A Joint Deep Learning Approach for Automated Liver and Tumor Segmentation

Nadja Gruber*, Stephan Antholzer*, Werner Jaschke†, Christian Kremser† and Markus Haltmeier*
* Department of Mathematics, University of Innsbruck, Technikerstraße 13, A-6020 Innsbruck.
Email: {nadja.gruber,stephan.antholzer,markus.haltmeier}@uibk.ac.at
† Department of Radiology, Medical University of Innsbruck, Anichstraße 35, 6020 Innsbruck, Austria

***Abstract*—Hepatocellular carcinoma (HCC) is the most common type of primary liver cancer in adults, and the most common cause of death of people suffering from cirrhosis. The segmentation of liver lesions in CT images allows assessment of tumor load, treatment planning, prognosis and monitoring of treatment response. Manual segmentation is a very time-consuming task and, in many cases, prone to inaccuracies. Therefore, automatic tools for tumor detection and segmentation are highly desirable. We propose a network architecture that consists of two consecutive nested fully convolutional neural networks together with a joint minimization strategy. The first sub-network segments the liver whereas the second sub-network segments the actual tumor inside the liver. We compare the nested network architecture to a one-step approach, where a neural network performs both segmentation tasks simultaneously. Both architectures are trained on a subset of the LiTS (Liver Tumor Segmentation) Challenge and evaluated on data provided from the radiological center in Innsbruck. The nested approach is shown to significantly outperform the one-step network in terms of various accuracy measures.**

I. Introduction

Liver cancer remains associated with a high mortality rate, in part related to initial diagnosis at an advanced stage of disease. Prospects can be significantly improved by earlier treatment beginning, and analysis of CT images is a main diagnostic tool for early detection of liver tumors. Manual inspection and segmentation is a labor- and time-intensive process yielding relatively imprecise results in many cases. Thus, there is significant interest in developing automated strategies to support the early detection of lesions. Due to complex backgrounds, significant variations in location, shape and intensity across different patients, both, the automated liver segmentation and the further detection of tumors, remain challenging tasks.

Semantic segmentation of CT images has been an active area of research over the past few years. Recent developments of deep learning have dramatically improved the performance of artificial intelligence. Deep learning algorithms, especially deep convolutional neural networks (CNN) have considerably outperformed their competitors in medical imaging. One of the most successful CNN architectures is the so-called U-Net [1], which has won several competitions in the field of biomedical image segmentation.

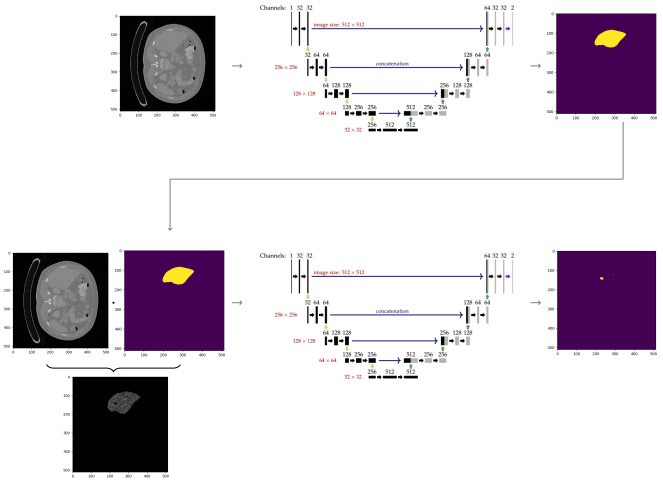

Figure I.1: **Illustration of the nested network architecture for automated semantic liver and tumor segmentation.** The model consists of two sequential U-Nets. The raw images are fed into the first network, and the output is a binary image. The original image multiplied by the obtained liver mask represents the input of the second U-Net. The final output is a binary image in which label 1 is assigned to tumor.

We investigate a deep learning strategy that jointly segments the liver and the lesions in CT images. Similar to [2], we use a network architecture that is formed of two consecutive U-Nets; see for related FCN architectures [3]–[5]. The first sub-network performs liver segmentation, while the second one incorporates the output of the first network and segments the lesion. We propose a joint weighted loss function combining the outputs of both networks (Figure I.1). The network is trained on a subset of the LiTS (Liver Tumor Segmentation Challenge) and evaluated on different data collected at the radiological center in Innsbruck. For our initial experiments, we perform consecutive training, with which we already obtain quite accurate results. For comparison purpose, we also implement a one-step approach, where a single multi-class

network is used for simultaneous classification of background, liver and tumor (Figure I.2). As the main finding of our work, we demonstrate that the nested approach significantly outperforms the one-step approach in terms of various accuracy measures. Moreover, our finding indicates that both networks are quite robust in the sense that even when trained on one dataset (LiTS Challenge), they well predict tumors when evaluated on different dataset (from radiological center Innsbruck).

