Computer Aided Medical Procedures

Uncertainty-based graph convolutional networks for organ segmentation refinement

Roger D. Soberanis-Mukul[1], Nassir Navab[1,2], Shadi Albarqouni[1,3]

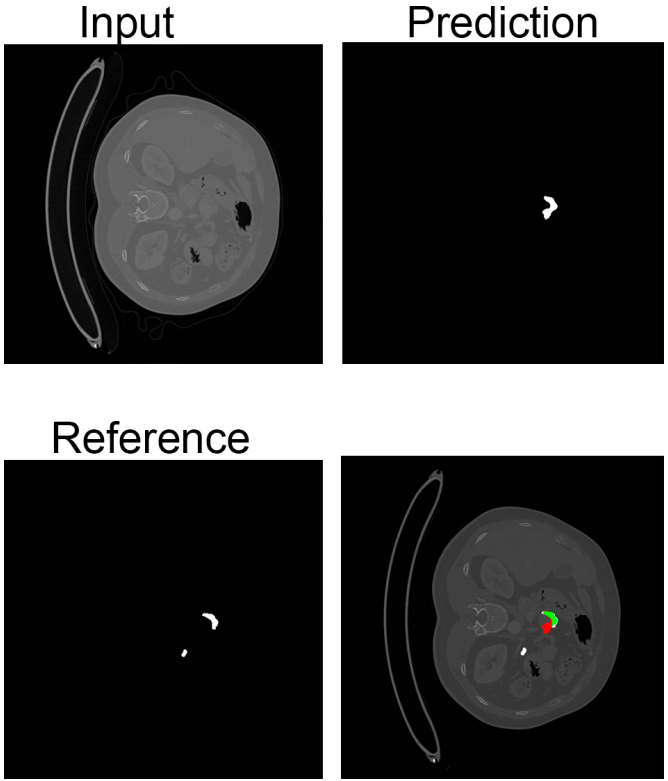
[1]Computer Aided Medical Procedures, Technische Universität München, Germany
[2]Computer Aided Medical Procedures, Johns Hopkins University, Baltimore, USA
[3]Computer Vision Laboratory, ETH Zurich, Switzerland
roger.soberanis@tum.de

Technische Universität München

JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

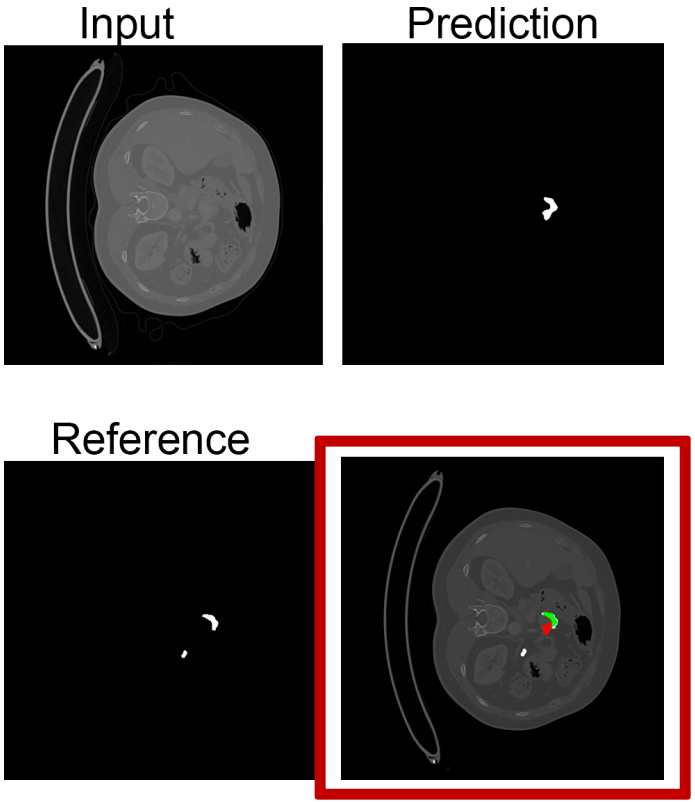
Motivation

- Segmentation of anatomical structures is an important step in many computer-aided procedures.
- Deep convolutional networks (CNN) are the current state of the art in this problem.
- Inter-patient variability and similarity between organs and background can lead to errors in the segmentation process.
- Refinement strategies are desirable.



Motivation

- Segmentation of anatomical structures is an important step in many computer-aided procedures.
- Deep convolutional networks (CNN) are the current state of the art in this problem.
- Inter-patient variability and similarity between organs and background can lead to errors in the segmentation process.
- Refinement strategies are desirable.



Motivation

Refinement strategies have been also applied as an intermediate step in semi-supervised learning problems[1]

Conditional random field (CRF[2]) is commonly used as post-processing step for refining output[1,3].

- Retraining is not necessary.
- All the information comes from CNN's output
- Based on networks predictions, spatial and intensity relationships.

[1] Wenjia Bai, Ozan Oktay, Matthew Sinclair, Hideaki Suzuki, Martin Rajchl, Giacomo Tarroni, Ben Glocker, Andrew King, Paul M. Matthews, and Daniel Rueckert. Semi- supervised learning for network-based cardiac mr image segmentation. MICCAI 2017.
[2]Philip Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. NIPS 2011.
[3]Guotai Wang, Wenqi Li, Maria A. Zuluaga, Rosalind Pratt, Premal A. Patel, Michael Aertsen, Tom Doel, Anna L. David, Jan Deprest, Sebastien Ourselin, and Tom Vercauteren. Interactive medical image segmentation using deep learning with image-specific fine tuning. IEEE TMI 2018.

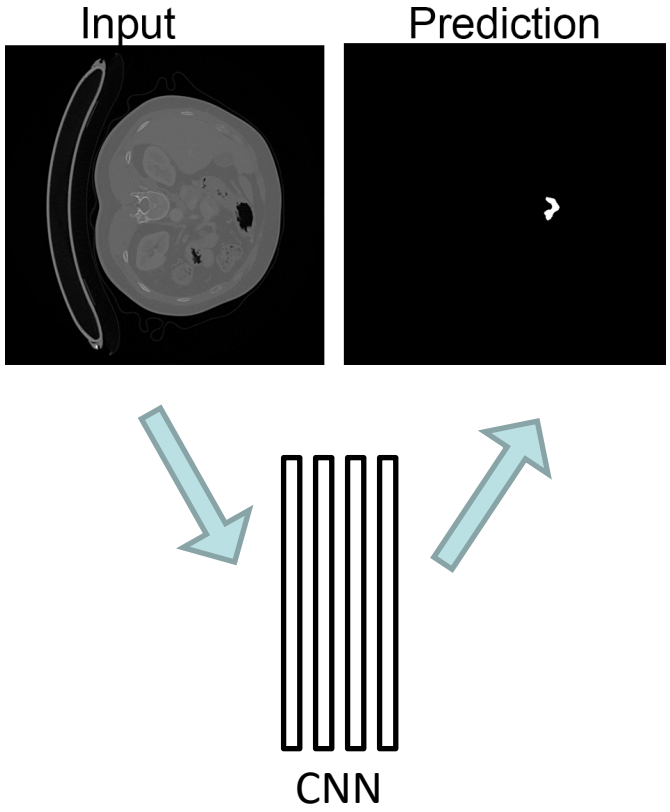
Motivation

CRF is constructed based on the CNN's prediction. However, some elements of the prediction could be incorrect (but we do not know which ones).

