# Recurrent Saliency Transformation Network: Incorporating Multi-Stage Visual Cues for Small Organ Segmentation

Qihang Yu[1], Lingxi Xie[2](✉), Yan Wang[2], Yuyin Zhou[2], Elliot K. Fishman[3], Alan L. Yuille[2]
[1]Peking University [2]The Johns Hopkins University [3]The Johns Hopkins Medical Institute
{yucornetto,198808xc,wyanny.9,zhouyuyiner,alan.l.yuille}@gmail.com efishman@jhmi.edu

## Abstract

*We aim at segmenting small organs (e.g., the pancreas) from abdominal CT scans. As the target often occupies a relatively small region in the input image, deep neural networks can be easily confused by the complex and variable background. To alleviate this, researchers proposed a coarse-to-fine approach [46], which used prediction from the first (coarse) stage to indicate a smaller input region for the second (fine) stage. Despite its effectiveness, this algorithm dealt with two stages individually, which lacked optimizing a global energy function, and limited its ability to incorporate multi-stage visual cues. Missing contextual information led to unsatisfying convergence in iterations, and that the fine stage sometimes produced even lower segmentation accuracy than the coarse stage.*

*This paper presents a* **Recurrent Saliency Transformation Network**. *The key innovation is a saliency transformation module, which repeatedly converts the segmentation probability map from the previous iteration as spatial weights and applies these weights to the current iteration. This brings us two-fold benefits. In training, it allows joint optimization over the deep networks dealing with different input scales. In testing, it propagates multi-stage visual information throughout iterations to improve segmentation accuracy. Experiments in the NIH pancreas segmentation dataset demonstrate the state-of-the-art accuracy, which outperforms the previous best by an average of over 2%. Much higher accuracies are also reported on several small organs in a larger dataset collected by ourselves. In addition, our approach enjoys better convergence properties, making it more efficient and reliable in practice.*

## 1. Introduction

This paper focuses on small organ (*e.g.*, the *pancreas*) segmentation from abdominal CT scans, which is an important prerequisite for enabling computers to assist human doctors for clinical purposes. This problem falls into the research area named *medical imaging analysis*. Recently, great progress has been brought to this field by the fast development of deep learning, especially convolutional neural networks [18][27]. Many conventional methods, such as the graph-based segmentation approaches [1] or those based on handcrafted local features [41], have been replaced by deep segmentation networks, which typically produce higher segmentation accuracy [33][34].

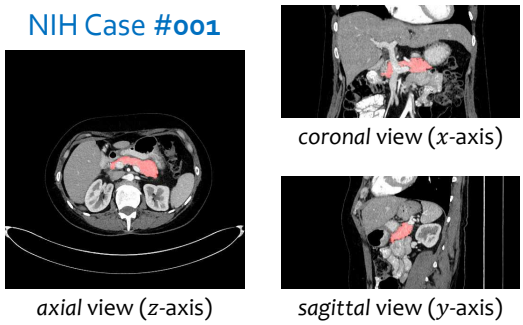


Figure 1. A typical example from the NIH *pancreas* segmentation dataset [34] (best viewed in color). We highlight the *pancreas* in red seen from three different viewpoints. It is a relatively small organ with irregular shape and boundary.

Segmenting a small organ from CT scans is often challenging. As the target often occupies a *small part* of input data (*e.g.*, less than 1.5% in a 2D image, see Figure 1), deep segmentation networks such as FCN [27] and DeepLab [5] can be easily confused by the background region, which may contain complicated and variable contents. This motivates researchers to propose a *coarse-to-fine* approach [46] with two *stages*, in which the coarse stage provides a rough localization and the fine stage performs accurate segmentation. But, despite state-of-the-art performance achieved in pancreas segmentation, this method suffers from *inconsistency* between its training and testing flowcharts, which is to say, the training phase dealt with coarse and fine stages individually and did not minimize a global energy function, but the testing phase assumed that these two stages can

cooperate with each other in an iterative process. From another perspective, this also makes it difficult for multi-stage visual cues to be incorporated in segmentation, *e.g.*, the previous segmentation mask which carries rich information is discarded except for the bounding box. As a part of its consequences, the fine stage consisting of a sequence of iterations cannot converge very well, and sometimes the fine stage produced even lower segmentation accuracy than the coarse stage (see Section 3.1).

Motivated to alleviate these shortcomings, we propose a **Recurrent Saliency Transformation Network**. The chief innovation is to relate the coarse and fine stages with a saliency transformation module, which repeatedly transforms the segmentation probability map from previous iterations as spatial priors in the current iteration. This brings us two-fold advantages over [46]. First, in the training phase, the coarse-scaled and fine-scaled networks are optimized jointly, so that the segmentation ability of each of them gets improved. Second, in the testing phase, the segmentation mask of each iteration is preserved and propagated throughout iterations, enabling multi-stage visual cues to be incorporated towards more accurate segmentation. To the best of our knowledge, this idea was not studied in the computer vision community, as it requires making use of some special properties of CT scans (see Section 3.4).

We perform experiments on two CT datasets for small organ segmentation. On the NIH *pancreas* segmentation dataset [34], our approach outperforms the state-of-the-art by an average of over $2\%$, measured by the average Dice-Sørensen coefficient (DSC). On another multi-organ dataset collected by the radiologists in our team, we also show the superiority of our approach over the baseline on a variety of small organs. In the testing phase, our approach enjoys better convergence properties, which guarantees its efficiency and reliability in real clinical applications.

The remainder of this paper is organized as follows. Section 2 briefly reviews related work, and Section 3 describes the proposed approach. After experiments are shown in Sections 4 and 5, we draw our conclusions in Section 6.

## 2. Related Work

Computer-aided diagnosis (CAD) is an important technique which can assist human doctors in many clinical scenarios. An important prerequisite of CAD is medical imaging analysis. As a popular and cheap way of medical imaging, contrast-enhanced computed tomography (CECT) produces detailed images of internal organs, bones, soft tissues and blood vessels. It is of great value to automatically segment organs and/or soft tissues from these CT volumes for further diagnosis [2][40][13][45]. To capture specific properties of different organs, researchers often design individualized algorithms for each of them. Typical examples include the the liver [25][15], the *spleen* [26], the *kidneys* [23][1], the *lungs* [16], the *pancreas* [7][41], *etc*. Small organs (*e.g.*, the *pancreas*) are often more difficult to segment, partly due to their low contrast and large anatomical variability in size and (most often irregular) shape.

Compared to the papers cited above which used conventional approaches for segmentation, the progress of deep learning brought more powerful and efficient solutions. In particular, convolutional neural networks have been widely applied to a wide range of vision tasks, such as image classification [18][37][14], object detection [10][32], and semantic segmentation [27][5]. Recurrent neural networks, as a related class of networks, were first designed to process sequential data [11][39], and later generalized to image classification [22] and scene labeling [31] tasks. In the area of medical imaging analysis, in particular organ segmentation, these techniques have been shown to significantly outperform conventional approaches, *e.g.*, segmenting the *liver* [8], the *lung* [12], or the *pancreas* [35][3][36]. Note that medical images differ from natural images in that data appear in a volumetric form. To deal with these data, researchers either slice a 3D volume into 2D slices (as in this work), or train a 3D network directly [29][30][17][43]. In the latter case, limited GPU memory often leads to patch-based training and testing strategies. The tradeoff between 2D and 3D approaches is discussed in [20].

