent studies that do not rely on a neural network, but still deliver a competitive accuracy, employing techniques such as a variation-based method and graph-cut algorithms [18][19]. Many research efforts are conducted toward multi-task segmentation or methods generalizable to perform segmentation for other organs or lesions [20]–[25].

Since this work aims at providing a generalizable method for CT-image enhancement and data augmentation applicable to neural network training and inference, we mainly cover related work focusing on cascaded/attention-based segmentation, highly competent models, preprocessing, and data augmentation.

2.1 Cascaded Segmentation and Attention Mechanism

Recently, there have been several studies in cascaded segmentation in medical image processing. Cascaded segmentation provides an attention mechanism for a model to focus on the neighborhood of a target region. Christ et al. showed that a cascaded U-Net was significantly more accurate than a basic U-Net in liver and tumor segmentation [26][27]. Yin et al. utilized a similar approach for kidney segmentation [28], while Myronenko and Hatamizadeh used boundary-aware networks to create an attention gate for kidneys and kidney tumor segmentation [15]. Jiang et al. proposed an AHCNet, which successively employs cascaded segmentation to segment both livers and liver tumors. The last part of AHCNet applies the segmented liver to enhance contrast between livers and associated tumors.

2.2 Highly Competent Models

There are several models having high accuracy on medical image segmentation, and some of them are highly competitive in segmentation challenges. In the LiTS challenge, H-DenseUNet was shown to be significantly accurate in both liver and liver-tumor segmentation, especially in the 3DIRCADb dataset [3]. In fact, H-DenseUNet is also a cascaded segmentation model. However, its main contribution is a hybrid feature-fusion layer where both 2D and 3D features are jointly optimized for final segmentation. In the KiTS challenge, Isensee and Maier-Hein, the top performer, employed a Residual 3D U-Net [4], along with extensive data augmentation, and marginally outperformed an ensembled model in a composite Dice score of kidney and kidney-tumor segmentation [29].

Interestingly, these highly competent models are all inspired by U-Net [3][4][29]. However, Xia et al. proposed a deep adversarial network based on DeepLab-v3 for segmentation tasks and their own Pix2Pix to complete their adversarial model [30]–[32]. It was shown in experiments that their adversarial model outperformed models competing in the LiTS challenge and demonstrated improvements over the baseline DeepLab-v3 in many performance metrics.

2.3 Preprocessing and Data Augmentation

Common preprocessing steps for medical segmentation tasks include resampling, truncating HU values, image equalization, normalizing HU values by a zero-mean, and using unit-variance approaches. Data augmentation in 3D medical segmentation, nonetheless, appears to be generic for typical image data and not specially designed for 3D medical segmentation tasks. For example, Isensee and Maier-Hein employed scaling, rotations, brightness, contrast, gamma, and Gaussian noise augmentations [4].

The typical HU range for each target region is also different. H-DenseUNet and AHCNet used an HU range of $[-200, 250]$ for liver and liver-tumor segmentation [3][14], while Isensee and Maier-Hein employed an HU range of $[-79, 304]$ for kidney and kidney-tumor segmentation [4]. Wang et al. set an HU range for a liver tumor to $[-110, 190]$ and to $[-100, 200]$ for pancreas segmentation in their Nested Dilation Network (NDN) [33].

3. METHODOLOGY

3.1 Overview

To perform discriminative image enhancement, we first acquire HU values of a target region and its vicinity in a bounding box. This is done by region localization based on a neural network. We create an HU histogram and assume that the HU with maximum frequency in the bounding box of the target region represents typical HU values of the target. We assume further that a low contrast problem can be ameliorated if contrast is mainly stretched around the maximum-frequency HU. Once the image is enhanced, a neural network model for segmentation can employ it for training. During prediction, the image is enhanced and segmented in the same way. Since contrast can be stretched with a different parameter, utilizing multiple parameters during training effectively augments the data. During prediction, how-

ever, only the default image enhancement parameter is employed.

3.2 Region Localization

Region localization is done in the context of object detection. For the sake of concreteness, assume that we are to localize the liver. We employ a RetinaNet model implemented by Fizyr B.V. as a base model [6]. Since the model localizes the liver in a 2D context, we need additional steps to localize the liver and accumulate needed HU data.

The method first finds the largest 2D bounding box detected during liver localization. Then it keeps incorporating other consecutive slices with detected liver regions. At this point, the method has the largest 2D bounding box and related slices. Next, it expands the related slices in the superior direction (to the patient's head) and inferior direction (to the patient's feet) by 10 slices in each direction. This step is needed since the top and bottom parts of the liver are small and mis-detection often occurs.

Since the segmentation model assumes a cross-section size of 384 voxels $\times$ 384 voxels, the final 3D bounding box has the same cross-section size. In detail, the method expands the largest detected bounding box from its center to the size of 384×384. From the expanded cross-section, it incorporates all the related slices to build a 3D bounding box of the liver. It is important to note that 2D detection may not be reliable in some slices, and that there may be multiple liver regions in a slice, but the aforementioned steps only need to know the slices involved. The cross-section expansion corrects regression errors of bounding-box scope.

Regarding data preprocessing for this step, the HU window was set to [-450, 550] as mentioned in Section 1.2. HU values in the window are linearly normalized to the range [0, 1]. This work later refers to this normalization procedure as NMIM.

To generalize the idea for other organs, we demonstrate the concept kidney localization where there are typically 2 kidneys per case study. In this scenario, the localization model is trained to detect each kidney as a different object. Consequently, there are usually two 2D bounding boxes of kidneys in each slice. The method then finds the two largest 2D bounding boxes that do not overlap and incorporates related slices for each kidney.

Since the kidneys' cross-sections are smaller than the liver, the size of their expanded cross-section is set to 192×192. There are a few cases where largest bounding box exceeds the expanded size. In such a case, the method shrinks the box to fit the 'expanded' size, i.e. 384×384 for the liver and 192×192 for the kidneys. Fig.4 depicts the relationship between a bounding box directly obtained from a RetinaNet model and its corresponding expanded box.

3.3 Discriminative Image Enhancement

As mentioned earlier, the proposed image enhancement needs to know the HU value that can represent the target organ well in order to focus on the value during intensity stretching. Although the localization step creates a 3D bounding box for segmentation, 2D bounding boxes lying inside the 3D one play a key role in this adaptive image enhancement.

First of all, those 2D bounding boxes fit target organs better than the 3D one, since they adjust to actual presence of a target region in each slice. Therefore, when an HU histogram for a target region is to be created, voxels inside the 2D ones provides more accurate data. In other words, our discriminative HU mapping function is built upon statistics accumulated inside 2D bounding boxes obtained from region localization, not the expanded 3D bounding box defining the region scope for segmentation.

Specifically, HU values from all voxels inside these 2D bounding boxes are accumulated to create the HU histogram. Then, the HU value whose frequency is the maximum is selected as the center for contrast stretching. The process of creating the HU histogram is illustrated in Fig.5.

Fig.6 and Fig.7 show examples of HU histograms for the liver and kidneys. These histograms are created from HU statistics accumulated from 2D bounding boxes directly obtained by a RetinaNet model. It is worth noting that these 2D bounding boxes may not fully cover the target organs, but in those cases, the missing statistical values are relatively small since they come from the upper and lower edges of the organs. Also, the HUs in the histograms are from the input image, not values which have undergone truncation by an HU window. The maximum-frequency HU in accumulated 2D bounding boxes will be referred to as the center HU (μ), which is employed in the proposed parameterized discriminative HU mapping function.