Information about prediction's correctness can be helpful for a refinement strategy.

Since in inference time the only information available is the input, the model, and the prediction,

How can we estimate the correctness of the CNN's prediction?



Motivation

Uncertainty estimations

- Gal shows that a deep model with dropout applied is equivalent to a Bayesian Model[4]
- Uncertainty regions can give highlights in quality and potential errors in the segmentation results[5,6,7]
- Monte Carlo dropout[4] (MCDO) strategy estimates uncertainty with no modifications to the network

We can use the uncertainty of the model to find potentially correct/incorrect points.

How can we use the uncertainty estimation to refine the CNN's prediction?

[4]Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. ICML 2016.
[5] Philipe Ambrozio Dias and Henry Medeiros. Semantic segmentation refinement by monte carlo region growing of high confidence detections. ACCV 2019.
[6]Abhijit Guha Roy, Sailesh Conjeti, Nassir Navab, and Christian Wachinger. Inherent brain segmentation quality control from fully convnet monte carlo ampling. MICCAI 2018.
[7]Tanya Nair, Doina Precup, Douglas L. Arnold, and Tal Arbel. Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation. MICCAI 2018.

Motivation

We can use the uncertainty estimation to define confident and unconfident points. Using a graph-like representation of our data, we can use the confidence information to define a partially labeled graph.

Semi-supervised graph convolutional neural networks (GCN)

- Recent works have applied GCN in semi-supervised problems to learn a node classifier from a partially labeled graph[8].
- Graphs provided more flexibility for representing image data.

[8]Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. ICLR 2017.

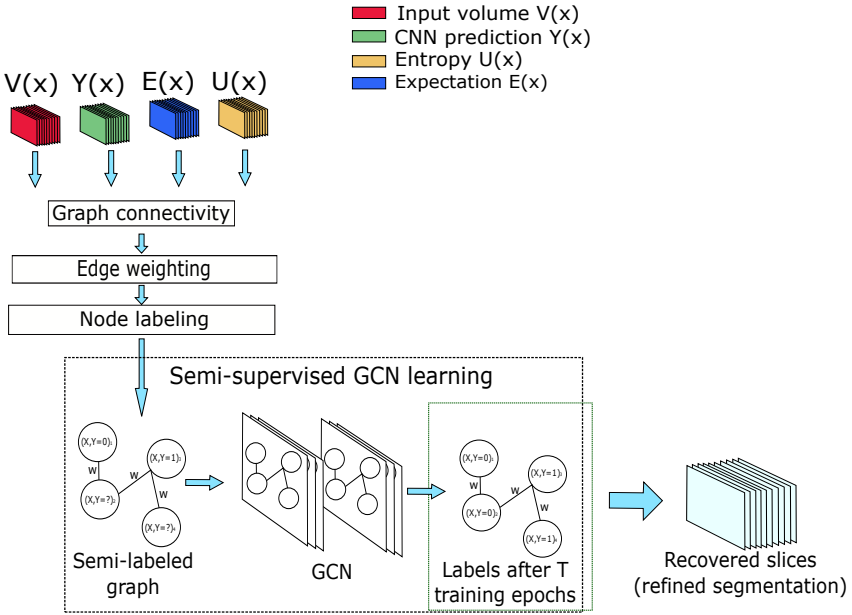
Overview and Contribution

We proposed a 2-step refinement process for the single organ segmentation problem in CT volumes:

- Uncertainty Analysis.
  - Finding high uncertainty and low uncertainty predictions.
  - High uncertainty is assumed to be potentially incorrect.

- GCN Refinement
  - Graph definition.
  - Semi-supervised gcn training, and graph evaluation (refined segmentation).

We show that our framework can increase the average dice score by 1% and 2% for pancreas and spleen segmentation models, respectively.



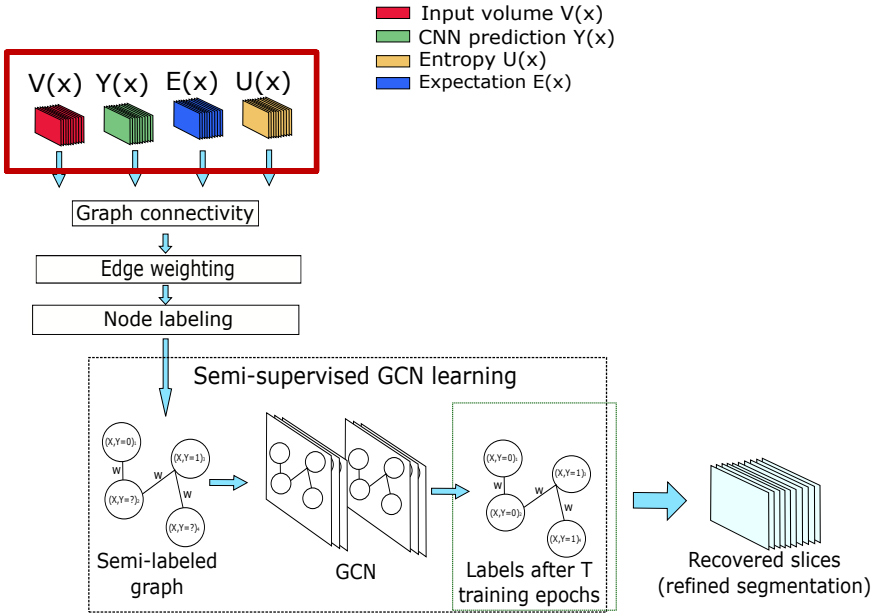
Overview and Contribution

We proposed a 2-step refinement process for the single organ segmentation problem in CT volumes:

- Uncertainty Analysis.
  - Finding high uncertainty and low uncertainty predictions.
  - High uncertainty is assumed to be potentially incorrect.

- GCN Refinement
  - Graph definition.
  - Semi-supervised gcn training, and graph evaluation (refined segmentation).