By comparison to the entire CT volume, the organs considered in this paper often occupy a relatively small area. As deep segmentation networks such as FCN [27] are less accurate in depicting small targets, researchers proposed two types of ideas to improve detection and/or segmentation performance. The first type involved rescaling the image so that the target becomes comparable to the training samples [42], and the second one considered to focus on a subregion of the image for each target to obtain higher accuracy in detection [4] or segmentation [46]. The coarse-to-fine idea was also well studied in the computer vision area for saliency detection [19] or semantic segmentation [21][24]. This paper is based on a recent coarse-to-fine framework [46], but we go one step further by incorporating multi-stage visual cues in optimization.

## 3. Our Approach

We investigate the problem of segmenting an organ from abdominal CT scans. Let a CT image be a 3D volume $\mathbf{X}$ of size $W \times H \times L$ which is annotated with a binary ground-truth segmentation $\mathbf{Y}$ where $y_i = 1$ indicates a foreground voxel. The goal of our work is to produce a binary output volume $\mathbf{Z}$ of the same dimension. Denote $\mathcal{Y}$ and $\mathcal{Z}$ as the set of foreground voxels in the ground-truth and prediction, *i.e.*, $\mathcal{Y} = \{i \mid y_i = 1\}$ and $\mathcal{Z} = \{i \mid z_i = 1\}$. The accuracy of segmentation is evaluated by the Dice-Sørensen coefficient (DSC): $\mathrm{DSC}(\mathcal{Y}, \mathcal{Z}) = \frac{2\times|\mathcal{Y}\cap\mathcal{Z}|}{|\mathcal{Y}|+|\mathcal{Z}|}$. This metric falls in the range of $[0, 1]$ with 1 implying perfect segmentation.

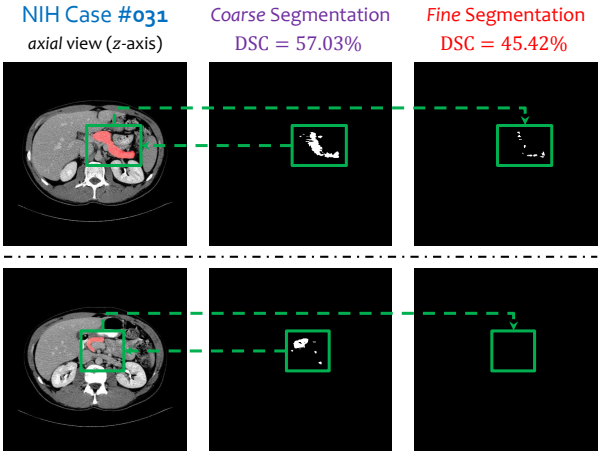


Figure 2. A failure case of the stage-wise *pancreas* segmentation approach [46] (in the *axial* view, best viewed in color). The red masks show ground-truth segmentations, and the green frames indicate the bounding box derived from the coarse stage. In either slice, unsatisfying segmentation is produced at the fine stage, because the cropped region does not contain enough contextual information, whereas the coarse-scaled probability map carrying such information is discarded. This is improved by the proposed Recurrent Saliency Transformation Network, see Figure 5.

### 3.1. Coarse-to-Fine Segmentation and Drawbacks

We start with training 2D deep networks for 3D segmentation[1]. Each 3D volume $\mathbf{X}$ is sliced along three axes, the *coronal*, *sagittal* and *axial* views, and these 2D slices are denoted by $\mathbf{X}_{\mathrm{C},w}$ ($w = 1, 2, \ldots, W$), $\mathbf{X}_{\mathrm{S},h}$ ($h = 1, 2, \ldots, H$) and $\mathbf{X}_{\mathrm{A},l}$ ($l = 1, 2, \ldots, L$), where the subscripts C, S and A stand for *coronal*, *sagittal* and *axial*, respectively. On each axis, an individual 2D-FCN [27] on a 16-layer VGGNet [37] is trained[2]. Three FCN models are denoted by $\mathbb{M}_{\mathrm{C}}$, $\mathbb{M}_{\mathrm{S}}$ and $\mathbb{M}_{\mathrm{A}}$, respectively. We use the DSC loss [30] in the training phase so as to prevent the models from being biased towards the background class. Both multi-slice segmentation (3 neighboring slices are combined as a basic unit in training and testing) and multi-axis fusion (majority voting over three axes) are performed to incorporate pseudo-3D information into segmentation.

The organs investigated in this paper (*e.g.*, the *pancreas*) are relatively small. In each 2D slice, the fraction of the foreground pixels is often smaller than $1.5\%$. To prevent deep networks such as FCN [27] from being confused by the complicated and variable background contents, [46] proposed to focus on a smaller input region according to an estimated bounding box. On each viewpoint, two networks were trained for coarse-scaled segmentation and fine-scaled segmentation, respectively. In the testing process, the coarse-scaled network was first used to obtain the rough position of the *pancreas*, and the fine-scaled network was executed several times and the segmentation mask was updated iteratively until convergence.

Despite the significant accuracy gain brought by this approach, we notice a drawback originating from the *inconsistency* between its training and testing strategies. That is to say, the training stage dealt with two networks individually without enabling global optimization, but the testing phase assumed that they can cooperate with each other in a sequence of iterations. From another perspective, a pixel-wise segmentation probability map was predicted by the coarse stage, but the fine stage merely preserved the bounding box and discarded the remainder, which is a major information loss. Sometimes, the image region within the bounding box does not contain sufficient spatial contexts, and thus the fine stage can be confused and produce even lower segmentation accuracy than the coarse stage. A failure case is shown in Figure 2. This motivates us to connect these two stages with a saliency transformation module so as to jointly optimize their parameters.

### 3.2. Recurrent Saliency Transformation Network

Following the baseline approach, we train an individual model for each of the three viewpoints. Without loss of generality, we consider a 2D slice along the *axial* view, denoted by $\mathbf{X}_{\mathrm{A},l}$. Our goal is to infer a binary segmentation mask $\mathbf{Z}_{\mathrm{A},l}$ of the same dimensionality. In the context of deep neural networks [27][5], this is often achieved by first computing a *probability map* $\mathbf{P}_{\mathrm{A},l} = \mathbf{f}[\mathbf{X}_{\mathrm{A},l}; \boldsymbol{\theta}]$, where $\mathbf{f}[\cdot; \boldsymbol{\theta}]$ is a deep segmentation network (FCN throughout this paper) with $\boldsymbol{\theta}$ being network parameters, and then binarizing $\mathbf{P}_{\mathrm{A},l}$ into $\mathbf{Z}_{\mathrm{A},l}$ using a fixed threshold of $0.5$, *i.e.*, $\mathbf{Z}_{\mathrm{A},l} = \mathbb{I}[\mathbf{P}_{\mathrm{A},l} \geqslant 0.5]$.