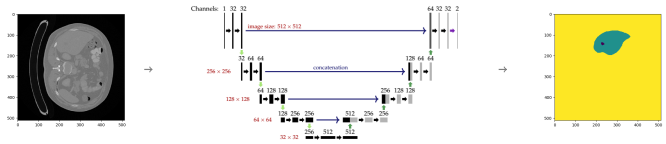

Figure I.2: Illustration of network architecture for automated liver and tumor segmentation executed in one step. The final output is a discrete class label output in which liver corresponds to label 1, tumor to label 2 and background to label 0.

II. Joint Liver and Tumor Segmentation

For the following, let $\{X_1, \dots, X_N\} \subseteq \mathbb{R}^{512\times512}$ and $\{Y_1, \dots, Y_N\} \subseteq \{0,1,2\}^{512\times512}$ denote the set of training images and the corresponding segmented images, respectively. Here, the label 1 stands for liver, 2 for tumor and 0 for background.

A. Nested Deep Learning Approach

For the task of semantic liver and tumor segmentation, we generate segmentation masks

$$\{A_1, \dots, A_N\} \subseteq \{0,1\}^{512\times512}$$
$$\{B_1, \dots, B_N\} \subseteq \{0,1\}^{512\times512}$$

representing binary images A_k where class label 1 stands for the liver or tumor, and binary images B_k where class label 1 stands for tumor.

Our nested approach is to train two networks

$$\mathbb{A}_\theta \colon \mathbb{R}^{512\times512} \to [0,1]^{512\times512}$$
$$\mathbb{B}_\eta \colon \mathbb{R}^{512\times512} \to [0,1]^{512\times512}$$

that separately perform liver and tumor segmentation. In the fist step, the network $\mathbb{A}_\theta$ is applied such that $\mathbb{A}_\theta(X_k) \simeq A_k$. After decision making by selecting a threshold $t_a \in (0,1)$, we obtain a liver mask $\mathbb{M}_\theta \colon \mathbb{R}^{512\times512} \to \{0,1\}^{512\times512}$ that is applied to each input image. Additionally, we applied windowing w pointwise to the intensity values, which results in new training data

$$\bar{X}_k = w\big(\mathbb{M}_\theta(X_k)X_k\big)$$
$$\bar{B}_k = \mathbb{M}_\theta(X_k)B_k\,.$$

These data serve as input and corresponding ground truth for training the second network $\mathbb{B}_\eta$. By selecting another threshold, a mask $\mathbb{T}_\eta$ for the tumors is given.

The final classification can be performed in assigning a pixel (i,j) to class label 2 if $\mathbb{M}_\theta = \mathbb{T}_\eta = 1$, to class label 1 if $\mathbb{M}_\theta = 1$ and $\mathbb{T}_\eta = 0$, and class label 0 otherwise. The goal is to find the high dimensional parameter vectors θ and η such that the overall classification error is small. This is achieved by minimizing a loss function that describes how well the network performs on the training data. Here we propose to use the joint loss function

$$\mathcal{L}(\theta,\eta) = \frac{c}{N}\sum_{k=1}^{N} L\big(\mathbb{A}_\theta(X_k), A_k\big) + \frac{1-c}{N}\sum_{k=1}^{N} L\big(\mathbb{B}_\eta(w\big(\mathbb{M}_\theta(X_k)X_k\big)), \mathbb{M}_\theta(X_k)B_k\big)\,, \quad \text{(II.1)}$$

where L denotes the categorical cross-entropy-loss and the constant c weights the importance of the two classification outcomes. It has been demonstrated in [6] that a joint loss function can improve results compared to sequential approaches for joint image reconstruction and segmentation.