We show that our framework can increase the average dice score by 1% and 2% for pancreas and spleen segmentation models, respectively.



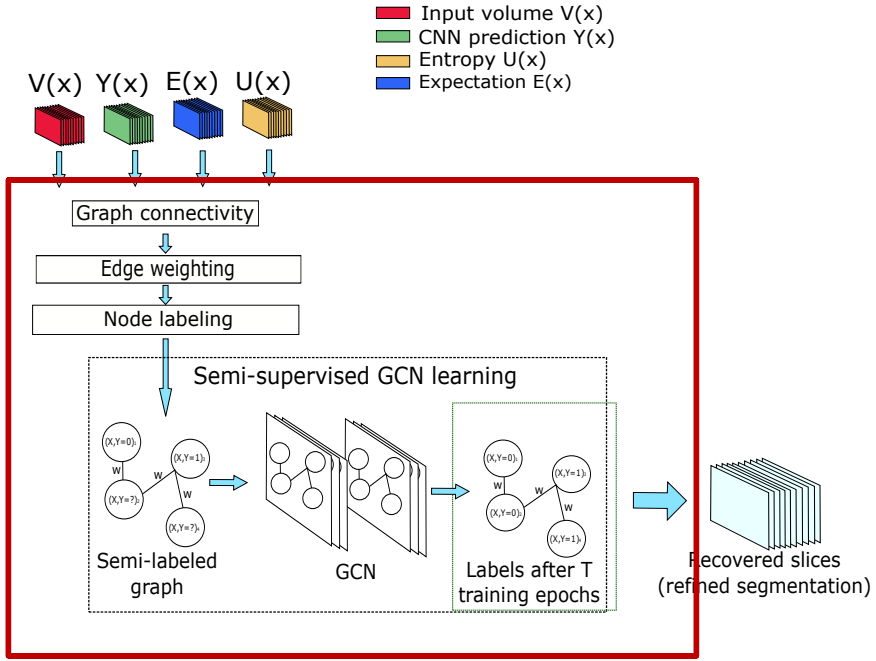
Overview and Contribution

We proposed a 2-step refinement process for the single organ segmentation problem in CT volumes:

- Uncertainty Analysis.
  - Finding high uncertainty and low uncertainty predictions.
  - High uncertainty is assumed to be potentially incorrect.

- GCN Refinement
  - Graph definition.
  - Semi-supervised gcn training, and graph evaluation (refined segmentation).

We show that our framework can increase the average dice score by 1% and 2% for pancreas and spleen segmentation models, respectively.

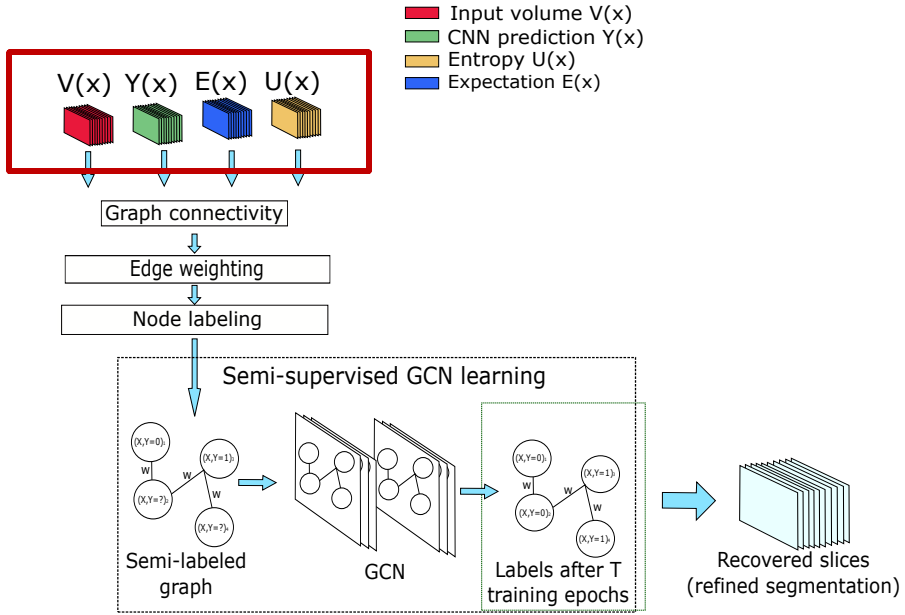


Expectation, Uncertainty and Wrong Elements Proposal

Consider a trained CNN model $\mathrm{Y} = g(V(x), \theta)$ with parameters θ, and an input $V(x)$ with x point elements (pixels or voxels). Following MCDO, we apply dropout at inference time, and perform T stochastics passes to get the model's expectation as:

$$\mathbb{E}(x) \approx \frac{1}{T} \sum_{t=1}^{T} g(V(x), \theta_t),$$

With θ_t the model's weights after applying dropout in the *t* stochastic pass.

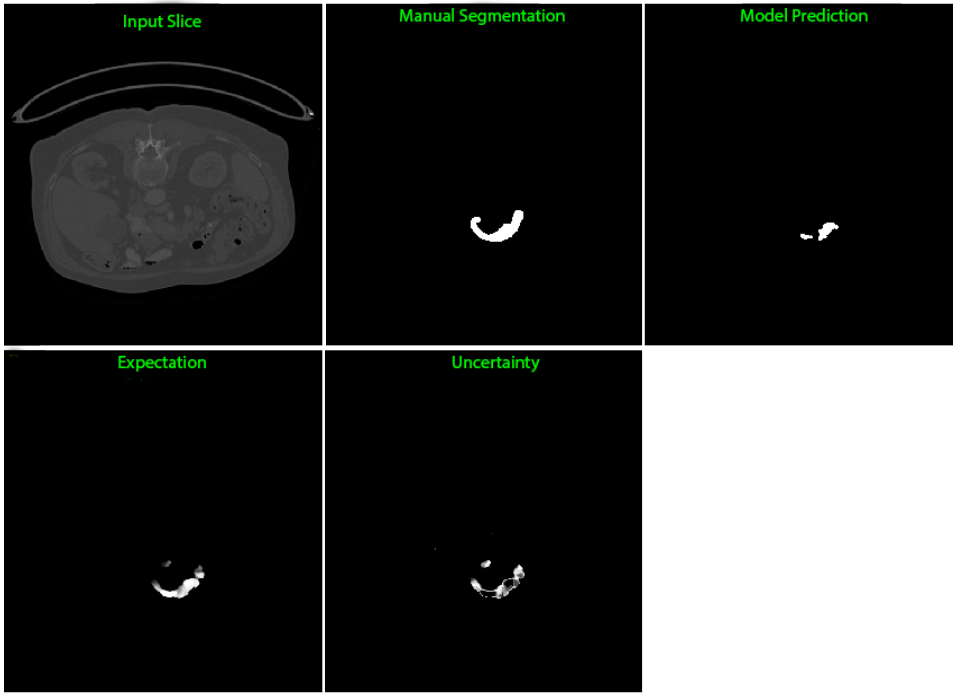


Expectation, Uncertainty and Wrong Elements Proposal

Uncertainty is obtained based on the model's entropy:

$$\mathbb{U}(x) = H(x) = -\sum_{c=1}^{M} P(x)^c \log P(x)^c,$$

With M the number of classes and $P(x)^c$ the probability for class c (given by $\mathbb{E}$)