In order to assist segmentation with the probability map, we introduce $\mathbf{P}_{\mathrm{A},l}$ as a latent variable. We introduce a *saliency transformation* module, which takes the probability map to generate an updated input image, *i.e.*, $\mathbf{I}_{\mathrm{A},l} = \mathbf{X}_{\mathrm{A},l} \odot \mathbf{g}(\mathbf{P}_{\mathrm{A},l}; \boldsymbol{\eta})$, and uses the updated input $\mathbf{I}_{\mathrm{A},l}$ to replace $\mathbf{X}_{\mathrm{A},l}$. Here $\mathbf{g}[\cdot; \boldsymbol{\eta}]$ is the transformation function with parameters $\boldsymbol{\eta}$, and $\odot$ denotes element-wise product, *i.e.*, the transformation function adds spatial weights to the original input image. Thus, the segmentation process becomes:

$$\mathbf{P}_{\mathrm{A},l} = \mathbf{f}[\mathbf{X}_{\mathrm{A},l} \odot \mathbf{g}(\mathbf{P}_{\mathrm{A},l}; \boldsymbol{\eta}); \boldsymbol{\theta}]. \tag{1}$$

This is a recurrent neural network. Note that the saliency transformation function $\mathbf{g}[\cdot, \boldsymbol{\eta}]$ needs to be differentiable so that the entire recurrent network can be optimized in an end-to-end manner. As $\mathbf{X}_{\mathrm{A},l}$ and $\mathbf{P}_{\mathrm{A},l}$ share the same

[1] Please see Section 4.3 for the comparison to 3D networks.

[2] This is a simple segmentation baseline with a relatively shallow network. Deeper network structures such as ResNet [14] and more complicated segmentation frameworks such as DeepLab [5], while requiring a larger memory and preventing us from training two stages jointly (see Section 3.2), often result in lower segmentation accuracy as these models seem to over-fit in these CT datasets.

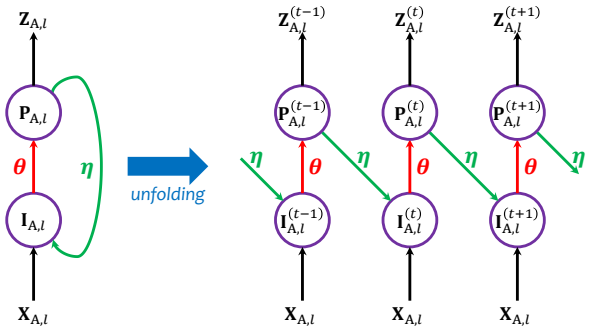


Figure 3. We formulate our approach into a recurrent network, and unfold it for optimization and inference.

spatial dimensionality, we set $\mathbf{g}[\cdot, \boldsymbol{\eta}]$ to be a *size-preserved* convolution, which allows the weight added to each pixel to be determined by the segmentation probabilities in a small neighborhood around it. As we will show in the experimental section (see Figure 5), the learned convolutional kernels are able to extract complementary information to help the next iteration.

To optimize Eqn (1), we unfold the recurrent network into a plain form (see Figure 3). Given an input image $\mathbf{X}_{\mathrm{A},l}$ and an integer $T$ which is the maximal number of iterations, we update $\mathbf{I}_{\mathrm{A},l}^{(t)}$ and $\mathbf{P}_{\mathrm{A},l}^{(t)}$, $t = 0, 1, \ldots, T$:

$$\mathbf{I}_{\mathrm{A},l}^{(t)} = \mathbf{X}_{\mathrm{A},l} \odot \mathbf{g}\Big(\mathbf{P}_{\mathrm{A},l}^{(t-1)}; \boldsymbol{\eta}\Big), \tag{2}$$

$$\mathbf{P}_{\mathrm{A},l}^{(t)} = \mathbf{f}\Big[\mathbf{I}_{\mathrm{A},l}^{(t)}; \boldsymbol{\theta}\Big]. \tag{3}$$

Note that the original input image $\mathbf{X}_{\mathrm{A},l}$ does not change, and the parameters $\boldsymbol{\theta}$ and $\boldsymbol{\eta}$ are shared by all iterations. At $t = 0$, we directly set $\mathbf{I}_{\mathrm{A},l}^{(0)} = \mathbf{X}_{\mathrm{A},l}$.

When segmentation masks $\mathbf{P}_{\mathrm{A},l}^{(t)}$ $(t = 0, 1, \ldots, T-1)$ are available for reference, deep networks benefit considerably from a shrunk input region especially when the target organ is very small. Thus, we define a *cropping* function $\mathrm{Crop}\Big[\cdot; \mathbf{P}_{\mathrm{A},l}^{(t)}\Big]$, which takes $\mathbf{P}_{\mathrm{A},l}^{(t)}$ as the *reference map*, binarizes it into $\mathbf{Z}_{\mathrm{A},l}^{(t)} = \mathbb{I}\Big[\mathbf{P}_{\mathrm{A},l}^{(t)} \geqslant 0.5\Big]$, finds the minimal rectangle covering all the activated pixels, and adds a $K$-pixel-wide margin (padding) around it. We fix $K$ to be 20; our algorithm is not sensitive to this parameter.

Finally note that $\mathbf{I}_{\mathrm{A},l}^{(0)}$, the original input (the entire 2D slice), is much larger than the cropped inputs $\mathbf{I}_{\mathrm{A},l}^{(t)}$ for $t > 0$. We train two FCN's to deal with such a major difference in input data. The first one is named the *coarse-scaled* segmentation network, which is used *only* in the first iteration. The second one, the *fine-scaled* segmentation network, takes the charge of all the remaining iterations. We denote their parameters by $\boldsymbol{\theta}^{\mathrm{C}}$ and $\boldsymbol{\theta}^{\mathrm{F}}$, respectively. These two FCN's are optimized jointly.

**Algorithm 1:** The Testing Phase

**Input** : input volume $\mathbf{X}$, viewpoint $\mathcal{V} = \{\mathrm{C}, \mathrm{S}, \mathrm{A}\}$;
parameters $\boldsymbol{\theta}_{\mathrm{v}}^{\mathrm{C}}$, $\boldsymbol{\theta}_{\mathrm{v}}^{\mathrm{F}}$ and $\boldsymbol{\eta}_{\mathrm{v}}$, $\mathrm{v} \in \mathcal{V}$;
max number of iterations $T$, threshold thr;

**Output**: segmentation volume $\mathbf{Z}$;

$t \leftarrow 0$, $\mathbf{I}_{\mathrm{v}}^{(0)} \leftarrow \mathbf{X}$, $\mathrm{v} \in \mathcal{V}$;

$\mathbf{P}_{\mathrm{v},l}^{(0)} \leftarrow \mathbf{f}\Big[\mathbf{I}_{\mathrm{v},l}^{(0)}; \boldsymbol{\theta}_{\mathrm{v}}^{\mathrm{C}}\Big]$, $\mathrm{v} \in \mathcal{V}$, $\forall l$;

$\mathbf{P}^{(0)} = \frac{\mathbf{P}_{\mathrm{C}}^{(0)} + \mathbf{P}_{\mathrm{S}}^{(0)} + \mathbf{P}_{\mathrm{A}}^{(0)}}{3}$, $\mathbf{Z}^{(0)} = \mathbb{I}\big[\mathbf{P}^{(0)} \geqslant 0.5\big]$;