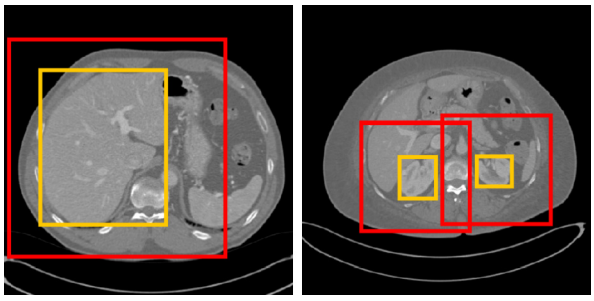

Fig.4: *Region localization and corresponding cropped regions fed to a segmentation model. Yellow (smaller boxes): largest 2D bounding boxes detected by RetinaNet. Red (larger boxes): corresponding cropped regions (corresponding expanded boxes). Left: results for liver detection. Right: results for kidney detection.*

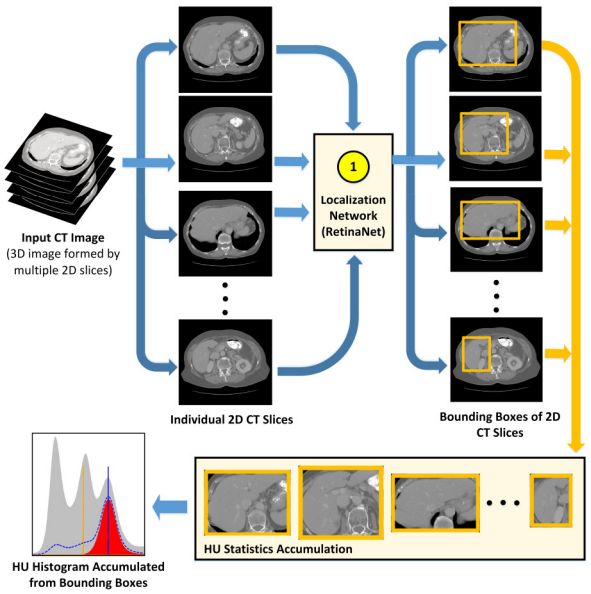

Fig.5: *Procedure for creating an HU histogram. This histogram is employed for discriminative image enhancement. Bounding boxes from involved slices detected by RetinaNet connected with the slice with the largest box are employed for HU statistics accumulation. These boxes may be significantly different in size, as displayed by the orange bounding boxes.*

Now that we have explained how the center HU for nonlinear contrast stretching is calculated, we next discuss how HU reassignment is computed for image enhancement. The rationale of our HU reassignment is based on observations that HUs of a target organ usually cluster around the center value μ. In addition, HUs of non-target voxels in detected bounding boxes are relatively far away from the center value. These HUs are represented by gray regions under the blue dotted line in Fig.6 and Fig.7.

Since we want to make the HU difference between the target and non-target voxels more pronounced to discriminate between the two groups, we opt to stretch the HUs around the center value μ. Because a sigmoid function can provide smooth HU stretching, the proposed HU reassignment function is based on a sigmoid function. In contrast, typical histogram equalization may be significantly affected by voxels irrelevant to a target organ since they can make up a large portion of the voxels inside bounding boxes detected by RetinaNet (Fig.6). If typical histogram equalization is applied, these irrelevant voxels may distort the probability distribution in a way such that it barely helps discriminate between a target and irrelevant voxels.

Our method reassigns HU values by using a function fx^*, which is described by using a the following equations:

$$fx'(x,\mu,w) = x - \mu + w/2 \tag{1}$$

$$c(x,\mu,\sigma) = \frac{1}{1+e^{-\frac{-(x-\mu)}{\sigma}}} \tag{2}$$

$$p(x,w,\sigma) = c\left(\frac{fx'(x,\mu,w)}{fx'(x-\frac{w}{2},\mu,w)}, 0.5, \sigma\right) \tag{3}$$

$$fx^*(x,\mu,w,\sigma) = \frac{p(x,w,\sigma) - c(0,0.5,\sigma)}{c(1,0.5,\sigma) - c(0,0.5,\sigma)} \tag{4}$$

x is an HU value, μ is the center HU, w is the width of the HU window, and σ is HU stretching factor. The first equation is a linear mapping of x that prevents a negative value of the first parameter of c in (3) and (4). c is a function based on sigmoid. Function fx^* employs a scaling method to make the maximum and minimum be one and zero respectively.

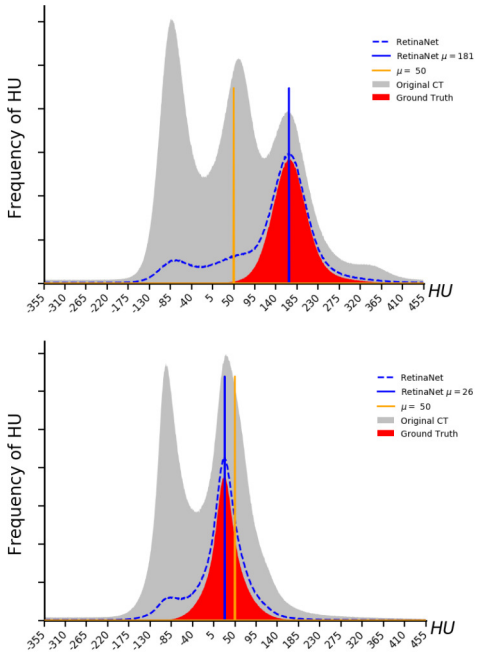

Fig.6: *HU histograms of liver ground truth (red), accumulated 2D bounding boxes detected by RetinaNet (blue dotted line), and original CT slices containing the liver region (gray). Top: histograms of the SLiver07 case in Fig.1. Bottom: histograms of the LiTS case in Fig.1. The horizontal axis represents HU values, while the vertical one displays frequency of voxels having corresponding HUs. The HU values of the SLiver07 and LiTS cases significantly differ. The blue line indicates the HU with maximum frequency within the accumulated 2D bounding boxes. This HU is the same, or almost the same, as the maximum-frequency HU of the ground truth. The orange line is set at HU = 50, which is close to a mean value of the liver in unenhanced data [34].*

In our experiment's settings, the width of the HU window w is a constant. Although it seems that we change from depending on a specific HU range to depending on the window size, w can be relatively large and cover an HU range of many organs and lesions.

In our experiments, w=1,000 and the default value of σ=0.07. While the default σ may not be optimal in some cases, we empirically found that it significantly helped improve robustness of the method in most cases. Fig.8 and Fig.9 illustrate how the proposed method reassigns HU values and the importance of a suitable center HU.

The center HU μ obtained through accumulated 2D bounding boxes detected by RetinaNet significantly improves contrast of a target region. On the other hand, using a fixed center HU may result in contrast being stretched on an unrelated HU range of a target region. Also, using the maximum-frequency HU in the original CT slice (gray regions in Fig.6 and Fig.7) may suffer from the same issue. Once parameters for HU reassignment are determined, HU reassignment is executed on an input image as depicted in Fig.10. Outputs from a localization network (orange rectangles) are applied to crop neighborhoods of a target. These neighborhoods (red rectangles) have the same size (384×384 pixels in our settings), and they undergo HU reassignment.