Expectation, Uncertainty and Wrong Elements Proposal

In order to define potential misclassified candidates, $\mathbb{U}$ is binarized by a threshold τ.

$$U_b(x) = \mathbb{U}(x) > \tau,$$

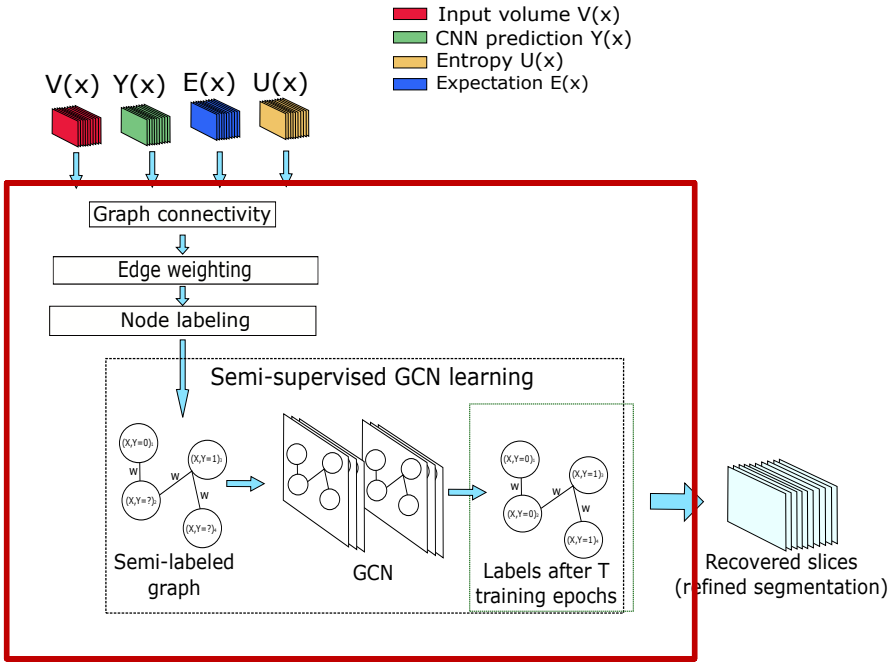
$U_b(x)$ indicates the high uncertainty voxels of the prediction of the CNN.

Refinement as a Semi-supervised GCN

The refined segmentation Y^* is obtained as the output of a GCN model Γ:

$$Y^* = \Gamma(\mathcal{G}(S); \phi),$$

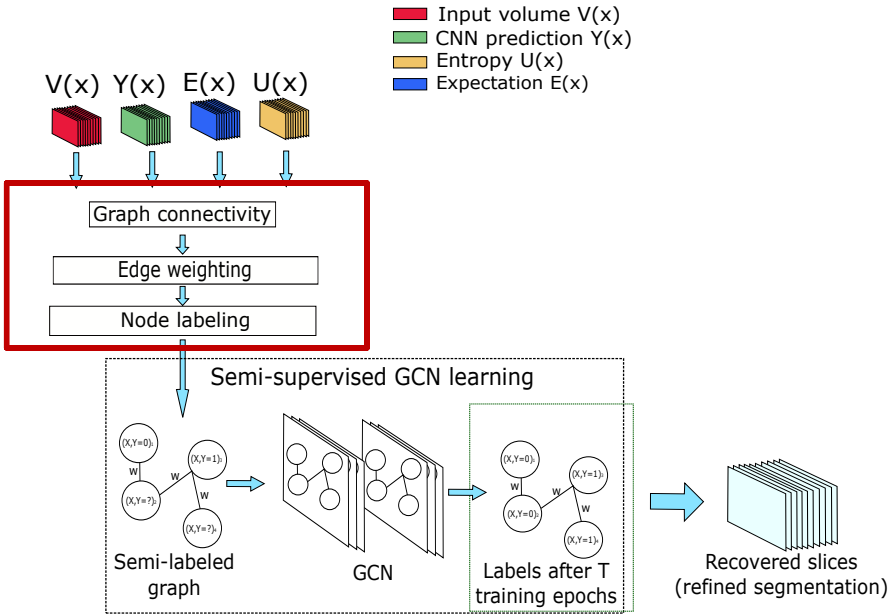
With $\mathcal{G}(S)$ a partially labeled graph constructed from a set of input volumes $S = \{\mathbb{E}, \mathbb{U}, V, Y\}$, and ϕ the trained GCN parameters.



Node Labeling

Each voxel is considered a node. A node is represented by its intensity, expectation, and entropy. The labels are set according to the following rule:

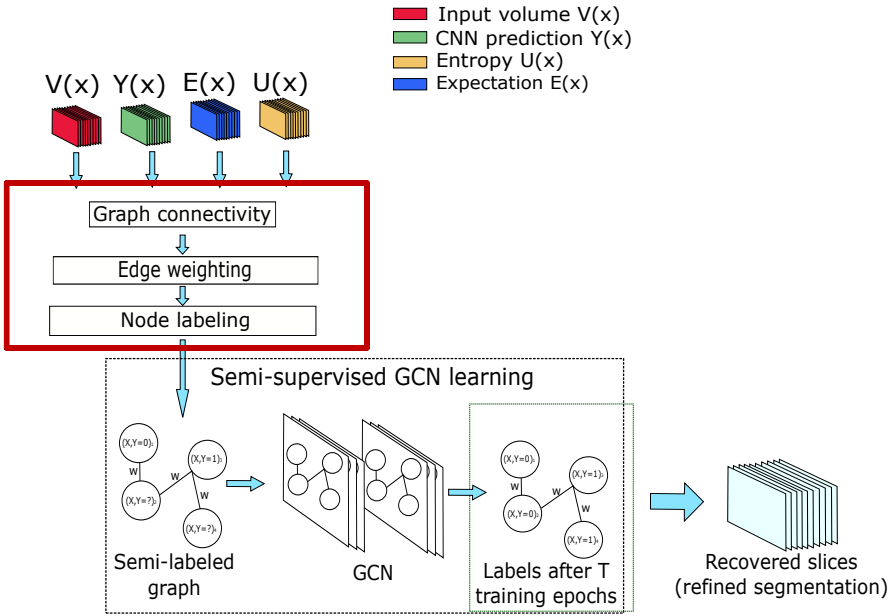
$$l(x) = \begin{cases} Y(x) & \text{if } U_b(x) = 0 \\ \text{unlabeled} & \text{if } U_b(x) = 1 \end{cases}$$