**repeat**

  $t \leftarrow t + 1$;

  $\mathbf{I}_{\mathrm{v},l}^{(t)} \leftarrow \mathbf{X}_{\mathrm{v},l} \odot \mathbf{g}\Big(\mathbf{P}_{\mathrm{v},l}^{(t-1)}; \boldsymbol{\eta}\Big)$, $\mathrm{v} \in \mathcal{V}$, $\forall l$;

  $\mathbf{P}_{\mathrm{v},l}^{(t)} \leftarrow \mathbf{f}\Big[\mathrm{Crop}\Big[\mathbf{I}_{\mathrm{v},l}^{(t)}; \mathbf{P}_{\mathrm{v},l}^{(t-1)}\Big]; \boldsymbol{\theta}_{\mathrm{v}}^{\mathrm{F}}\Big]$, $\mathrm{v} \in \mathcal{V}$, $\forall l$;

  $\mathbf{P}^{(t)} = \frac{\mathbf{P}_{\mathrm{C}}^{(t)} + \mathbf{P}_{\mathrm{S}}^{(t)} + \mathbf{P}_{\mathrm{A}}^{(t)}}{3}$, $\mathbf{Z}^{(t)} = \mathbb{I}\big[\mathbf{P}^{(t)} \geqslant 0.5\big]$;

**until** $t = T$ **or** $\mathrm{DSC}\big\{\mathbf{Z}^{(t-1)}, \mathbf{Z}^{(t)}\big\} \geqslant \mathrm{thr}$;

**Return**: $\mathbf{Z} \leftarrow \mathbf{Z}^{(t)}$.

We compute a DSC loss term on each probability map $\mathbf{P}_{\mathrm{A},l}^{(t)}$, $t = 0, 1, \ldots, T$, and denote it by $\mathcal{L}\Big\{\mathbf{Y}_{\mathrm{A},l}, \mathbf{P}_{\mathrm{A},l}^{(t)}\Big\}$. Here, $\mathbf{Y}_{\mathrm{A},l}$ is the ground-truth segmentation mask, and $\mathcal{L}\{\mathbf{Y}, \mathbf{P}\} = 1 - \frac{2 \times \sum_i Y_i P_i}{\sum_i Y_i + P_i}$ is based on a *soft* version of DSC [30]. Our goal is to minimize the overall loss:

$$\mathcal{L} = \sum_{t=0}^{T} \lambda_t \cdot \mathcal{L}\Big\{\mathbf{Y}_{\mathrm{A},l}^{(t)}, \mathbf{Z}_{\mathrm{A},l}^{(t)}\Big\}. \tag{4}$$

This leads to joint optimization over all iterations, which involves network parameters $\boldsymbol{\theta}^{\mathrm{C}}$, $\boldsymbol{\theta}^{\mathrm{F}}$, and transformation parameters $\boldsymbol{\eta}$. $\{\lambda_t\}_{t=0}^{T}$ controls the tradeoff among all loss terms. We set $2\lambda_0 = \lambda_1 = \ldots = \lambda_T = 2/(2T+1)$ so as to encourage accurate fine-scaled segmentation.

### 3.3. Training and Testing

**The training phase** is aimed at minimizing the loss function $\mathcal{L}$, defined in Eqn (4), which is differentiable with respect to all parameters. In the early training stages, the coarse-scaled network cannot generate reasonable probability maps. To prevent the fine-scaled network from being confused by inaccurate input regions, we use the ground-truth mask $\mathbf{Y}_{\mathrm{A},l}$ as the reference map. After a sufficient number of training, we resume using $\mathbf{P}_{\mathrm{A},l}^{(t)}$ instead of $\mathbf{Y}_{\mathrm{A},l}$. In Section 4.2, we will see that this "fine-tuning" strategy improves segmentation accuracy considerably.

Due to the limitation in GPU memory, in each mini-batch containing one training sample, we set $T$ to be the maximal integer (not larger than 5) so that we can fit the entire framework into the GPU memory. The overall framework is illustrated in Figure 4. As a side note, we find that setting $T \equiv 1$ also produces high accuracy, suggesting that major improvement is brought by joint optimization.

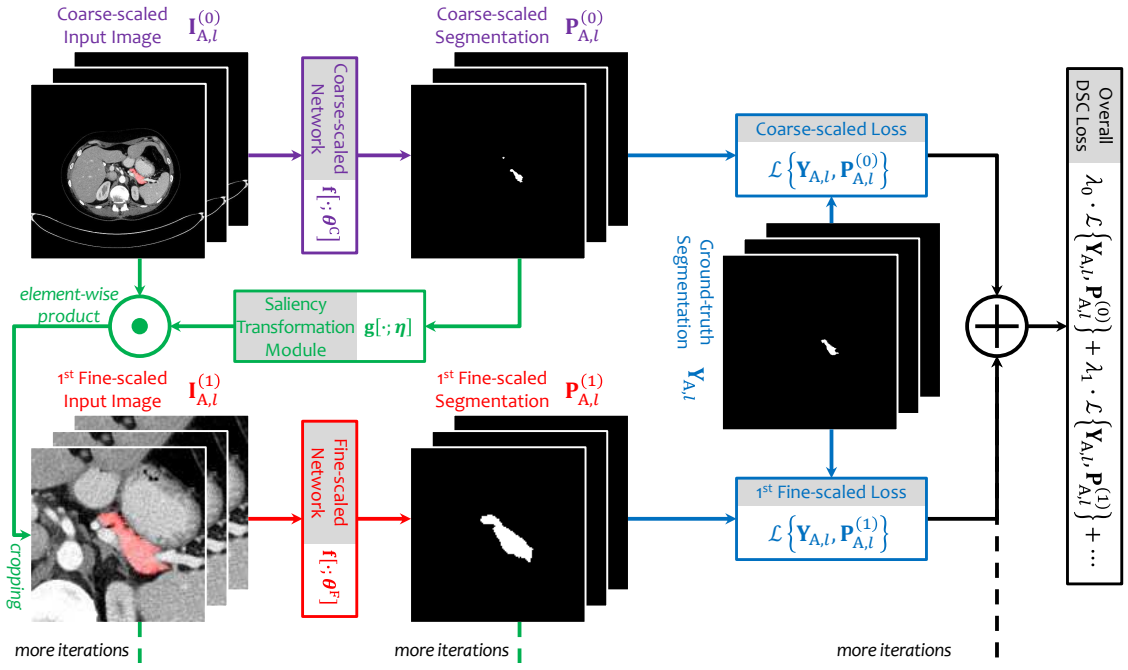


Figure 4. Illustration of the training process (best viewed in color). We display an input image along the *axial* view which contains 3 neighboring slices. To save space, we only plot the coarse stage and the first iteration in the fine stage.