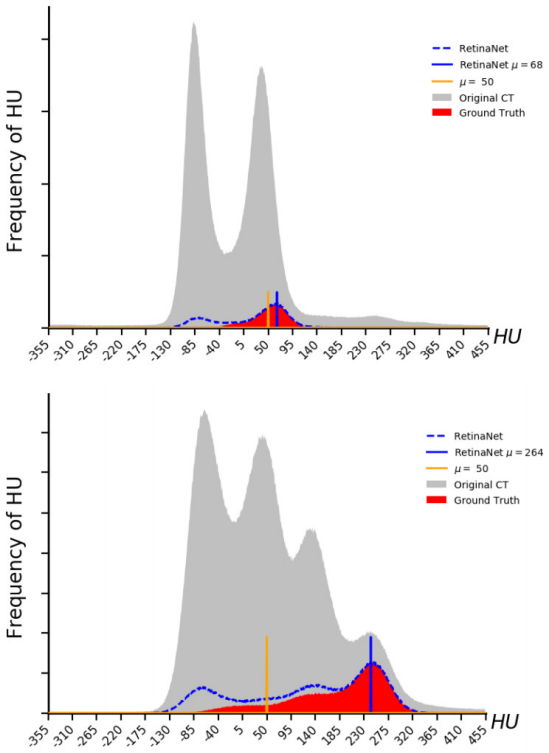

Fig.7: *HU histograms for a kidney-segmentation task from two cases in the KiTS dataset. Since the kidneys are much smaller than the liver, the HU frequencies appear much smaller when compared to the whole CT slices. Similar to the liver data, HU values from two cases may be greatly different, but the RetinaNet bounding boxes can find the center HUs as intended.*

The adaptability of the method is closely connected to the value of μ specific to each input image. That was illustrated earlier in Fig.1, Fig.2, Fig.6, and Fig.7. Since function $c(x, \mu, \sigma)$ is partially based on a sigmoid function, it provides discriminative advantages for segmentation of regions whose HUs are close to μ. It helps enhance contrast and normalize an image at the same time. The significance of parameter σ related to data augmentation will be discussed in the next section.

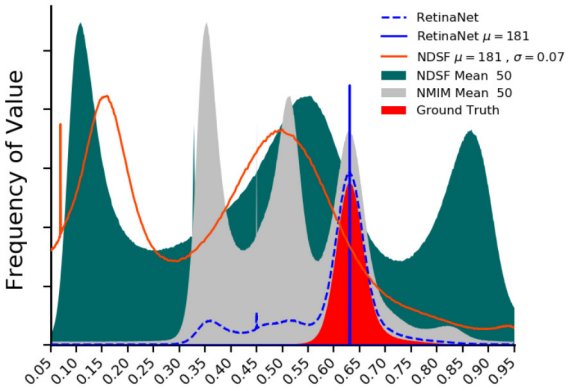

Fig.8: *Histograms of normalized HUs obtained from different methods of the SLiver07 case shown earlier in Fig.6. Histograms from Fig.6 are shown here only for comparisons between normalized and original histograms. The horizontal axis represents normalized HU in range [0, 1]. The vertical axis denotes frequency of HU values. Except for the NDSF histograms, which are based on function x, fx^*, value normalization is done through the HU windows [-450, 550] as their centers are fixed at HU = 50. The orange line is the histogram computed for the default NDSF model (σ=0.07 and the center HU calculated from accumulated 2D bounding boxes). The center HU for the NDSF model is shifted to 0.50. When the center of HU stretching is not close to the maximum-frequency HU, the HU reassignment does not focus on the target region (green histogram). In this example, the target region corresponds to the rightmost peak of the original CT slices (gray histogram), but the green histogram focuses on the middle one instead.*

3.4 Data Augmentation

When using the HU reassignment function fx^*, we can adjust parameter σ to change image-enhancement outcomes. Since the center HU mostly depends on the HU in the target organ, adjusting σ will create images with different contrast levels between the organ and its neighborhood. We have made the assumption that utilizing multiple σ values during model training will help the model learn features related to the organ boundary and improve robustness of the resulting trained model.

Fig.11 delineates how x, fx^*, and σ are related. By decreasing the value of σ, the graph appears

steeper since reassigned values are distributed to a smaller range close to μ, thereby producing more contrast around μ and providing discriminative benefits. Fig.12 depicts impacts of σ on contrast stretching. A larger value of σ has a higher probability of covering the entire HU range of a target region, but it may not focus well on the main body of the target region.

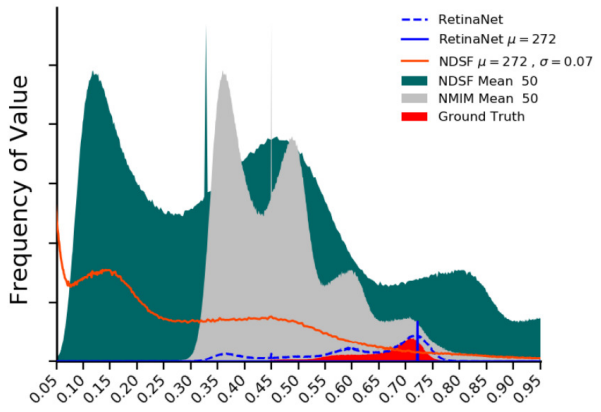

Fig.9: *Histograms of normalized HUs obtained from different methods of the KiTS image shown earlier at the bottom of Fig.7. Similar to the previous figure, the center HU μ helps the method stretch contrast in a relevant region and provides discriminative benefits for kidney segmentation.*

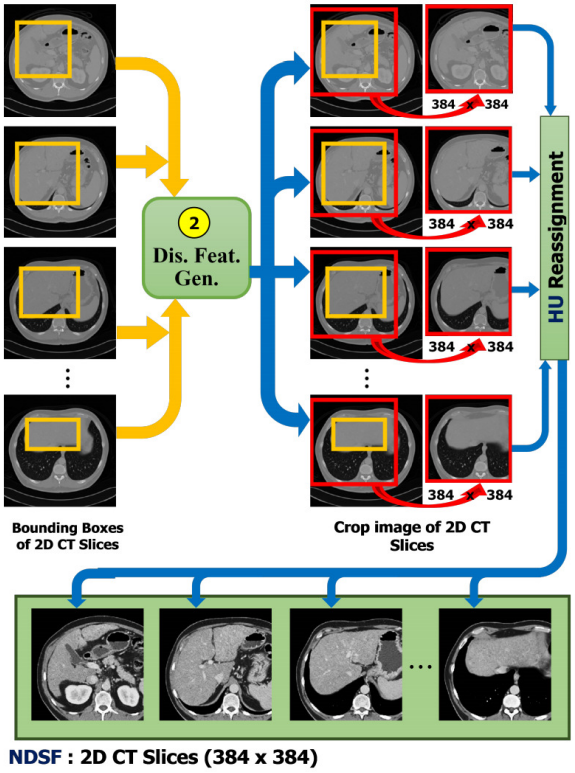

Fig.10: *Illustration of how HU reassignment is applied for a cascade segmentation of the liver.*

In the top diagram of Fig.12, $\sigma = 0.07$ produces a curve covering regions of two gray peaks from the right. This is a suitable choice of σ for providing discriminative features since it significantly helps differentiate between the liver (red) and its neighborhood (gray under dotted blue line). For the bottom diagram, however, $\sigma = 0.07$ produces a curve partially covering regions of three gray peaks, but it should also include the leftmost peak in order to cover the entire target region (red). Specifically, $\sigma = 0.15$ (dark green line) provides better discriminative features in this case and σ should be adaptive. Nonetheless, there are not many such cases in the dataset. We selected $\sigma = 0.07$ as a default value.

We refer to a segmentation model trained by images enhanced solely using the default σ=0.07 as an NDSF (Normalized Discriminative Sigmoid Function) model. Also, we refer to a segmentation model trained with additional σ's as an NDSF+ model. For a segmentation model where training images are normalized by NMIM, we regard it as a baseline model. All three types of models have exactly the same neural network architecture for both detection and segmentation tasks. The only difference between them is the training data utilized for a segmentation model. Their architecture is discussed next.