Node Labeling

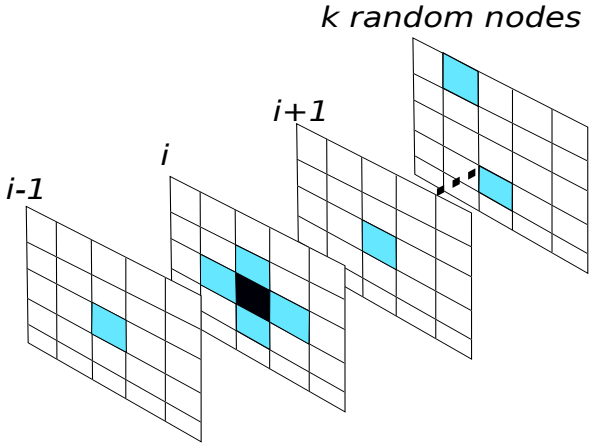
Each voxel is considered a node. A node is represented by its intensity, expectation, and entropy. The labels are set according to the following rule:

$$l(x) = \begin{cases} Y(x) & \text{if } U_b(x) = 0 \\ \text{unlabeled} & \text{if } U_b(x) = 1 \end{cases}$$



Connectivity and Weigthing

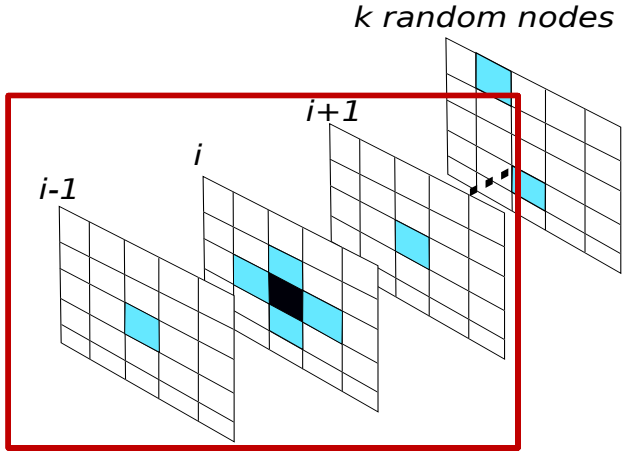
For a node, a connection is created with its six perpendicular neighbors. Additionally, connections with k=16 random nodes inside the ROI are added (long-range connections).



Graph connectivity

Connectivity and Weigthing

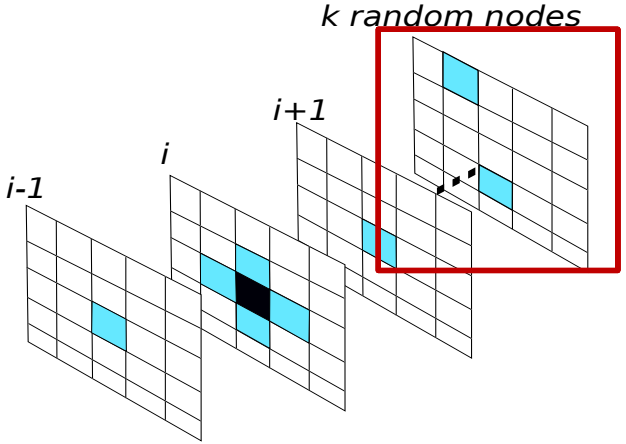
For a node, a connection is created with its six perpendicular neighbors. Additionally, connections with k=16 random nodes inside the ROI are added (long-range connections).



Graph connectivity

Connectivity and Weigthing

For a node, a connection is created with its six perpendicular neighbors. Additionally, connections with k=16 random nodes inside the ROI are added (long-range connections).



Graph connectivity

Connectivity and Weigthing

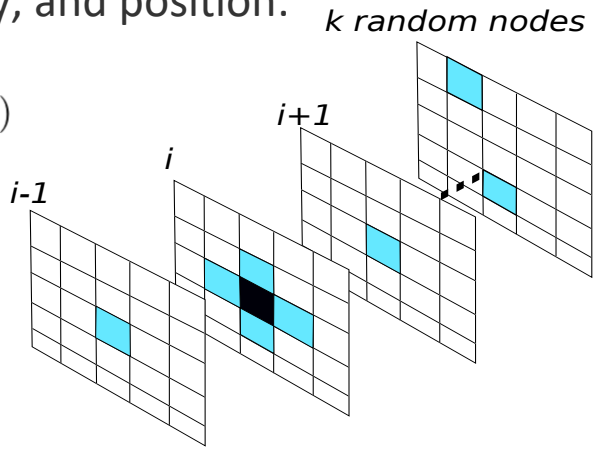
For a node, a connection is created with its six perpendicular neighbors. Additionally, connections with k=16 random nodes inside the ROI are added (long-range connections).

Edge weighting is given by **similarity** in **expectation**, intensity, and position:

$$w(\boldsymbol{x}_i, \boldsymbol{x}_j) = \lambda \mathrm{div}(\boldsymbol{x}_i, \boldsymbol{x}_j) + \exp\left(-\frac{||\boldsymbol{V}(\boldsymbol{x}_i) - \boldsymbol{V}(\boldsymbol{x}_j)||^2}{2\sigma_1}\right) + \exp\left(\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{2\sigma_2}\right)$$

With div the diversity[9]:

$$\mathrm{div}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{c=1}^{M} (p_i^c - p_j^c) \log \frac{p_i^c}{p_j^c} \text{ with } M = 2,\ p_i^1 = \mathbb{E}(\boldsymbol{x}_i)$$



Graph connectivity

[9] Zhou, Z., Shin, J, etal.: Fine-tuning Convolutional Neural Networks for Biomedical Image Analysis: Actively and Incrementally. CVPR 2017

Connectivity and Weigthing

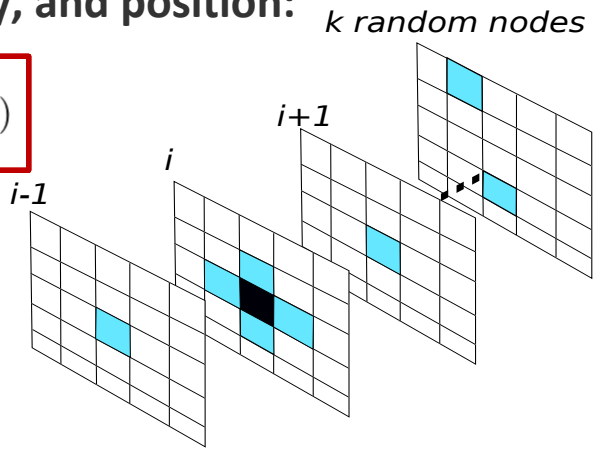
For a node, a connection is created with its six perpendicular neighbors. Additionally, connections with k=16 random nodes inside the ROI are added (long-range connections).