**The testing phase** follows the flowchart described in Algorithm 1. There are two minor differences from the training phase. First, as the ground-truth segmentation mask $\mathbf{Y}_{\mathrm{A},l}$ is not available, the probability map $\mathbf{P}^{(t)}_{\mathrm{A},l}$ is always taken as the reference map for image cropping. Second, the number of iterations is no longer limited by the GPU memory, as the intermediate outputs can be discarded on the way. In practice, we terminate our algorithm when the similarity of two consecutive predictions, measured by $\mathrm{DSC}\{\mathbf{Z}^{(t-1)}, \mathbf{Z}^{(t)}\} = \frac{2\times\sum_i Z_i^{(t-1)} Z_i^{(t)}}{\sum_i Z_i^{(t-1)} + Z_i^{(t)}}$, reaches a threshold $\mathrm{thr}$, or a fixed number ($T$) of iterations are executed. We will discuss these parameters in Section 4.4.2.

### 3.4. Discussions

Coarse-to-fine recognition is an effective idea in medical imaging analysis. Examples include [46], our baseline, and [4] for metosis detection. Our approach can be applied to most of them towards higher recognition performance.

Attention-based or recurrent models are also widely used for natural image segmentation [6][21][42][24]. Our approach differs from them in making full use of the special properties of CT scans, *e.g.*, each organ appears at a roughly fixed position, and has a fixed number of components. Our approach can be applied to detecting the lesion areas of an organ [17][45], or a specific type of vision problems such as *hair* segmentation in a *face* [28], or detecting the targets which are consistently small in the input images [38].

## 4. Pancreas Segmentation Experiments

### 4.1. Dataset and Evaluation

We evaluate our approach on the NIH *pancreas* segmentation dataset [34], which contains 82 contrast-enhanced abdominal CT volumes. The resolution of each scan is $512 \times 512 \times L$, where $L \in [181, 466]$ is the number of slices along the long axis of the body. The distance between neighboring voxels ranges from $0.5\mathrm{mm}$ to $1.0\mathrm{mm}$.

Following the standard cross-validation strategy, we split the dataset into 4 fixed folds, each of which contains approximately the same number of samples. We apply cross validation, *i.e.*, training the models on 3 out of 4 subsets and testing them on the remaining one. We measure the segmentation accuracy by computing the Dice-Sørensen coefficient (DSC) for each sample, and report the average and standard deviation over all 82 cases.

### 4.2. Different Settings

We use the FCN-8s model [27] pre-trained on PascalVOC [9]. We initialize the up-sampling layers with random weights, set the learning rate to be $10^{-4}$ and run 80,000 iterations. Different options are evaluated, including using different kernel sizes in saliency transformation, and whether to fine-tune the models using the predicted segmentations as reference maps (see the description in Section 3.3). Quantitative results are summarized in Table 1.

As the saliency transformation module is implemented

| Model | Average | Gain | Max | Min |
|---|---|---|---|---|
| Stage-wise segmentation [46] | $82.37 \pm 5.68$ | – | 90.85 | 62.43 |
| Using $3 \times 3$ kernels in saliency transformation (basic model) | $83.47 \pm 5.78$ | $+0.00$ | 90.63 | 57.85 |
| Using $1 \times 1$ kernels in saliency transformation | $82.85 \pm 6.68$ | $-0.62$ | 90.40 | 53.44 |
| Using $5 \times 5$ kernels in saliency transformation | $83.64 \pm 5.29$ | $+0.17$ | 90.35 | 66.35 |
| Two-layer saliency transformation ($3 \times 3$ kernels) | $83.93 \pm 5.43$ | $+0.46$ | 90.52 | 64.78 |
| Fine-tuning with noisy data ($3 \times 3$ kernels) | $83.99 \pm 5.09$ | $+0.52$ | 90.57 | 65.05 |

Table 1. Accuracy (DSC, %) comparison of different settings of our approach. Please see the texts in Section 4.2 for detailed descriptions of these variants. For each variant, the "gain" is obtained by comparing its accuracy with the basic model.

by a size-preserved convolution (see Section 3.2), the size of convolutional kernels determines the range that a pixel can use to judge its saliency. In general, a larger kernel size improves segmentation accuracy ($3 \times 3$ works significantly better than $1 \times 1$), but we observe the marginal effect: the improvement of $5 \times 5$ over $3 \times 3$ is limited. As we use $7 \times 7$ kernels, the segmentation accuracy is slightly lower than that of $5 \times 5$. This may be caused by the larger number of parameters introduced to this module. Another way of increasing the receptive field size is to use two convolutional layers with $3 \times 3$ kernels. This strategy, while containing a smaller number of parameters, works even better than using one $5 \times 5$ layer. But, we do not add more layers, as the performance saturates while computational costs increase.

As described in Section 3.3, we fine-tune these models with images cropped from the coarse-scaled segmentation mask. This is to adjust the models to the testing phase, in which the ground-truth mask is unknown, so that the fine-scaled segmentation needs to start with, and be able to revise the coarse-scaled segmentation mask. We use a smaller learning rate ($10^{-6}$) and run another $40{,}000$ iterations. This strategy not only reports $0.52\%$ overall accuracy gain, but also alleviates over-fitting (see Section 4.4.3).

In summary, all these variants produce higher accuracy than the state-of-the-art ($82.37\%$ by [46]), which verifies that the major contribution comes from our recurrent framework which enables joint optimization. In the later experiments, we inherit the best variant learned from this section, including in a large-scale multi-organ dataset (see Section 5). That is to say, we use two $3 \times 3$ convolutional layers for saliency transformation, and fine-tune the models with coarse-scaled segmentation. This setting produces an average accuracy of $84.50\%$, as shown in Table 2.

### 4.3. Comparison to the State-of-the-Art

We show that our approach works better than the baseline, *i.e.*, the coarse-to-fine approach with two stages trained individually [46]. As shown in Table 2, the average improvement over $82$ cases is $2.13 \pm 2.67\%$, which is impressive given such a high baseline accuracy ($82.37\%$ is already the state-of-the-art). The standard deviations ($5.68\%$ of [46] and $4.97\%$ of ours) are mainly caused by the difference in scanning and labeling qualities. The student's $t$-test suggests statistical significance ($p = 3.62 \times 10^{-8}$). A case-by-case study reveals that our approach reports higher accuracies on $67$ out of $82$ cases, with the largest advantage being $+17.60\%$ and the largest deficit being merely $-3.85\%$. We analyze the sources of improvement in Section 4.4.

| Approach | Average | Max | Min |
|---|---|---|---|
| Roth *et al.* [34] | $71.42 \pm 10.11$ | 86.29 | 23.99 |
| Roth *et al.* [35] | $78.01 \pm 8.20$ | 88.65 | 34.11 |
| Zhang *et al.* [44] | $77.89 \pm 8.52$ | 89.17 | 43.67 |
| Roth *et al.* [36] | $81.27 \pm 6.27$ | 88.96 | 50.69 |
| Zhou *et al.* [46] | $82.37 \pm 5.68$ | 90.85 | 62.43 |
| Cai *et al.* [3] | $82.4 \pm 6.7$ | 90.1 | 60.0 |
| Our Best Model | $\mathbf{84.50} \pm 4.97$ | **91.02** | **62.81** |

Table 2. Accuracy (DSC, %) comparison between our approach and the state-of-the-arts on the NIH *pancreas* segmentation dataset [34]. [44] was implemented in [46].

