Benjamin Albert
balbert2@jhu.edu

Review of Paper:
Multi-organ Segmentation over Partially Labeled Datasets with Multi-scale Feature Abstraction

Fang X, Yan P. Multi-organ segmentation over partially labeled datasets with multi-scale feature abstraction. IEEE Transactions on Medical Imaging. 2020 Jun 9;39(11):3619-29.

CIS II Project Overview:
We are developing software to automatically quantify the volume of blood in a hemothorax patient from CT imagery with known voxel dimensions. To do this, the software performs 3D segmentation of the hemothorax to generate a prediction mask; the voxels predicted as positive can then be summed to estimate the total hemothorax volume. In addition to quantifying the total volume, the 3D segmentation allows a human operator to assess the quality and legitimacy of the estimated hemothorax volume.

Reason for Choosing the selected Paper:
The reasons behind choosing this paper are five-fold:

1. The implementation is open-sourced and written in PyTorch so it can be evaluated on the hemothorax dataset
2. The developed algorithm is evaluated on abdominal CT scans
3. The network is particularly designed for better fusing and contextualizing multiscale features, which our mentor believes will improve our current results
4. The paper was first published very recently, less than a year ago
5. The paper is published in a good journal: IEEE Transactions on Medical Imaging.

Authors' Hypothesis and Objectives:
The authors develop a novel network architecture to compete with benchmark models on four distinct organ segmentation challenges. To do so, the authors' architecture focuses on new ways to fuse hierarchically learned features so as to maintain local and global contexts. The authors hypothesize "that the semantic information in various depths can be further enhanced by utilizing hierarchical contextual features. PIPO-FAN [Pyramid-Input Pyramid-Output Feature Abstract Network] aims to effectively extract multi-scale features for medical image segmentation, on top of the multi-scale nature of U-net."

Terminology:

- Multi-scale: pertaining to layer inputs, multiple scales arise from skip connections and explicit rescaling/pooling, both of which combine local with contextual information
- Pyramid structure: reducing an image size through convolution, pooling, etc. though typically adding many channels / learned filters
- Deeply supervised: evaluation and prioritization of discriminative latent features introduced by: Lee CY, Xie S, Gallagher P, Zhang Z, Tu Z. Deeply-supervised nets. In Artificial intelligence and statistics 2015 Feb 21 (pp. 562-570). PMLR.
- Semantic gap: bridge between low-level latent features and high-level/human features
- Attention mechanism: a component of a network to assign importance to particular features or regions of features, similar to low-level human visual attention (e.g. superior colliculus)

Implementation:

There are three components and properties of the authors' architecture:

- Equal convolutional depth (ECD): features that are fused have passed through the same number of convolutional filters. The premise of the equal convolutional depth is that "all the fused features at each step are at the same semantic abstraction level to better exploit the pyramid shape of U-Net."
- Adaptive Fusion (AF): attention mechanism to indicate importance at each scale
- Target adaptive loss (TAL): treats unknown labels as background and the final layer is branched to segment multiple organs

The network architecture begins with pyramid spatial pooling the CT scans (windowed to Hounsfield units of [-200, 200]), as seen on the left within Fig. 3. The associated ground truth is similarly pooled for later backpropagation. After passing the multiple scales through the PIPO module, which has the ECD the results are passed to the FAN module, as illustrated in Fig. 4.

Also seen in Fig. 4 is the juxtaposition of the information relayed by the multiple scales; the top-most instance carries fine-grained segmentation detail whereas the bottom-most instance is fuzzy but reflects class information better. The adaptive fusion module, which uses shared weights for all scales to achieve scale invariant inference, is then used to merge the pyramid output before the softmax prediction layer.

The target adaptive loss is simply a branched module off of the softmax layer that enables multiclass segmentations from partially labeled merged datasets. Given datasets A and B with corresponding binary classes C and D, the merged dataset would have instances from A that are unlabeled for class D, and instances from B with unlabeled class C instances. To still train on this partially labeled merged dataset, the authors simply branch the network off the softmax layer to

perform multiple binary segmentations, one per organ type with binary cross entropy loss. This is reflective of how a multiclass SVM and other multiclass ensembles of binary models work.