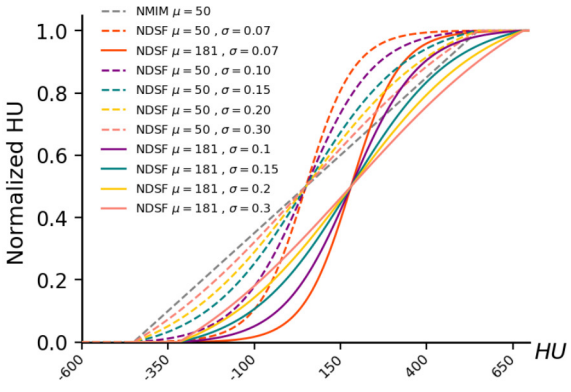

Fig.11: *Diagrams of HU reassignment functions fx^* and NMIM. An input HU (horizontal axis) is mapped to normalized HU range [0, 1] (vertical axis). The dotted lines represent the functions with $\mu = 50$, while the solid lines represent the functions when μ is found by the proposed method in the SLiver07 case. In brief, center HU μ only translates the diagrams, while σ affects how much contrast there will be.*

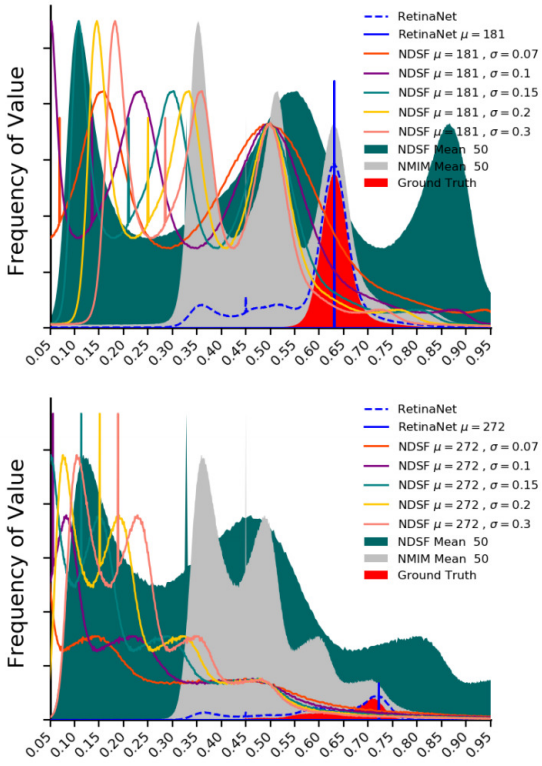

Fig.12: *Relationship of σ and normalized HU histograms of enhanced images.*

3.5 Segmentation Model

Our segmentation model is essentially a basic U-Net proposed by Ronneberger et al. [35]. The main differences are (1) incorporation of dropout and batch normalization [36][37], and (2) different input and output sizes. For liver segmentation, the input and output sizes are 384×384, while they are 192×192 for kidney segmentation. Fig.13 depicts the U-Net architecture employed in this work. Another difference from the original U-Net model is the number of convolutional layers and feature maps for each resolution.

3.6 Cascaded Prediction

Due to the goals of our study, we explicitly separate detection and segmentation tasks. First, explicit separation allows us to study the impact of the proposed method. It excludes side-effects from region localization during ablation study. Second, there is no need to repeat the detection task that provides information about the center HU. Although we can combine both detection and segmentation capabilities in a single model and jointly optimize hyperparameters for both tasks, doing so may make it harder to study the impact of the proposed method.

Consequently, our cascaded prediction has two explicit stages executed. First the method resamples the input image so that its voxel spacing is 0.75 mm $\times$ 0.75 mm $\times$ 0.75 mm. Next, it performs NMIM normalization and employs RetinaNet to find the slice whose detected bounding box is the largest. Finally, it builds up the 3D bounding box to finalize the detection task, while using 2D bounding boxes to accumulate HU values and compute the center HU.

The segmentation task depends on the model type. For a baseline, the same NMIM normalization is employed for segmentation. For the NDSF and NDSF+ models, image enhancement steps with default $\sigma = 0.07$ are executed. Fig.14 illustrates how an NDSF model is applied to segment regions in HU-reassigned 2D CT slices. Other models are processed in the same manner.

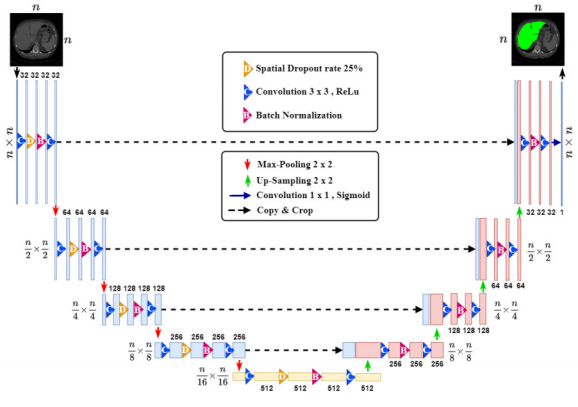

Fig.13: *Segmentation step for an NDSF model.*

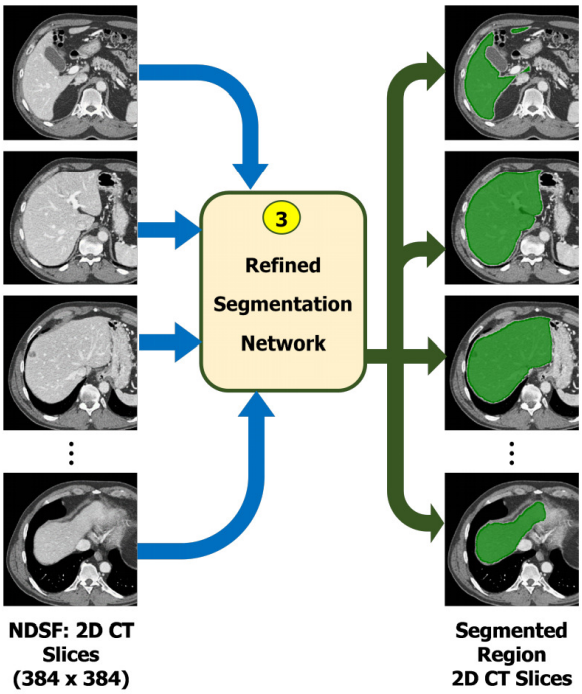

Fig.14: *The U-Net architecture employed in this work.*

4. EXPERIMENTS AND RESULTS

4.1 Experiment Setup

The experiments were conducted on liver and kidney segmentation tasks. For liver segmentation, the LiTS dataset was utilized for both training and testing, while the 3DIRCADb and SLiver07 datasets were employed solely for testing. For kidney segmentation, both the training and testing data was from the KiTS dataset. We merged tumor regions with their corresponding organ into a single region since images in SLiver07 were created as such. Due to the termination of evaluation service of SLiver07, we needed to use SLiver07's 20 training images as test images here.

Robustness of the proposed method was validated in various aspects: (1) accuracy, (2) variance of accuracy between folds and datasets, (3) ablation analysis, and (4) comparison with existing methods. The accuracy indices are Dice similarity, Volume Overlap Error (VOE), Relative Volume Difference (RVD), Average Symmetric Surface Distance (ASSD), Root Mean Square Symmetric Surface Distance (RMSD), and Maximum Symmetric Surface Distance (MSSD) [38]. All Dice scores in this section were 'Dice per case'.

4.2 Data Partitioning for Multi-fold Cross Validation

For both the LiTS and KiTS datasets, we first partitioned data into three groups. For the sake of concreteness, we initially discuss our data partitioning for the LiTS dataset. Data partitioning for the KiTS dataset in our experiments was done in the same manner.

There were 131 samples with ground truth in the LiTS dataset. We randomly partitioned them into three groups G1, G2, and G3 with 45, 45, and 41 samples, respectively. Then, 31 samples in group G1 were randomly picked as test samples when fold F1 was utilized for training and testing. In other words, out of 131 samples in the LiTS dataset, when we work with fold F1, 31 samples were reserved as test samples and they were taken from group G1. Then, there were 100 training samples for fold F1; 14 were from G1, 45 from G2, and 41 from G3.

Similarly, when fold F2 was utilized for training and testing, there were 100 training samples; 45 were from G1, 14 from G2, and 41 from G3. The same was done for F3. Table 1 summarizes the data partitioning. Regarding Datasets 3DIRCADb and SLiver07, they both have 20 samples with ground truth and all of them were employed as test data.