Another related work is [44] which stacks two FCN's for segmentation. Our work differs from it by **(i)** our model is recurrent, which allows fine-scaled segmentation to be updated iteratively, and **(ii)** we crop the input image to focus on the salient region. Both strategies contribute significantly to segmentation accuracy. Quantitatively, [44] reported an average accuracy of $77.89\%$. Our approach achieves $78.23\%$ in the *coarse* stage, $82.73\%$ after *only one iteration*, and an entire testing phase reports $84.50\%$.

We briefly discuss the advantages and disadvantages of using 3D networks. 3D networks capture richer contextual information, but also require training more parameters. Our 2D approach makes use of 3D contexts more efficiently. At the end of each iteration, predictions from three views are fused, and thus the saliency transformation module carries these information to the next iteration. We implement VNet [30], and obtain an average accuracy of $83.18\%$ with a 3D *ground-truth* bounding box provided for each case. Without the ground-truth, a sliding-window process is required which is really slow – an average of $5$ minutes on a Titan-X Pascal GPU. In comparison, our approach needs $1.3$ minutes, slower than the baseline [46] ($0.9$ minutes), but faster than other 2D approaches [34][35] (2–3 minutes).

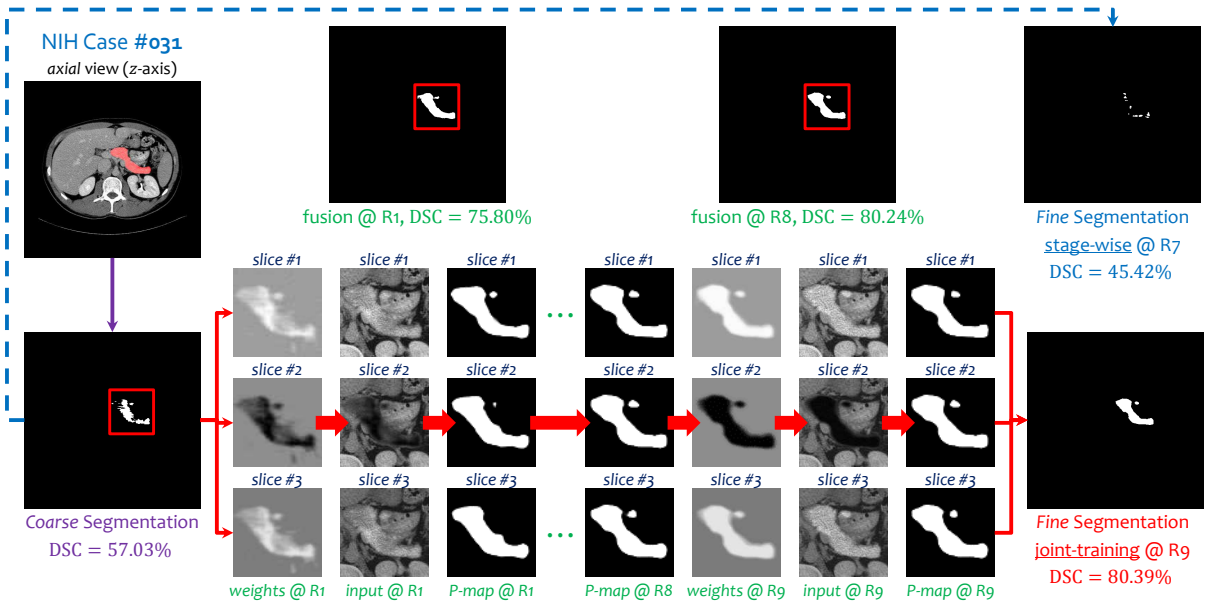


Figure 5. Visualization of how recurrent saliency transformation works in coarse-to-fine segmentation (best viewed in color). This is a failure case of the stage-wise approach [46] (see Figure 2), but segmentation accuracy is largely improved by making use of the probability map from the previous iteration to help the current iteration. Note that three weight maps capture different visual cues, with two of them focused on the foreground region, and the remaining one focused on the background region.

## 4.4. Diagnosis

### 4.4.1 Joint Optimization and Mutli-Stage Cues

Our approach enables joint training, which improves both the coarse and fine stages individually. We denote the two networks trained in [46] by $\mathbb{I}^{\mathrm{C}}$ and $\mathbb{I}^{\mathrm{F}}$, and similarly, those trained in our approach by $\mathbb{J}^{\mathrm{C}}$ and $\mathbb{J}^{\mathrm{F}}$, respectively. In the coarse stage, $\mathbb{I}^{\mathrm{C}}$ reports $75.74\%$ and $\mathbb{J}^{\mathrm{C}}$ reports $78.23\%$. In the fine stage, applying $\mathbb{J}^{\mathrm{F}}$ on top of the output of $\mathbb{I}^{\mathrm{C}}$ gets $83.80\%$, which is considerably higher than $82.37\%$ ($\mathbb{I}^{\mathrm{F}}$ on top of $\mathbb{I}^{\mathrm{C}}$) but lower than $84.50\%$ ($\mathbb{J}^{\mathrm{F}}$ on top of $\mathbb{J}^{\mathrm{C}}$). Therefore, we conclude that both the coarse-scaled and fine-scaled networks benefit from joint optimization. A stronger coarse stage provides a better starting point, and a stronger fine stage improves the upper-bound.

In Figure 5, We visualize show how the recurrent network assists segmentation by incorporating multi-stage visual cues. This is a failure case by the baseline approach [46] (see Figure 2), in which fine-scaled segmentation worked even worse because the missing contextual information. It is interesting to see that in saliency transformation, different channels deliver complementary information, *i.e.*, two of them focus on the target organ, and the remaining one adds most weights to the background region. Similar phenomena happen in the models trained in different viewpoints and different folds. This reveal that, except for foreground, background and boundary also contribute to visual recognition [47].

### 4.4.2 Convergence

We study convergence, which is a very important criterion to judge the reliability of our approach. We choose the best model reporting an average accuracy of $84.50\%$, and record the inter-iteration DSC throughout the testing process: $d^{(t)} = \mathrm{DSC}\{\mathbf{Z}^{(t-1)}, \mathbf{Z}^{(t)}\} = \frac{2\times\sum_i Z_i^{(t-1)} Z_i^{(t)}}{\sum_i Z_i^{(t-1)} + Z_i^{(t)}}$.

After 1, 2, 3, 5 and 10 iterations, these numbers are $0.9037$, $0.9677$, $0.9814$, $0.9908$ and $0.9964$ for our approach, and $0.8286$, $0.9477$, $0.9661$, $0.9743$ and $0.9774$ for [46], respectively. Each number reported by our approach is considerably higher than that by the baseline. The better convergence property provides us with the opportunity to set a more strict terminating condition, *e.g.*, using $\mathrm{thr} = 0.99$ rather than $\mathrm{thr} = 0.95$.