B. Employed U-Net Architecture

We implement the nested model using two U-Nets $\mathbb{A}_\theta$, $\mathbb{B}_\eta$, one on top of the other. The combined network architecture is shown in Figure I.1. The inputs for both CNNs are grey-scale images of size $512 \times 512 \times 1$ and their outputs are binary images of size 512×512. While the input of the first U-Net is of the form displayed in Figure III.1, the input of the second U-Net is produced by the output of the first one as explained in Section II-A.

In both networks, the input passes through an initial convolution layer and is then processed by a sequence of convolution blocks at decreasing resolutions (contracting path). The expanding path of the U-Net then reverses this downsampling process. Skip connections between down- and upsampling path intend to provide local information to the global information while upsampling. As final step the output of the network is passed to a linear classifier that outputs (via sigmoid) a probability for each pixel being within the liver/tumor. The model is implemented in Keras[1] with the TensorFlow backend[2].

C. Sequential Optimization

While in future work we will jointly minimize (II.1), for our initial studies presented here we train the networks sequentially. This means that first we optimize for θ and then use the output of $\mathbb{A}_\theta$ as input for $\mathbb{B}_\eta$. Specifically, for training the second U-net we minimize

$$\mathcal{L}_{\mathbb{B}}(\theta,\eta) = -\frac{1}{N}\sum_{k=1}^{N}\Big[\sum_{i,j=1}^{512} \alpha\mathbb{1}_{\{(a,b)|\bar{B}_k^{a,b}=0\}}(i,j)\log\big(\mathbb{B}_\eta(X_k)^{i,j}\big) + (1-\alpha)\mathbb{1}_{\{(a,b)|\bar{B}_k^{a,b}=1\}}(i,j)\log\big(1-\mathbb{B}_\eta(X_k)^{i,j}\big)\Big]. \quad \text{(II.2)}$$

[1] https://keras.io/
[2] https://www.tensorflow.org/

Here $\bar{B}_k^{i,j}$ is the value of $\bar{B}_k$ at pixel (i,j), and the indicator function $\mathbb{1}$ declares whether (i,j) belongs to the class tumor or not. The weight $\alpha \in (0,1)$ controls the relative importance assigned to the two classes. The best results are achieved by applying balanced loss, where the constant α is replaced by the weights of the form

$$\alpha_k = 1 - \frac{|\{(a,b) \mid B_k^{a,b} = 1\}|}{|B_k|} \tag{II.3}$$

for $k \in 1, \ldots, N$. Here $|\cdot|$ is used to count the number of elements in some set.

Both models have been trained using stochastic gradient decent with momentum for 300 and 600 epochs, respectively. Each iteration takes about 70 seconds on NVIDIA standard GPU. To avoid overfitting, we applied a dropout of 0.4 in the upsampling path. Both U-Nets were trained with a learning rate of 0.001 and categorical cross-entropy loss. Since the tumor area only accounts for a small area compared to the full size of the image, we applied balanced loss (II.2) in a second optimization of the network and reduced the learning rate to 0.0001. Comparison with the joint loss (II.1) is subject of future work.

D. One-Step Approach

For comparison purpose we also use a basic one-step approach, whose workflow is visualized in Figure I.2. In this context, the segmentation task is regarded as multi-class label classification whereas each pixel is assigned a certain probability of belonging to class liver, tumor or background. For that purpose we generate three binary masks

$$\{(C_1)_0, \ldots, (C_N)_0\} \subseteq \{0,1\}^{512\times512}$$
$$\{(C_1)_1, \ldots, (C_N)_1\} \subseteq \{0,1\}^{512\times512}$$
$$\{(C_1)_2, \ldots, (C_N)_2\} \subseteq \{0,1\}^{512\times512}$$

indicating whether or not a pixel corresponds to class liver, tumor or background, respectively. We then set up a single U-net architecture with three output channels,

$$\mathbb{C}_\xi \colon \mathbb{R}^{512\times512} \to [0,1]^{512\times512\times3} \ ,$$

with $\mathbb{C}_\xi(X_k)_c \simeq (C_k)_c$ for $k = 1, \ldots, N$ and $c = 0, 1, 2$.

The one-step architecture is pre-trained for 50 epochs applying categorical-cross-entropy loss (similar to (II.2)) and fine-tuned for further 30 epochs using the balanced version of the loss (similar to (II.3)). Balanced loss proved very useful in detecting the lesion for both methods.