For the KiTS dataset, there were 210 samples with ground truth. We partitioned them into three groups, each with 70 samples. Data in each group was further partitioned for training and testing in the same fashion as LiTS. Out of 70 samples of a group, 42 samples were reserved for test data when its corresponding fold was studied.

Table 1: *This table shows how samples from each group in the LiTS dataset was partitioned for each fold. As shown in Column G1, out of the total 45 samples of G1, 31 were employed as test data for F1 and none for F2 and F3. During training, however, 14 samples from G1 were training data for F1, while all 45 samples were training data for F2 and F3. Groups G2 and G3 underwent a similar procedure.*

		G1	G2	G3	Total
Type	**Samples**	**45**	**45**	**41**	**131**
Test	F1	31	0	0	31
	F2	0	31	0	31
	F3	0	0	31	31
Train	F1	14	45	41	100
	F2	45	14	41	100
	F3	45	45	10	100

These three groups for each dataset were introduced to make robustness validation simpler when fewer training samples were employed. Specifically, we wanted to validate each model's robustness when it was trained with smaller training sets. In our experiments, there were four sizes of training sets: TR025, TR050, TR075, and TR100, denoting 25%, 50%, 75%, and 100% of the total training samples in a dataset. For example, out of 168 training samples in the KiTS dataset, TR025, TR050, TR075, and TR100 had 42, 84, 126, and 168 training samples, respectively.

4.3 Model Training

All models in our experiments were based on RetinaNet and 2D U-Net as discussed in Section 3, and their HU windows were set to [-450, 550] during localization. The localization model was trained for 20 epochs and there were 3 folds. All of these folds used 100% of the training data (TR100). Data augmentation for training RetinaNet included the default random transform implemented by Fizyr B.V. [6].

Regarding image enhancement, models were trained with three configurations: baseline, NDSF, and NDSF+. The baseline model linearly mapped HUs in the window to range [0, 1]. The NDSF reassigned HUs by the proposed image enhancement with $\sigma = 0.07$. NDSF+ reassigned HUs with $\sigma = 0.07, 0.10, 0.15, 0.20$, and 0.30. We trained all of the models with the Nadam optimizer for 150 epochs. The number of training samples fed to each network was exactly the same. For example, data augmentation for an NDSF+ model was performed online with probability 0.20 for each σ.

4.4 Ablation Study

As in the first part, we trained and tested models against the LiTS, 3DIRCADb, and SLiver07 datasets. The results from the test sets are shown in Table 2.

In the LiTS test set, even the baseline model performed well in the Dice metric, but in the surface-distance metrics, its performance fell significantly behind the NDSF+ model in all training sets. Overall, the NDSF+ models slightly outperformed the NDSF ones in Dice scores, but not in surface-distance metrics.

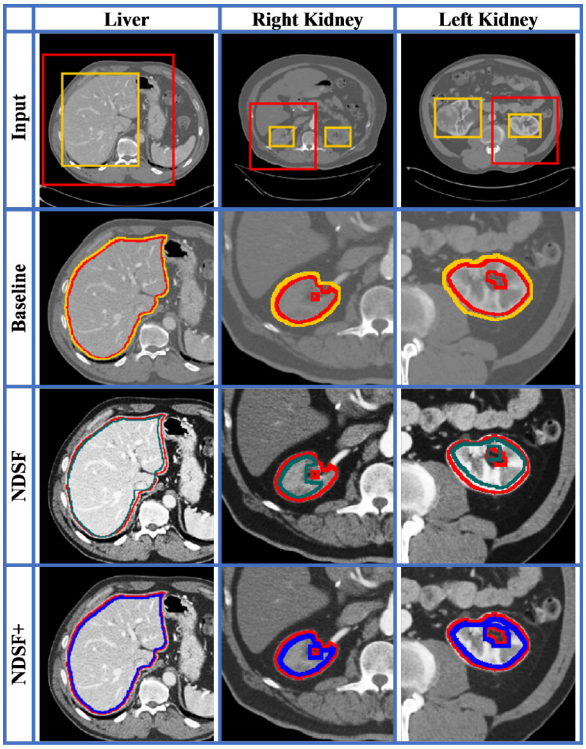

Fig.15: *Example segmentation outputs from the methods. Top: bounding boxes from RetinaNet (light yellow) and expanded bounding boxes for segmentation (red). 2^{nd} – 4^{th} rows: example segmentation outputs compared with the ground truth. Red is the ground-truth boundary, while yellow, blue, and green are the boundaries of outputs from baseline (2^{nd} row), NDSF (3^{rd} row), and NDSF+ (4^{th} row) models, respectively. Left: examples of liver-segmentation outputs. Middle and right: example segmentation outputs of the right and left kidneys. The intensities of images in this figure corresponds to actual intensities seen by the models. For instance, the intensities of the last two rows are outcomes of the proposed adaptive image enhancement with $\sigma = 0.07$.*

NDSF+ models, however, performed more consistently on all datasets (discussed in more detail later). Interestingly, increasing the number of training samples did not improve Dice scores, but accuracy in other metrics was affected. Fig.15 illustrates outputs from the bounding-box detection process and certain segmentation differences from the three methods. The proposed method significantly improved contrast of target regions for both unenhanced and enhanced CT data.

The most interesting finding of the liver-segmentation results is that the NDSF and NDSF+ models were barely affected by the test sets, but the baseline model performed poorly in the SLiver07 test set. Visually, CT images from the SLiver07 dataset looked much different from those in LiTS and 3DIRCADb (Fig.1). The proposed method provided robustness to the trained models and prevented them from performing poorly when the input images significantly differed in appearance.

In a kidney-segmentation task, the NDSF and NDSF+ models significantly outperformed the baseline model in both volume-based and surface-distance metrics (Table 3). Overall, the NDSF+ tended to provide advantages over NDSF in volume-based metrics, which are directly related to the loss function utilized in model training.

We also observed robustness of the proposed method through the standard deviation of Dice scores in the test sets. It is clear from Fig.16 and Fig.17 that NDSF models produced more consistent Dice scores when compared to the baseline scores. Also, Dice scores of NDSF+ models were the most consistent and they outperformed the NDSF scores in every training combination. Interestingly, while increasing the number of training samples hardly affected Dice scores for all model types, the NDSF-based models' Dice scores became less varied for TR100 in the SLiver07 dataset.

Further observations on the minimum of Dice scores reveal that baseline models tended to have poor Dice scores for some cases in both liver and kidney segmentation tasks. Although overall Dice scores of the baseline models were arguably satisfactory, this indicates that the baseline method lacks robustness against great variation of input data. The NDSF and NDSF+ models, on the other hand, had much better minimum Dice scores (Fig.18).

Robustness gained from the proposed methods was additionally observed from the distributions of Dice scores of each method (Fig.19). It is clear that the proposed methods successfully handled cases whose Dice scores were low in the baseline method. It is interesting to see that although NDSF had many cases with very high Dice scores, it tended to have more cases whose Dice scores were somewhat low. This indicates that NDSF+ could segment regions more consistently than the other two methods.