We note that [46] also tried to increase the threshold from $0.95$ to $0.99$, but only $3$ out of $82$ cases converged after $10$ iterations, and the average accuracy went down from $82.37\%$ to $82.28\%$. In contrary, when the threshold is increased from $0.95$ to $0.99$ in our approach, $80$ out of $82$ cases converge (in an average of $5.22$ iterations), and the average accuracy is improved from $83.93\%$ to $84.50\%$. In addition, the average number of iterations needed to achieve $\mathrm{thr} = 0.95$ is also reduced from $2.89$ in [46] to $2.02$ in our approach. On a Titan-X Pascal GPU, one iteration takes $0.2$ minutes, so using $\mathrm{thr} = 0.99$ requires an average of $1.3$ minutes in each testing case. In comparison, [46] needs an average of $0.9$ minutes and [35] needs 2-3 minutes.

| Organ | [46]-**C** | [46]-**F** | Ours-**C** | Ours-**F** |
|---|---|---|---|---|
| *adrenal g.* | 57.38 | 61.65 | 60.70 | **63.76** |
| *duodenum* | 67.42 | 69.39 | 71.40 | **73.42** |
| *gallbladder* | 82.57 | ♯82.12 | 87.08 | **87.10** |
| *inferior v.c.* | 71.77 | ♯71.15 | 79.12 | **79.69** |
| *kidney l.* | 92.56 | 92.78 | 96.08 | **96.21** |
| *kidney r.* | 94.98 | 95.39 | 95.80 | **95.97** |
| *pancreas* | 83.68 | 85.79 | 86.09 | **87.60** |

Table 3. Comparison of coarse-scaled (**C**) and fine-scaled (**F**) segmentation by [46] and our approach on our own dataset. A fine-scaled accuracy is indicated by ♯ if it is lower than the coarse-scaled one. The *pancreas* segmentation accuracies are higher than those in Table 2, due to the increased number of training samples and the higher resolution in CT scans.

#### 4.4.3 The Over-Fitting Issue

Finally, we investigate the over-fitting issue by making use of *oracle* information in the testing process. We follow [46] to use the ground-truth bounding box *on each slice*, which is used to crop the input region in *every* iteration. Note that annotating a bounding box in each slice is expensive and thus not applicable in real-world clinical applications. This experiment is aimed at exploring the upper-bound of our segmentation networks under perfect localization.

With oracle information provided, our best model reports $86.37\%$, which is considerably higher than the number ($84.50\%$) without using oracle information. If we do not fine-tune the networks using coarse-scaled segmentation (see Table 1), the above numbers are $86.26\%$ and $83.68\%$, respectively. This is to say, fine-tuning prevents our model from relying on the ground-truth mask. It not only improves the average accuracy, but also alleviates over-fitting (the disadvantage of our model against that with oracle information is decreased by $0.67\%$).

## 5. Mutli-Organ Segmentation Experiments

To verify that out approach can be applied to other organs, we collect a large dataset which contains 200 CT scans, 11 abdominal organs and 5 blood vessels. This corpus took 4 full-time radiologists around 3 months to annotate. To the best of our knowledge, this dataset is larger and contains more organs than any public datasets. We choose 5 most challenging targets including the *pancreas* and a blood vessel, as well as two *kidneys* which are relatively easier. Other easy organs such as the *liver* are ignored. To the best of our knowledge, some of these organs were never investigated before, but they are important in diagnosing pancreatic diseases and detecting the pancreatic cancer at an early stage. We randomly partition the dataset into 4 folds for cross validation. Each organ is trained and tested individually. When a pixel is predicted as more than one organs, we choose the one with the largest confidence score.

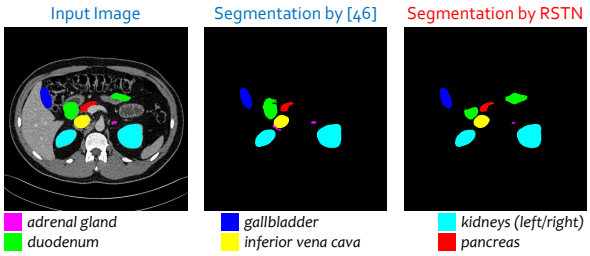


Figure 6. Mutli-organ segmentation in the *axial* view (best viewed in color). Organs are marked in different colors (input image is shown with the ground-truth annotation).

Results are summarized in Table 3, We first note that [46] sometimes produced a lower accuracy in the fine stage than in the coarse stage. Apparently this is caused by the unsatisfying convergence property in iterations, but essentially, it is the loss of contextual information and the lack of globally optimized energy function. Our approach solves this problem and reports a $4.29\%$ average improvement over 5 challenging organs (the *kidneys* excluded). For some organs, *e.g.*, the *gallbladder*, we do not observe significant accuracy gain by iterations. But we emphasize that in these scenarios, our coarse stage already provides much higher accuracy than the fine stage of [46], and the our fine stage preserves such high accuracy through iterations, demonstrating stability. An example is displayed in Figure 6.

## 6. Conclusions

This work is motivated by the difficulty of small organ segmentation. As the target is often small, it is required to focus on a local input region, but sometimes the network is confused due to the lack of contextual information. We present the **Recurrent Saliency Transformation Network**, which enjoys three advantages. **(i)** Benefited by a (recurrent) global energy function, it is easier to generalize our models from training data to testing data. **(ii)** With joint optimization over two networks, both of them get improved individually. **(iii)** By incorporating multi-stage visual cues, more accurate segmentation results are obtained. As the fine stage is less likely to be confused by the lack of contexts, we also observe better convergence during iterations.

Our approach is applied to two datasets for *pancreas* segmentation and multi-organ segmentation, and outperforms the baseline (the state-of-the-art) significantly. Confirmed by the radiologists in our team, these segmentation results are helpful to computer-assisted clinical diagnoses.

**Acknowledgements:** This paper was supported by the Lustgarten Foundation for Pancreatic Cancer Research. We thank Wei Shen, Seyoun Park, Weichao Qiu, Song Bai, Zhuotun Zhu, Chenxi Liu, Yan Wang, Siyuan Qiao, Yingda Xia and Fengze Liu for discussions.

## References


[1] A. Ali, A. Farag, and A. El-Baz. Graph Cuts Framework for Kidney Segmentation with Prior Shape Constraints. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2007.

[2] T. Brosch, L. Tang, Y. Yoo, D. Li, A. Traboulsee, and R. Tam. Deep 3D Convolutional Encoder Networks with Shortcuts for Multiscale Feature Integration Applied to Multiple Sclerosis Lesion Segmentation. *IEEE Transactions on Medical Imaging*, 35(5):1229–1239, 2016.

[3] J. Cai, L. Lu, Y. Xie, F. Xing, and L. Yang. Improving Deep Pancreas Segmentation in CT and MRI Images via Recurrent Neural Contextual Learning and Direct Loss Function. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2017.

[4] H. Chen, Q. Dou, X. Wang, J. Qin, and P. Heng. Mitosis Detection in Breast Cancer Histology Images via Deep Cascaded Networks. *AAAI Conference on Artificial Intelligence*, 2016.

[5] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. *International Conference on Learning Representations*, 2015.

[6] L. Chen, Y. Yang, J. Wang, W. Xu, and A. Yuille. Attention to Scale: Scale-aware Semantic Image Segmentation. *Computer Vision and Pattern Recognition*, 2016.

[7] C. Chu, M. Oda, T. Kitasaka, K. Misawa, M. Fujiwara, Y. Hayashi, Y. Nimura, D. Rueckert, and K. Mori. Multi-organ Segmentation based on Spatially-Divided Probabilistic Atlas from 3D Abdominal CT Images. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2013.