III. Experimental Results

A. Datasets

The network training is run using a subset of the publicly available LiTS-Challenge[3]dataset containing variable kinds of liver lesions (HCC, metastasis, ...). The dataset consists of CT scans coming from different clinical institutions. Trained radiologists have manually segmented annotation of the liver and tumors. All of the volumes were enhanced with a contrast agent, imaged in the portal venous phase. Each volume contains a variable number of axial slices with a resolution of 512×512 pixels and an approximate slice thickness ranging from 0.7 to 5 mm. The training is applied on 765 axial slices, 50 are used for validation and 50 for testing.

Further test data is provided by radiological center at the medical university of Innsbruck. The dataset contains CT scans of patients suffering from HCC and the belonging reference annotations were drafted by medical scientists. Because deep learning algorithms achieve better performance if the data has a consistent scale or distribution, all data are standardized to have intensity values between $[0,1]$ before starting the optimization.

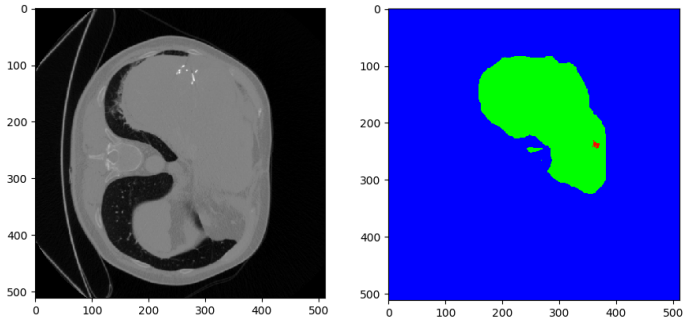

Figure III.1: Training data provided by LiTS-challenge.

B. Evaluation on Test Data

Each pixel of the image is assigned to one of the two classes liver/other tissue and tumor/other tissue, respectively, with a certain probability. Results of the automated liver and tumor segmentation are visualized in Figure III.2. Comparison with ground truth and segmented liver and tumor give rise to the assumption that our approach is highly promising for obtaining high performance metrics.

To qualitatively evaluate performance, we applied some of the commonly used evaluation metrics in semantic image segmentation.

- AUC metric: Area under ROC Curve (AUC) is a performance metric for binary classification problems. We applied ROC analysis to find the threshold that achieves the best results for the tumor segmentation task. Due to the very low rate of false classified pixels (most of them has probability close to one or close to zero), we decided to restrict the ROC curve to pixels whose probability for belonging to class tumor lies between 0.01 and 0.99.
  In Figure III.4 we can see that the best restricted AUC value (rAUC) conducting 0.88 is achieved by applying balanced loss. We further calculated the corresponding threshold and could achieve an improvement of the tumor segmentation results [7].

[3]https://competitions.codalab.org/competitions/17094

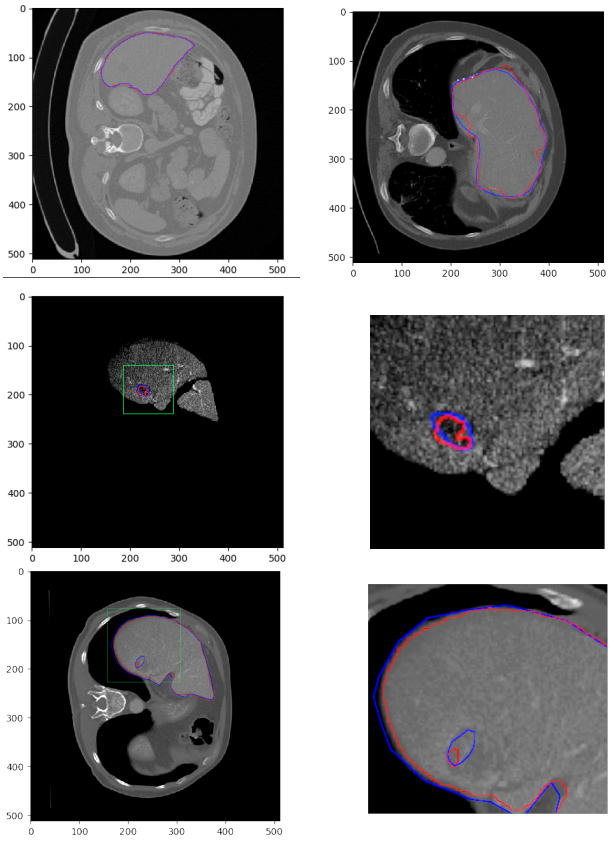

Figure III.2: **Results on HCC data. Top**: liver segmentation results (red) compared to ground truth boundary (blue). The left image pertains to the LiTS-Challenge dataset, the right one is part of the test set from Innsbruck. **Second row**: tumor segmentation result (red) compared to ground truth (blue) of radiological center in Innsbruck resulting from the nested network approach described in Section II-B. **Bottom**: Segmentation maps preserved by applying II-D.