Table 2: *Accuracy indices from three HU mapping techniques, grouped by the size of training sets, when models were tested against the three datasets. Column 'Tr Size' denotes the size of training set, while Columns Dice and RMSD represent performance metrics discussed in Section 4.1. The baseline model is Cascaded 2D U-Net with NMIM normalization.*

		LiTS		3DIRCADb		SLiver07	
TrSize	Model	Dice	RMSD	Dice	RMSD	Dice	RMSD
TR025	baseline	0.963	4.481	0.942	4.929	0.868	6.016
	NDSF	**0.977**	2.770	0.963	7.224	0.948	**4.888**
	NDSF+	**0.977**	**1.968**	**0.966**	**6.068**	**0.954**	5.629
TR050	baseline	0.962	4.404	0.942	4.509	0.867	6.300
	NDSF	**0.976**	1.719	0.965	3.058	0.947	**4.418**
	NDSF+	**0.976**	**1.631**	**0.966**	**3.018**	**0.953**	4.765
TR075	baseline	0.961	4.447	0.942	3.929	0.864	10.67
	NDSF	0.976	1.666	0.966	**2.456**	0.948	**4.167**
	NDSF+	**0.977**	**1.634**	**0.967**	2.528	**0.953**	4.754
TR100	baseline	0.963	4.326	0.943	3.165	0.868	6.512
	NDSF	**0.977**	1.653	0.967	**1.881**	0.951	**3.659**
	NDSF+	**0.977**	**1.581**	**0.968**	2.048	**0.958**	3.755

Table 3: *Accuracy indices from three HU mapping techniques, grouped by the size of training sets, when models were tested against the KiTS dataset.*

TrSize	Model	Dice	VOE	RVD	ASD	RMSD	MSSD
TR025	baseline	0.930	12.05	-7.530	2.606	6.427	69.48
	NDSF	0.960	7.387	-1.602	1.836	5.358	75.43
	NDSF+	**0.963**	**6.903**	**-1.226**	**1.692**	**5.099**	**69.77**
TR050	baseline	0.930	12.04	-8.011	2.259	4.840	45.85
	NDSF	0.961	7.320	-2.261	**1.352**	**3.309**	**37.25**
	NDSF+	**0.963**	**6.960**	**-1.761**	1.387	3.429	37.43
TR075	baseline	0.930	12.04	-8.058	2.551	5.718	51.88
	NDSF	0.961	7.189	-2.092	**1.281**	**2.961**	**35.18**
	NDSF+	**0.964**	**6.848**	**-1.598**	1.351	3.238	36.19
TR100	baseline	0.937	10.97	-7.319	1.968	3.974	35.31
	NDSF	0.963	6.948	-2.216	**1.130**	**2.520**	**26.79**
	NDSF+	**0.966**	**6.518**	**-1.765**	1.143	2.625	27.59

4.5 Comparison with Other Methods

Due to different settings and goals from prior work, it is important to note that the results of our work cannot be directly compared with others. For example, SLiver07 includes both liver and lesions as a single region, while LiTS and 3DIRCADb separate them. Also, we have more interest in knowing the volumes of the liver and kidneys to help diagnose symptoms not directly connected to tumors [1][39].

However, comparison with other work still provides some insights about the benefits of the proposed method, especially when considering the fact that the models of our methods were using just simple 2D U-Net, while others mostly relied on much more complex 3D models. In this comparison, the average accuracy from 3 folds where our models where trained with 100% of the training data (TR100) represents the performance of the proposed method in this section.

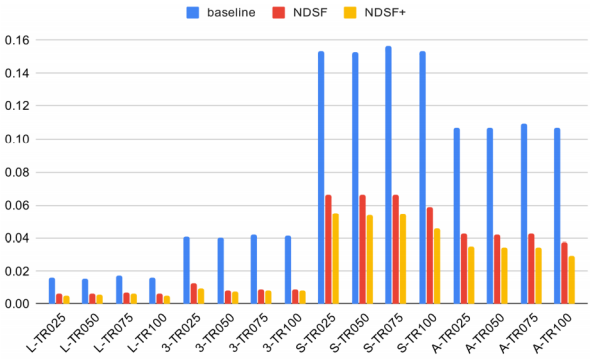

Fig. 16: *Comparison of standard deviation of liver segmentation Dice scores from models trained by different methods (lower is better). Prefixes L-, 3-, and S- denote results from LiTS, 3DIRCADb, and SLiver datasets, while Prefix A- denote average results from the three datasets. Suffixes TR025, TR050, TR075, and TR100 denote the training sizes as described earlier.*

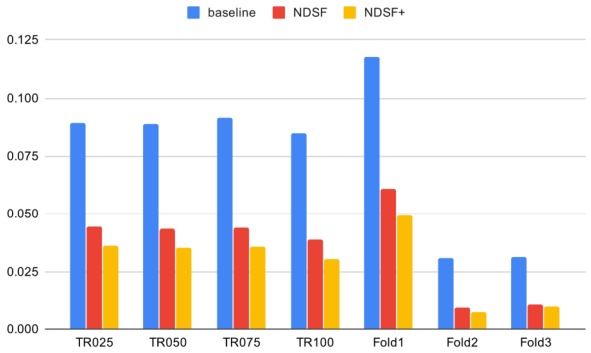

Fig. 17: *Comparison of standard deviation of kidney-segmentation Dice scores from models trained by different methods (lower is better). Since there was only one dataset (KiTS), we observed results with regard to training sizes and folds. Once again, NDSF+ was the best in this aspect and outperformed other methods in all training sizes and folds.*

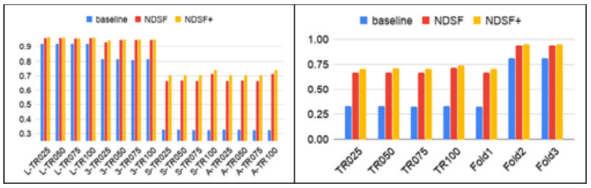

Fig. 18: *Comparison of minimum of Dice scores from models trained by different methods (higher is better). Left: Dice scores of liver segmentation. Right: Dice scores of kidney segmentation. These results are Dice scores from specific images where models performed worst.*

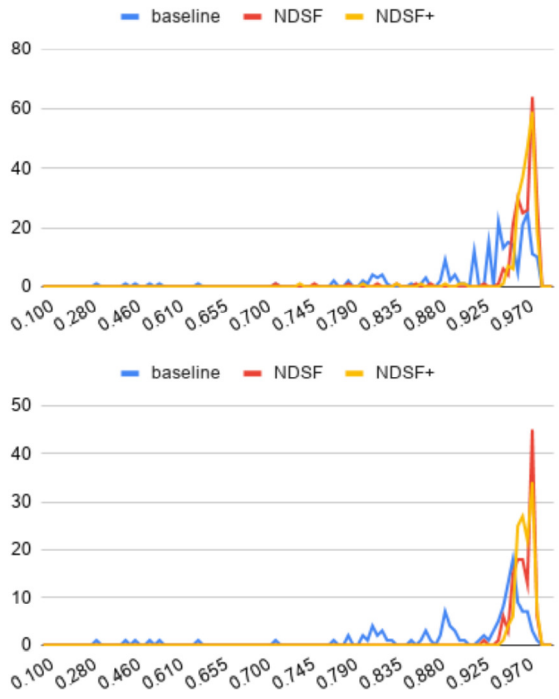

Fig.19: *Distributions of Dice scores from each method. Top: histograms of liver-segmentation Dice scores. Bottom: histograms of kidney-segmentation Dice scores.*

According to Tables 4 through 6, the proposed method performed comparatively well across the three datasets. When compared to most methods reporting results on both 3DIRCADb and SLiver07 datasets [40]–[42], it is clear that our method was much less influenced by change of datasets. In addition, on average, our method outperformed those methods in VOE. Only the work of S. Zheng et al. achieved the same accuracy as our method [18]. Nonetheless, our models were solely trained with the LiTS dataset and not trained with the 3DIRCADb or SLiver07 datasets, while others had more opportunities to tune their methods for the two datasets. This could make our models appear less competent when tested against these datasets. In addition, prior work dealt with only one or two datasets, while ours dealt with three.

X. Li et al. reported accuracy for both LiTS and 3DIRCADb [3]. Their method performed extraordinarily well in 3DIRCADb. However, their accuracy in the LiTS dataset was lower. Our proposed method, on the other hand, delivers similar accuracy for both datasets (difference < 0.01 in a Dice score) and it also achieved better average accuracy for the two datasets. F. Lu et al.'s method [42] was the best one for SLiver07, but it performed relatively poorly for 3DIRCADb. Once again, on average, our NDSF+ outperformed F. Lu et al.'s. This implies that our method performed consistently well across multiple datasets.