The model is trained on a single Titan X Pascal GPUs for up to 4000 epochs in 3 hours. Note, however, that the authors downsample the original inputs of size 512x512 to 256x256 for faster computation. Each epoch uses 3 contiguous axial slices that cropped to 224x224 at a random center. The authors use RMSprop to optimize cross entropy with learning rate decay. For the first 2000 epochs, the deep supervision module is applied to improve feature extraction, while the last 2000 epochs freeze the deep supervision module and activate the adaptive fusion module. The authors claim that this is helpful for selectively optimizing the adaptive fusion module.

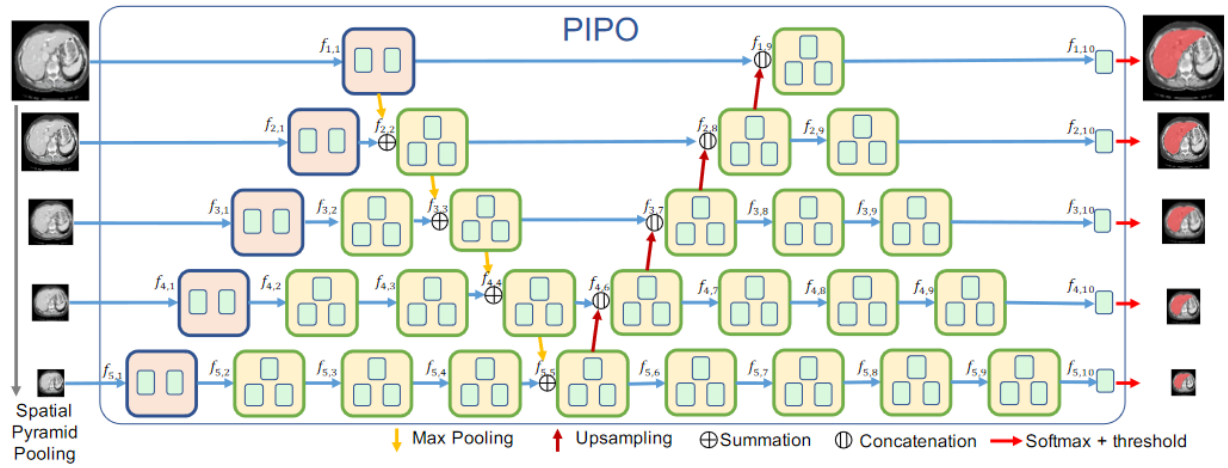

Fig. 3. Overview of the PIPO architecture. With the designed architecture, image information propagates from pyramid input to pyramid output through hierarchical abstraction and combination at each level.