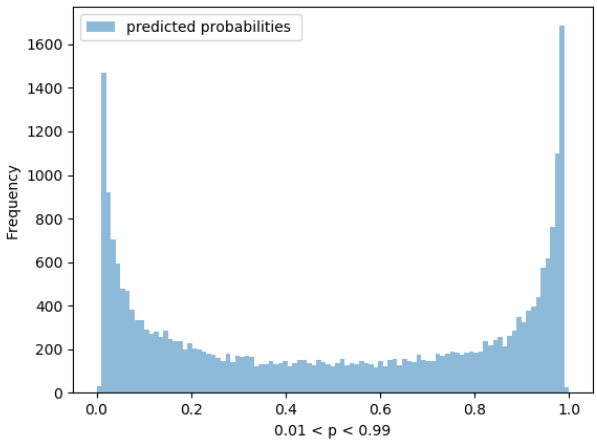

Figure III.3: Histogram that displays the number of pixels predicted falling into class tumor with probability $p \in (0.01, 0.99)$ (predictions made by the nested network).

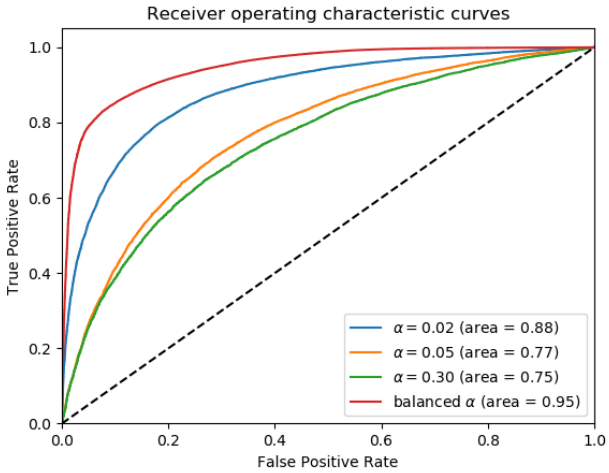

Figure III.4: Restricted ROC curves for varying weights α of weighted and balanced loss. The red curve corresponds to the outcomes produced by applying balanced loss, which apparently leads to the best tumor segmentation results. In general terms it can be stated that setting the importance of the background pixels lower seems to considerably improve segmentation accuracy of the lesion.

- PIXEL ACCURACY: With $\text{Pixel}_{\text{acc}}$ we denote the fraction of correctly classified pixels.
- INTERSECTION OVER UNION: For a more complete evaluation of the segmentation results we use class accuracy in conjunction with the so called IoU metric. The latter is essentially a method to quantify the percent overlap between the ground truth and the prediction output. The IoU measure gives the similarity between predicted and ground-truth regions for the object of interest. The formula for quantifying the IoU score is:

$$\text{IoU} = \frac{\text{TP}}{\text{FP} + \text{TP} + \text{FN}} \tag{III.1}$$

  where TP, FP and FN denote the True Positive Rate, False Positive Rate and False Negative Rate, respectively.
- RAND INDEX: Since the segmentation task can be regarded as clustering of pixels, Rand index [8], which is a measure of the similarity between two data clusterings, has been proposed as a measure of segmentation performance. Small differences in the location of object boundaries will increase the rand error slightly while merging or splitting of objects leads to a big increase of the Rand error.