For the kidney segmentation task, the models competing in the KiTS challenge that were based on a cascaded 3D U-Net performed well and achieved 0.967 or more in the Dice score.

Table 7 displays the top-5 teams from the competition leaderboard [43]. Although the proposed method significantly improved the accuracy of a baseline cascaded 2D U-Net and delivered high segmentation accuracy (Dice score > 0.965), its accuracy was less than those top-5 3D U-Net variants. Since their work did not report standard variation of Dice scores, we could not evaluate the robustness of their methods against ours.

Table 4: *Comparison of liver segmentation accuracy with prior work in the LiTS dataset. Accuracy of Models deepX, leHealth, mabc, and H-DenseUNet (HDUNet) were taken from the leaderboard of MICCAI 2017 challenge. Prior works whose names are in bold face reported performance in at least two datasets.*

Model	Dice	VOE	RVD	ASD	RMSD	MSSD
deepX	0.963	7.1	-0.010	1.104	2.303	23.85
leHealth	0.961	7.5	0.023	1.268	2.776	27.02
mabc	0.960	7.0	**0.000**	1.130	2.390	24.45
HDUNet[3]	0.961	7.4	**0.000**	1.692	3.729	29.41
Xia[30]	0.970	7.9	0.006	1.925	-	35.58
our baseline	0.963	7.085	-3.912	1.754	4.326	31.34
NDSF	**0.977**	4.506	-1.126	0.680	1.653	**19.31**
NDSF+	**0.977**	**4.460**	-0.730	**0.661**	**1.581**	21.30

Table 5: *Comparison of liver segmentation accuracy with prior work in the 3DIRCADb dataset.*

Model	Dice	VOE	RVD	ASD	RMSD	MSSD
Christ[26]	0.943	10.7	-1.4	1.5	-	24.01
G.Li[40]	-	9.15	-0.07	1.55	3.15	28.22
F.Ju[42]	-	9.36	0.97	1.89	4.15	33.14
X.Lu[41]	-	9.21	1.27	1.75	3.95	36.17
HDUNet[3]	**0.982**	**3.57**	**0.01**	1.28	3.58	-
S.Zheng[18]	-	6.5	2.1	1.9	2.1	**18.9**
Jiang[14]	0.959	-	-	-	-	-
our baseline	0.943	10.539	-6.209	1.733	3.165	26.19
NDSF	0.967	6.411	-0.770	0.955	**1.881**	22.24
NDSF+	0.968	6.279	-0.972	**0.962**	2.048	26.71

5. CONCLUSION

This work presents a discriminative image enhancement and data augmentation method which can be generalized for multiple segmentation tasks and applied to other existing cascaded models. During model training, we can adjust the image-enhancement parameters to produce more image data in the middle of a segmentation workflow.

Robustness and accuracy were significantly improved from a common cascaded segmentation method when the discriminative image enhancement was utilized. The proposed data augmentation further improved robustness and the trained models had less variation in Dice scores. It is interesting that we could significantly improve robustness and accuracy of a cascaded model without any change to the model itself. This can help avoid unnecessary model complexity while achieving comparable accuracy to more

complex models. It can also improve robustness of a competent model if needed.

The proposed discriminative feature generation may also be applied to an attention-based model in that we create an HU histogram based on voxels having high likelihood of belonging to a target region. This means that instead of employing a localization network to find bounding boxes and accumulate HU statistics from them, we may resort to an attention mechanism and gather HU statistics from attention masks. This is future work we think worthy of exploration.

Table 6: *Comparison of liver segmentation accuracy with prior work in the SLiver07 dataset.*

Model	Dice	VOE	RVD	ASD	RMSD	MSSD
Wu[19]	-	7.54	4.16	0.95	1.94	18.48
G.Li[40]	-	6.24	1.18	1.13	2.11	18.82
F.Ju[42]	-	**5.09**	2.70	0.91	1.88	18.94
X.Lu[41]	-	5.92	1.03	1.06	1.68	12.33
S.Zheng[18]	-	7.60	**-0.01**	**0.8**	**1.5**	20.8
Y.Zheng[44]	-	7.83	5.06	1.06	1.39	**11.12**
our baseline	0.868	20.649	-19.60	3.569	6.512	49.009
NDSF	0.951	8.787	-4.469	1.554	3.659	35.686
NDSF+	0.958	7.827	-3.134	1.579	3.755	31.858

Table 7: *Comparison of kidney segmentation accuracy with prior work in the KiTS dataset. Accuracy of Models X. Hou, G. Mu, and Y. Zhang were taken from the leaderboard of KiTS19 (MICCAI 2019) challenge [17], [43]. The technical reports of these methods are accessible through the leaderboard website [43]. Since only Dice scores were reported, most entries in this table have no data.*

Model	Dice	VOE	RVD	ASD	RMSD	MSSD
Isensee[4]	0.9737	-	-	-	-	-
X.Hou	0.9674	-	-	-	-	-
G.Mu	0.9729	-	-	-	-	-
Y.Zhang	**0.9742**	-	-	-	-	-
our baseline	0.9368	10.968	-7.3189	1.9680	3.9742	35.315
NDSF	0.9628	6.948	-2.2160	1.1299	2.5199	26.789
NDSF+	0.9655	6.518	-1.7649	1.1427	2.6249	27.595

Our method, however, has some limitations that can be addressed in future work. First, it employs a predefined constant stretching parameter $\sigma = 0.07$, but we observed that HUs of several kidney regions were not well stretched. They were far left of the center value. This issue may be handled by adaptively finding a more appropriate stretching parameter value for σ. Second, if a target region has multiple dominant peaks far away from one another in the HU histogram, the proposed HU reassignment function may not provide adequate discriminative features to certain regions. In all datasets in our experiments, however, no such case was ever encountered. More testing with larger and more varied datasets is needed to investigate this.

ACKNOWLEDGMENT

This work was supported in part by a Mahidol Wittayanusorn School scholarship.

References

[1] H. Ryu et al., "Total kidney and liver volume is a major risk factor for malnutrition in ambulatory patients with autosomal dominant polycystic kidney disease," *BMC Nephrol.*, vol. 18, no. 1, p. 22, Dec. 2017.

[2] A. L. Simpson et al., "A large annotated medical image dataset for the development and evaluation of segmentation algorithms," in *arxiv.org*, 2019.

[3] X. Li, H. Chen, X. Qi, Q. Dou, C.-W. Fu, and P.-A. Heng, "H-DenseUNet: Hybrid Densely Connected UNet for Liver and Tumor Segmentation From CT Volumes," *IEEE Trans. Med. Imaging*, vol. 37, no. 12, pp. 2663–2674, Dec. 2018.

[4] F. Isensee and K. H. Maier-Hein, "An attempt at beating the 3D U-Net," in *arxiv.org*, 2019.

[5] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal Loss for Dense Object Detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, vol. 2017-Octob, pp. 2999–3007.

[6] F. B.V., "fizyr/keras-retinanet: Keras implementation of RetinaNet object detection," [Online]. Available: `github.com/fizyr/keras-retinanet`. [Accessed: 23-Nov-2019].

[7] P. Bilic et al., "The Liver Tumor Segmentation Benchmark (LiTS)," in *arxiv.org*, 2019.

[8] L. Soler et al., "3D Image reconstruction for comparison of algorithm database : A patient specific anatomical and medical image database," [Online]. Available: `ircad.fr/softwares/3dircadb`.

[9] B. Van Ginneken, T. Heimann, and M. Styner, "3D segmentation in the clinic: A grand challenge," in *In: MICCAI Workshop on 3D Segmentation in the Clinic: A Grand Challenge (2007)*, 2007.

[10] P. F. Jaeger et al., "Retina U-Net: Embarrassingly Simple Exploitation of Segmentation Supervision for Medical Object Detection," Nov. 2018.