Edge weighting is given by **similarity** in expectation, **intensity, and position:**

$$w(\boldsymbol{x}_i, \boldsymbol{x}_j) = \lambda \mathrm{div}(\boldsymbol{x}_i, \boldsymbol{x}_j) + \exp\left(-\frac{||\boldsymbol{V}(\boldsymbol{x}_i) - \boldsymbol{V}(\boldsymbol{x}_j)||^2}{2\sigma_1}\right) + \exp\left(\frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||^2}{2\sigma_2}\right)$$

With div the diversity[9]:

$$\mathrm{div}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{c=1}^{M} (p_i^c - p_j^c) \log \frac{p_i^c}{p_j^c} \text{ with } M = 2,\ p_i^1 = \mathbb{E}(\boldsymbol{x}_i)$$



Graph connectivity

[9] Zhou, Z., Shin, J, etal.: Fine-tuning Convolutional Neural Networks for Biomedical Image Analysis: Actively and Incrementally. CVPR 2017

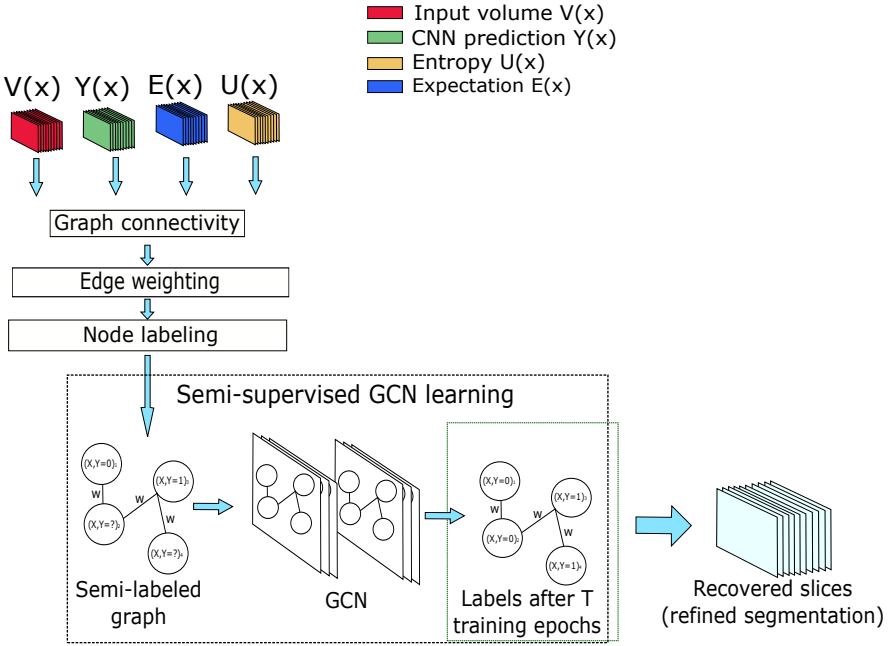
GCN Refinement

We use the semi-supervised GCN defined by Kipf to train the model:

$$Z = f(X, A) = \mathrm{softmax}\Big(\hat{A}\ \mathrm{ReLU}\Big(\hat{A}XW^{(0)}\Big)\,W^{(1)}\Big)$$

$$\hat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$$

$$\tilde{A} = A + I_N \text{ and } \tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$$



Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. ICLR 2017.

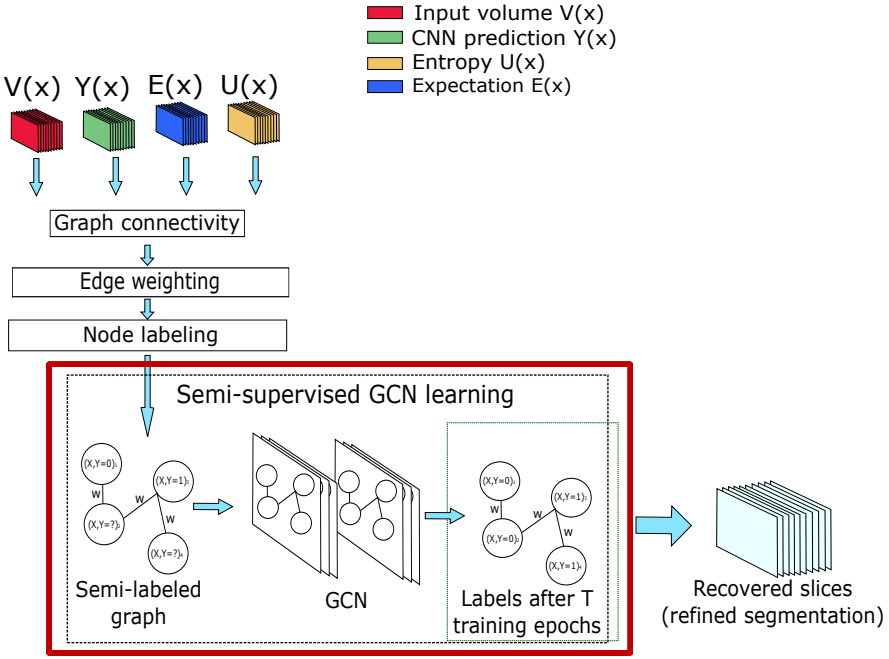
GCN Refinement

We use the semi-supervised GCN defined by Kipf to train the model:

$$Z = f(X, A) = \mathrm{softmax}\Big(\hat{A}\ \mathrm{ReLU}\Big(\hat{A}XW^{(0)}\Big)\,W^{(1)}\Big)$$

$$\hat{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$$

$$\tilde{A} = A + I_N \text{ and } \tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$$



Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. ICLR 2017.

Implementation Details

- The CNN is given by a 2D U-Net trained on axial slices.
- To refine the CNN's output, the uncertainty analysis is performed and a GCN is trained for each individual input volume according to:
  - The uncertainty analysis is performed using Monte Carlo dropout with a dropout rate of 0.3 and T=20
  - The GCN model is a two-layered network with 32 feature maps in the hidden layer and a single output.
  - The GCN trains for 200 epochs with a learning rate of 1e-2, using the binary cross-entropy loss with the Adam optimizer.
- We compare with traditional Conditional Random Field refinement.