The evaluation metrics are summarized in Table IV.1. The liver segmentation evaluation scores indicate that our models perform remarkable good, provided that the nested network outperforms the one-step method primarily in the

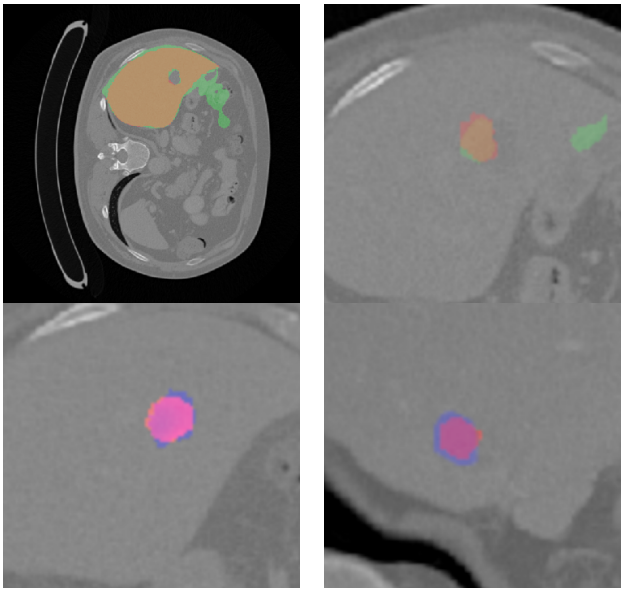

Figure III.5: **Top:** Intersection over Union of liver and tumor segmentation resulting from the one-step model. The green area indicates the predicted masks, the orange area is their overlap and the share highlighted in red, is the ground truth. **Bottom:** Intersection over Union of tumor segmentation for balanced loss with balanced α resulting from the nested network. The light pink, light blue and pink areas mark the prediction mask, ground truth and Intersection over Union, respectively.

tumor segmentation task. Pixel accuracy, Intersection over union (IoU) and Rand Index (RI) have values very close to one. IoU and Rand Index performance score of the tumor segmentation show that the application of balanced loss with achieves the best results.

IV. Conclusions

In this paper, we proposed a joint deep learning framework for the automated joint liver and tumor (and background) segmentation using a nested network architecture and a joint loss function. We compare the nested network with a one-step segmentation approach that simultaneously segments into the three classes. Metrics to evaluate the segmentation of detected lesions are comprised of a restricted AUC, an overlap Index (IoU) and Rand Index (RI). Even when the nested model is trained sequentially, it clearly outperforms the one-step model. The one-step network approach works fast but is prone to misclassification, especially in the tumor segmentation task. Future work will be done to develop an accurate minimization strategy for the joint loss function in (II.1). Another interesting topic to address is the classification of the tumors detected by a deep learning algorithm.

	α	rAUC	$\text{Pixel}_{\text{acc}}$	IoU	RI
One-Step					
Liver			0.9935	0.8898	0.9316
Tumor	bal	0.87	0.9995	0.6782	0.8075
Sequential					
Liver			0.9999	0.9385	0.9628
Tumor	0.02	0.88	0.9996	0.77108	0.8706
	0.05	0.77	0.9996	0.7261	0.8490
	0.30	0.75	0.9995	0.73879	0.8433
	bal	0.95	0.9997	0.7917	0.9106

Table IV.1: Performance evaluation metrics for tumor segmentation models applied on test data (112 images).

References

[1] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[2] P. F. Christ, F. Ettlinger *et al.*, "Automatic liver and tumor segmentation of ct and mri volumes using cascaded fully convolutional neural networks," *arXiv:1702.05970*, 2017.

[3] E. Vorontsov, A. Tang, C. Pal, and S. Kadoury, "Liver lesion segmentation informed by joint liver segmentation," in *Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on*. IEEE, 2018, pp. 1332–1335.

[4] X. Han, "Automatic liver lesion segmentation using a deep convolutional neural network method," *arXiv:1704.07239*, 2017.

[5] G. Chlebus, A. Schenk, J. H. Moltz, B. van Ginneken, H. K. Hahn, and H. Meine, "Deep learning based automatic liver tumor segmentation in ct with shape-based post-processing," 2018.

[6] J. Adler, S. Lunz, O. Verdier, C.-B. Schönlieb, and O. Öktem, "Task adapted reconstruction for inverse problems," *arXiv:1809.00948*, 2018.

[7] R. Kumar and A. Indrayan, "Receiver operating characteristic (roc) curve for medical researchers," *Indian pediatrics*, vol. 48, no. 4, pp. 277–287, 2011.

[8] V. Jain, B. Bollmann, M. Richardson, D. R. Berger, M. N. Helmstaedter, K. L. Briggman, W. Denk, J. B. Bowden, J. M. Mendenhall, W. C. Abraham *et al.*, "Boundary learning by optimization with topological constraints," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 2488–2495.