[11] A. Suzani, A. Seitel, Y. Liu, S. Fels, R. N. Rohling, and P. Abolmaesumi, "Fast automatic vertebrae detection and localization in pathological CT scans - a deep learning approach," in *Lecture Notes in Computer Science*, 2015, vol. 9351, pp. 678–686.

[12] R. Janssens, G. Zeng, and G. Zheng, "Fully automatic segmentation of lumbar vertebrae from CT images using cascaded 3D fully convolutional networks," in *2018 IEEE 15th Int. Symp. on Biomedical Imaging (ISBI 2018)*, 2018, pp. 893–897.

[13] I. I. N. Lessmann, B. van Ginneken, P. A. de Jong, "Iterative convolutional neural networks for automatic vertebra identification and segmentation in CT images," in *Medical Imaging 2018: Image Processing*, 2018, vol. 10574, p. 7.

[14] H. Jiang, T. Shi, Z. Bai, and L. Huang, "AHCNet: An Application of Attention Mechanism and

Hybrid Connection for Liver Tumor Segmentation in CT Volumes," *IEEE Access*, vol. 7, pp. 24898–24909, 2019.

[15] A. Myronenko and A. Hatamizadeh, "3D Kidneys and Kidney Tumor Semantic Segmentation using Boundary-Aware Networks," in *arxiv.org*, 2019.

[16] Y. Yuan, "Hierarchical Convolutional-Deconvolutional Neural Networks for Automatic Liver and Tumor Segmentation," Oct. 2017.

[17] N. Heller et al., "The KiTS19 Challenge Data: 300 Kidney Tumor Cases with Clinical Context, CT Semantic Segmentations, and Surgical Outcomes."

[18] S. Zheng, B. Fang, L. Li, M. Gao, and Y. Wang, "A variational approach to liver segmentation using statistics from multiple sources," *Phys. Med. Biol.*, vol. 63, no. 2, Jan. 2018.

[19] W. Wu, Z. Zhou, S. Wu, and Y. Zhang, "Automatic Liver Segmentation on Volumetric CT Images Using Supervoxel-Based Graph Cuts," *Comput. Math. Methods Med.*, vol. 2016, pp. 1–14, 2016.

[20] D. Keshwani, Y. Kitamura, and Y. Li, "Computation of total kidney volume from CT images in autosomal dominant polycystic kidney disease using multi-task 3D convolutional neural networks," in *Lecture Notes in Computer Science*, 2018, vol. 11046 LNCS, pp. 380–388.

[21] F. Isensee et al., "nnU-Net: Self-adapting Framework for U-Net-Based Medical Image Segmentation," in *Informatik aktuell*, 2019, p. 22.

[22] M. Perslev, E. Dam, A. Pai, and C. Igel, "One Network To Segment Them All: A General, Lightweight System for Accurate 3D Medical Image Segmentation," *Med. Image Comput. Comput. Assist. Interv.*, vol. 11, no. 2, p. 24001, 2019.

[23] C. Huang, H. Han, Q. Yao, S. Zhu, and S. K. Zhou, "3D U-Net: A 3D Universal U-Net for Multi-domain Medical Image Segmentation," 2019, pp. 291–299.

[24] Y. Qin et al., "Autofocus layer for semantic segmentation," in *Lecture Notes in Computer Science*, 2018, vol. 11072 LNCS, pp. 603–611.

[25] B. Kayalibay, G. Jensen, and P. van der Smagt, "CNN-based Segmentation of Medical Imaging Data," Jan. 2017.

[26] P. F. Christ et al., "Automatic Liver and Tumor Segmentation of CT and MRI Volumes using Cascaded Fully Convolutional Neural Networks," in *arxiv.org*, 2017.

[27] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Springer*, Cham, 2015, pp. 234–241.

[28] K. Yin et al., "Deep learning segmentation of kidneys with renal cell carcinoma.," *J. Clin. Oncol.*, vol. 37, no. 15_suppl, pp. e16098–e16098, May 2019.

[29] J. A. O'Reilly, M. Sangworasil, and T. Matsuura, "Kidney and Kidney Tumor Segmentation using a Logical Ensemble of U-nets with Volumetric Validation," *arxiv.org*, Aug. 2019.

[30] K. Xia, H. Yin, P. Qian, Y. Jiang, and S. Wang, "Liver Semantic Segmentation Algorithm Based on Improved Deep Adversarial Networks in Combination of Weighted Loss Function on Abdominal CT Images," *IEEE Access*, vol. 7, pp. 96349–96358, Jul. 2019.

[31] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," *Lect. Notes Comput. Sci.*, vol. 11211 LNCS, pp. 833–851, Feb. 2018.

[32] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, 2018.

[33] L. Wang, R. Chen, S. Wang, N. Zeng, X. Huang, and C. Liu, "Nested Dilation Network (NDN) for Multi-Task Medical Image Segmentation," *IEEE Access*, vol. 7, pp. 44676–44685, 2019.

[34] R. Lamba et al., "CT Hounsfield numbers of soft tissues on unenhanced abdominal CT scans: Variability between two different manufacturers' MDCT scanners," *Am. J. Roentgenol.*, vol. 203, no. 5, pp. 1013–1020, Nov. 2014.

[35] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Lecture Notes in Computer Science*, vol. 9351, Springer, Cham, 2015, pp. 234–241.

[36] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, 2014.

[37] S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," in *Proceedings of the 32nd Int. Conf. on Machine Learning*, 2015, vol. 37, pp. 448–456.

[38] A. A. Taha and A. Hanbury, "Metrics for evaluating 3D medical image segmentation: Analysis, selection, and tool," *BMC Med. Imaging*, vol. 15, no. 1, Aug. 2015.

[39] K. Sharma et al., "Automatic Segmentation of Kidneys using Deep Learning for Total Kidney Volume Quantification in Autosomal Dominant Polycystic Kidney Disease," *Sci. Rep.*, vol. 7, no. 1, Dec. 2017.

[40] G. Li, X. Chen, F. Shi, W. Zhu, J. Tian, and D. Xiang, "Automatic Liver Segmentation Based

on Shape Constraints and Deformable Graph Cut in CT Images," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5315–5329, Dec. 2015.

[41] X. Lu, Q. Xie, Y. Zha, and D. Wang, "Fully automatic liver segmentation combining multi-dimensional graph cut with shape information in 3D CT images," *Sci. Rep.*, vol. 8, no. 1, p. 10700, Dec. 2018.

[42] F. Lu, F. Wu, P. Hu, Z. Peng, and D. Kong, "Automatic 3D liver location and segmentation via convolutional neural network and graph cut," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 12, no. 2, pp. 171–182, Feb. 2017.

[43] "KiTS19 Results." [Online]. Available: `http://results.kits-challenge.org/miccai2019/`. [Accessed: 22-Nov-2019].

[44] Y. Zheng et al., "Automatic liver segmentation based on appearance and context information," *Biomed. Eng. Online*, vol. 16, no. 1, p. 16, Dec. 2017.

Boonnatee Sakboonyarat is a teacher in the Department of Mathematics and Computing Science, Mahidol Wittayanusorn School, Thailand. He earned his B.Sc. in Computer Science from Nakhon Pathom Rajabhat University, Thailand, in 2001 and his M.S. in Computer Science from Silpakorn University in 2006. Currently, he is a Ph.D. cadidate in Information Technology from Silpakorn University, Thailand. His research involes machine learning, computer vision, and biomedical image processing.

Pinyo Taeprasartsit is an assistant professor in the Department of Computing, Faculty of Science, Silpakorn University, Thailand. He earned his B.Eng in Computer Engineering from Chulalongkorn University in 2001 and Ph.D. in Computer Science and Engineering from The Pennsylvania State University in 2011. His research involves biomedical image processing, machine learning, and computer vision.