Datasets

We validate our process for the pancreas and spleen segmentation problems

- Pancreas: NIH dataset*
  - For CNN Training: 45 CT volumes.
  - For Refinement Testing: 20 CT volumes.

- Spleen: Medical Image Segmentation Decathlon**
  - For CNN Training: 26 CT volumes.
  - For Refinement Testing: 9 CT Volumes

*https://wiki.cancerimagingarchive.net/display/Public/Pancreas-CT
**http://medicaldecathlon.com/

Refinement Performace

Dice Score before and after refinement. The CNN trains on 45 volumes for the pancreas and 26 volumes for the spleen.

Task	CNN 2D U-Net	CRF refinement	GCN Refinement (ours)
Pancreas	76.9 ± 6.6	77.2 ± 6.5	$\mathbf{77.8 \pm 6.3}$
Spleen	93.2 ± 2.5	93.4 ± 2.6	$\mathbf{95.1 \pm 1.3}$

Uncertainty Threshold

Dice score performance of the uncertainty-GCN refinement under different uncertainty thresholds τ.

Task	GCN $\tau = 1e-3$	GCN $\tau = 0.3$	GCN $\tau = 0.5$	GCN $\tau = 0.8$	GCN $\tau = 0.999$
Pancreas	77.71 ± 6.3	77.79 ± 6.4	77.77 ± 6.3	77.81 ± 6.3	77.79 ± 6.3
Spleen	95.01 ± 1.5	94.92 ± 1.4	94.98 ± 1.4	94.97 ± 1.4	95.07 ± 1.3

Influence of the Number of Training Samples

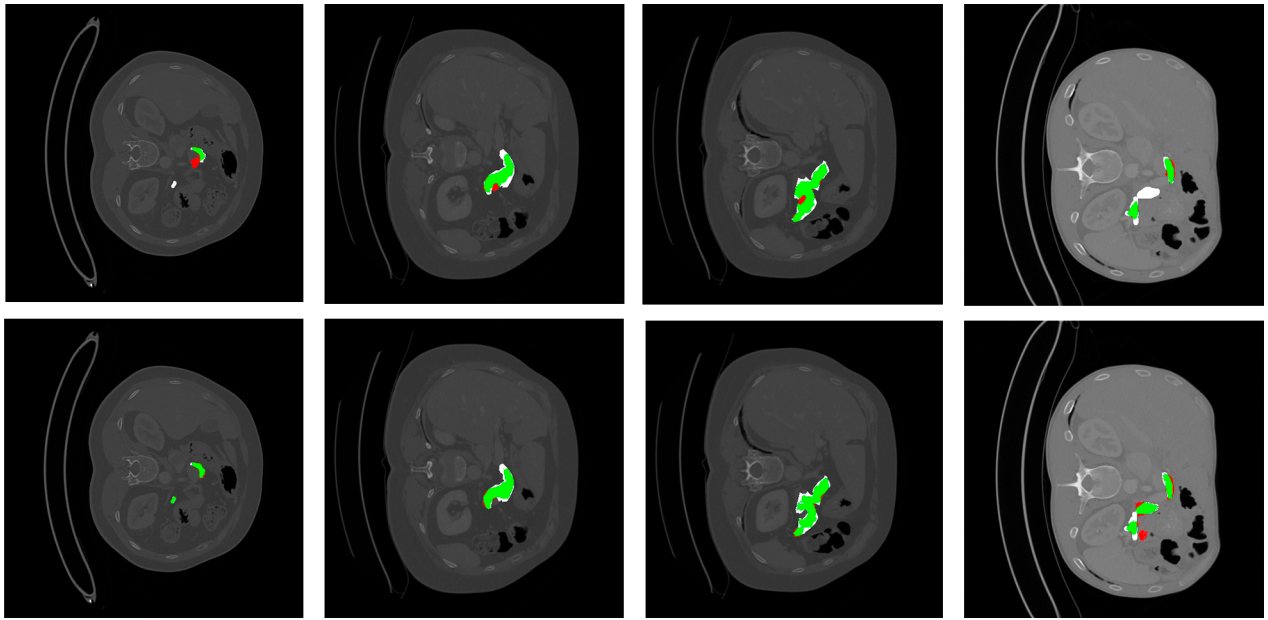
Dice Score before and after refinement when the CNN trains on a small number of samples. The CNN trains on **10 volumes** for the pancreas and **9 volumes** for the spleen.

Task	CNN 2D U-Net	CRF refinement	GCN Refinement (ours)
Pancreas-10	52.10 ± 22.61	52.20 ± 22.62	$\mathbf{54.50 \pm 22.15}$
Spleen-9	78.80 ± 28.40	78.80 ± 28.40	$\mathbf{81.15 \pm 28.90}$