[8] Q. Dou, H. Chen, Y. Jin, L. Yu, J. Qin, and P. Heng. 3D Deeply Supervised Network for Automatic Liver Segmentation from CT Volumes. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2016.

[9] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. *Computer Vision and Pattern Recognition*, 2014.

[11] A. Graves, A. Mohamed, and G. Hinton. Speech Recognition with Deep Recurrent Neural Networks. *International Conference on Acoustics, Speech and Signal Processing*, 2013.

[12] A. Harrison, Z. Xu, K. George, L. Lu, R. Summers, and D. Mollura. Progressive and Multi-Path Holistically Nested Neural Networks for Pathological Lung Segmentation from CT Images. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2017.

[13] M. Havaei, A. Davy, D. Warde-Farley, A. Biard, A. Courville, Y. Bengio, C. Pal, P. Jodoin, and H. Larochelle. Brain Tumor Segmentation with Deep Neural Networks. *Medical Image Analysis*, 2017.

[14] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. *Computer Vision and Pattern Recognition*, 2016.

[15] T. Heimann, B. Van Ginneken, M. Styner, Y. Arzhaeva, V. Aurich, C. Bauer, A. Beck, C. Becker, R. Beichel, G. Bekes, et al. Comparison and Evaluation of Methods for Liver Segmentation from CT Datasets. *IEEE Transactions on Medical Imaging*, 28(8):1251–1265, 2009.

[16] S. Hu, E. Hoffman, and J. Reinhardt. Automatic Lung Segmentation for Accurate Quantitation of Volumetric X-ray CT Images. *IEEE Transactions on Medical Imaging*, 20(6):490–498, 2001.

[17] K. Kamnitsas, C. Ledig, V. Newcombe, J. Simpson, A. Kane, D. Menon, D. Rueckert, and B. Glocker. Efficient Multi-Scale 3D CNN with Fully Connected CRF for Accurate Brain Lesion Segmentation. *Medical Image Analysis*, 36:61–78, 2017.

[18] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 2012.

[19] J. Kuen, Z. Wang, and G. Wang. Recurrent Attentional Networks for Saliency Detection. *Computer Vision and Pattern Recognition*, 2016.

[20] M. Lai. Deep Learning for Medical Image Segmentation. *arXiv preprint arXiv:1505.02000*, 2015.

[21] G. Li, Y. Xie, L. Lin, and Y. Yu. Instance-Level Salient Object Segmentation. *Computer Vision and Pattern Recognition*, 2017.

[22] M. Liang and X. Hu. Recurrent Convolutional Neural Network for Object Recognition. *Computer Vision and Pattern Recognition*, 2015.

[23] D. Lin, C. Lei, and S. Hung. Computer-Aided Kidney Segmentation on Abdominal CT Images. *IEEE Transactions on Information Technology in Biomedicine*, 10(1):59–65, 2006.

[24] G. Lin, A. Milan, C. Shen, and I. Reid. RefineNet: Multi-Path Refinement Networks with Identity Mappings for High-Resolution Semantic Segmentation. *Computer Vision and Pattern Recognition*, 2017.

[25] H. Ling, S. Zhou, Y. Zheng, B. Georgescu, M. Suehling, and D. Comaniciu. Hierarchical, Learning-based Automatic Liver Segmentation. *Computer Vision and Pattern Recognition*, 2008.

[26] M. Linguraru, J. Sandberg, Z. Li, F. Shah, and R. Summers. Automated Segmentation and Quantification of Liver and Spleen from CT Images Using Normalized Probabilistic Atlases and Enhancement Estimation. *Medical Physics*, 37(2):771–783, 2010.

[27] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. *Computer Vision and Pattern Recognition*, 2015.

[28] L. Luo, H. Li, and S. Rusinkiewicz. Structure-Aware Hair Capture. *ACM Transactions on Graphics*, 32(4):76, 2013.

[29] J. Merkow, D. Kriegman, A. Marsden, and Z. Tu. Dense Volume-to-Volume Vascular Boundary Detection. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2016.

[30] F. Milletari, N. Navab, and S. Ahmadi. V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. *International Conference on 3D Vision*, 2016.

[31] P. Pinheiro and R. Collobert. Recurrent Convolutional Neural Networks for Scene Labeling. *International Conference on Machine Learning*, 2014.

[32] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks. *Advances in Neural Information Processing Systems*, 2015.

[33] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015.

[34] H. Roth, L. Lu, A. Farag, H. Shin, J. Liu, E. Turkbey, and R. Summers. DeepOrgan: Multi-level Deep Convolutional Networks for Automated Pancreas Segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2015.

[35] H. Roth, L. Lu, A. Farag, A. Sohn, and R. Summers. Spatial Aggregation of Holistically-Nested Networks for Automated Pancreas Segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2016.

[36] H. Roth, L. Lu, N. Lay, A. Harrison, A. Farag, A. Sohn, and R. Summers. Spatial Aggregation of Holistically-Nested Convolutional Neural Networks for Automated Pancreas Localization and Segmentation. *arXiv preprint arXiv:1702.00045*, 2017.

[37] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. *International Conference on Learning Representations*, 2015.

[38] S. Singh, D. Hoiem, and D. Forsyth. Learning to Localize Little Landmarks. *Computer Vision and Pattern Recognition*, 2016.

[39] R. Socher, C. Lin, C. Manning, and A. Ng. Parsing Natural Scenes and Natural Language with Recursive Neural Networks. *International Conference on Machine Learning*, 2011.

[40] D. Wang, A. Khosla, R. Gargeya, H. Irshad, and A. Beck. Deep Learning for Identifying Metastatic Breast Cancer. *arXiv preprint arXiv:1606.05718*, 2016.

[41] Z. Wang, K. Bhatia, B. Glocker, A. Marvao, T. Dawes, K. Misawa, K. Mori, and D. Rueckert. Geodesic Patch-based Segmentation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2014.

[42] F. Xia, P. Wang, L. Chen, and A. Yuille. Zoom Better to See Clearer: Human and Object Parsing with Hierarchical Auto-Zoom Net. *European Conference on Computer Vision*, 2016.

[43] L. Yu, X. Yang, H. Chen, J. Qin, and P. Heng. Volumetric ConvNets with Mixed Residual Connections for Automated Prostate Segmentation from 3D MR Images. *AAAI Conference on Artificial Intelligence*, 2017.

[44] Y. Zhang, M. Ying, L. Yang, A. Ahuja, and D. Chen. Coarse-to-Fine Stacked Fully Convolutional Nets for Lymph Node Segmentation in Ultrasound Images. *IEEE International Conference on Bioinformatics and Biomedicine*, 2016.

[45] Y. Zhou, L. Xie, E. Fishman, and A. Yuille. Deep Supervision for Pancreatic Cyst Segmentation in Abdominal CT Scans. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2017.

[46] Y. Zhou, L. Xie, W. Shen, Y. Wang, E. Fishman, and A. Yuille. A Fixed-Point Model for Pancreas Segmentation in Abdominal CT Scans. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2017.

[47] Z. Zhu, L. Xie, and A. Yuille. Object Recognition with and without Objects. *International Joint Conference on Artificial Intelligence*, 2017.