Visual Results for Pancreas Segmentation

CNN Prediction

GCN Refinement



TP

FN

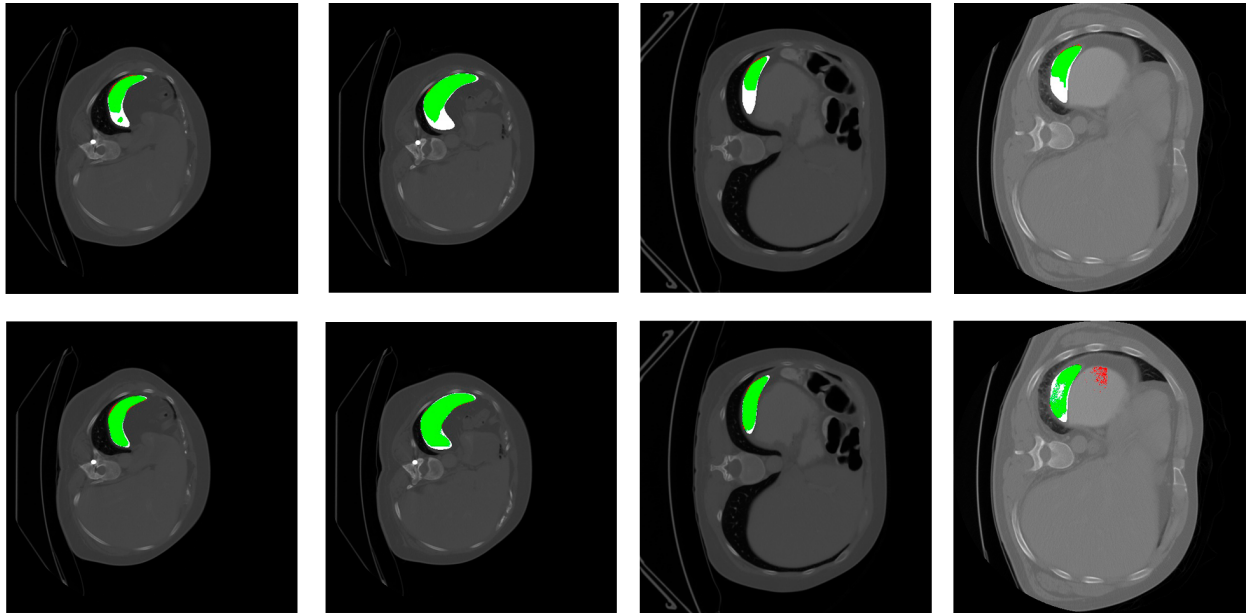
FP

https://wiki.cancerimagingarchive.net/display/Public/Pancreas-CT

Visual Results for Spleen Segmentation

CNN Prediction

GCN Refinement



TP

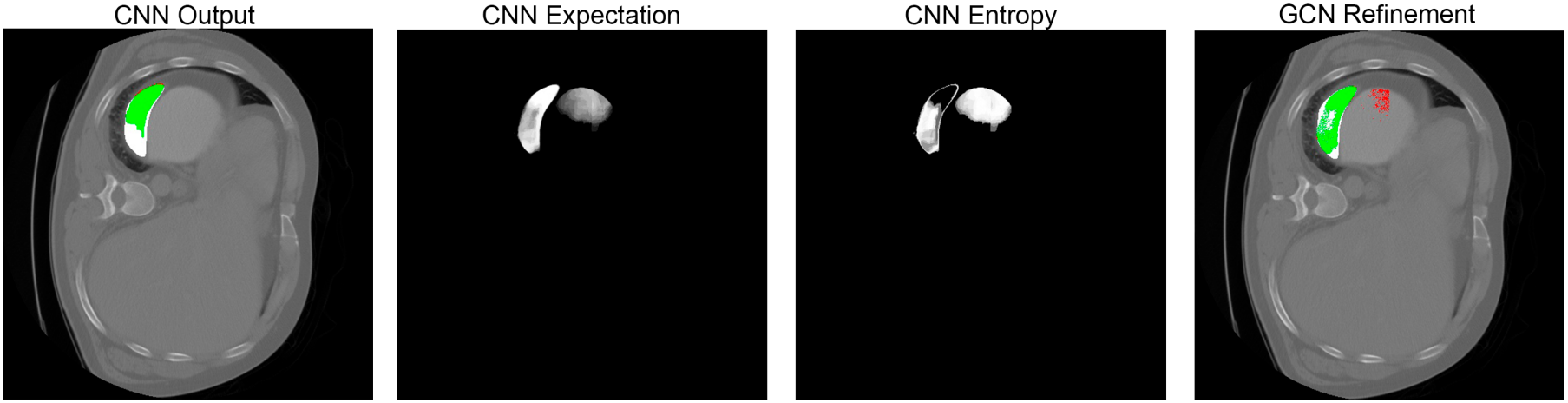
FN

FP

http://medicaldecathlon.com/

Prediction, Expectation, and Entropy

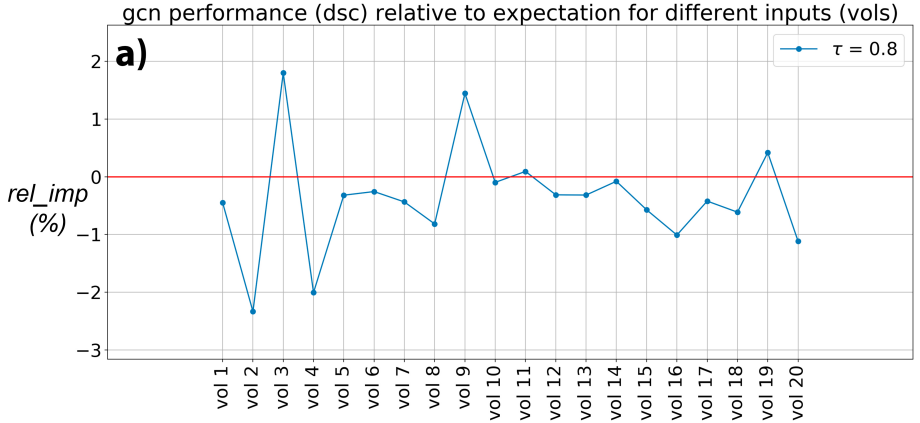
Intermediate outputs of the process

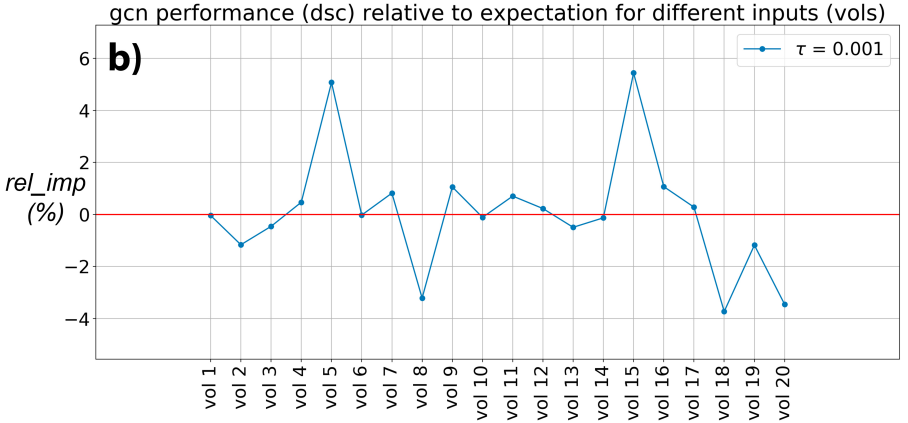


Prediction, Expectation, and Entropy

We threshold the expectation by 0.5 and compare the relative improvement of the GCN refinement and the expectation, with a CNN trained on 45 (a) and 10 (b) pancreas volumes.

$$rel_imp = \frac{gcn_{dsc} - expectation_{dsc}}{expectation_{dsc}} \times 100.$$





Concluding Remarks and Future Work

- We have presented a method to define an uncertainty-based partially-labeled graph representation of CT data.
- We have shown an application of GCNs in the segmentation refinement tasks.
- We have employed MCDO for uncertainty analysis and found that expectation could be a good choice for well-trained models, while the GCN refinement shows better performance in low-data regime.
- Our work is simple and modular, allowing future analysis of different uncertainty estimation methods.
- Future research can focus on different weighting and connectivity mechanisms, and in the inclusion of prior knowledge in the graph definition.

Thank you