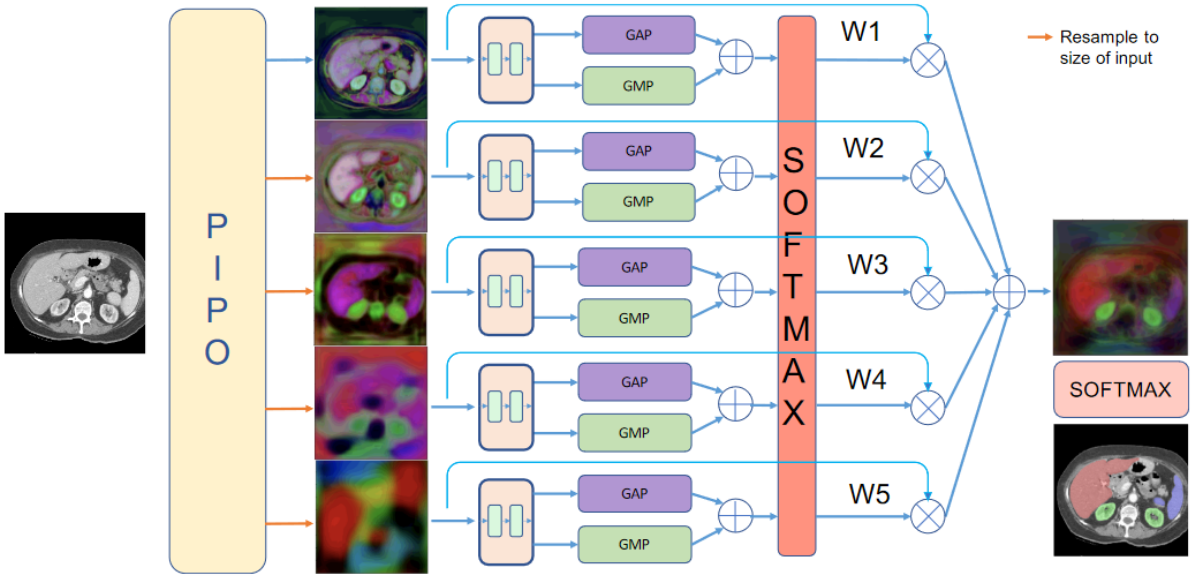

Fig. 4. Adaptive fusion of the multi-scale output segmentation features from PIPO-FAN. Features from lower scales tend to represent specific local segmentation, while features from higher scales are blurry but carry class information. Adaptive weights are computed by applying a shared convolutional module to the pyramid output features.

Results:

The Dice scores are tabulated below. In addition to the Dice scores, the authors provided example segmentations from PIPO-FAN and benchmark models for qualitative assessment. As can be seen from the example segmentations, PIPO-FAN appears to maintain crisp edges in segmentation structures that have complicated contours. By contrast, the benchmark models have fuzzier boundaries, particularly U-Net, and make more incorrect predictions. However, the assessment of incorrect predictions is better left to the Dice scores, which corroborate the qualitative results. In particular, observing the Table IV results on the combination of all datasets, PIPO-FAN outperforms DeepLabV3 on all organs and overall performs better than U-Net. The authors also note that the inference time is fast, reaching 0.04 seconds per slice on a single GPU.

TABLE III
PERFORMANCE COMPARISON WITH OTHER NETWORKS ON THE BTCV DATASET. (DICE %)

Architecture	Liver	Kidney	Spleen	Average
U-Net [7]	95.6	89.7	91.0	92.1
ResU-Net [11]	95.1	91.3	90.9	92.4
DeepLabV3 [53]	94.2	86.0	87.4	89.2
PIPO	95.7	92.6	90.1	92.8
PIPO-FAN	**95.8**	**92.7**	**92.3**	**93.6**

TABLE IV
PERFORMANCE COMPARISON WITH OTHER NETWORKS ON THE COMBINED ALL DATASETS. (DICE %)

Architecture	Liver	Kidney	Spleen	Average
U-Net [7]	**95.9**	**92.7**	93.5	94.0
DeepLabV3 [53]	94.1	89.6	90.9	91.5
PIPO-FAN	**95.9**	91.9	**95.5**	**94.4**

TABLE V
FIVE-FOLD CROSS VALIDATION AGAINST OTHER BENCHMARK METHODS ON TWO OPEN CHALLENGE DATASETS. (DICE %)

Architecture	LiTS	KiTS
U-Net [7]	93.9 ± 0.50	95.8 ± 0.91
ResU-Net [11]	94.1 ± 0.88	94.8 ± 1.06
DenseU-Net [2]	94.1 ± 0.30	94.2 ± 2.08
PIPO	95.3 ± 0.62	**96.5 ± 0.55**
PIPO-FAN	**95.6 ± 0.48**	96.2 ± 1.02

TABLE VI
ABLATION STUDY OF PIPO-FAN NETWORK STRUCTURES ON LITS DATASET (DICE %)

Architecture	Avg. Dice	Glb. Dice
Single-scale input/output	94.1	94.5
PIPO w/o ECD	95.1	95.2
PIPO-FAN w/o ECD	95.2	95.1
PIPO with ECD	95.3	95.4
PIPO-FAN with ECD	**95.6**	**95.8**

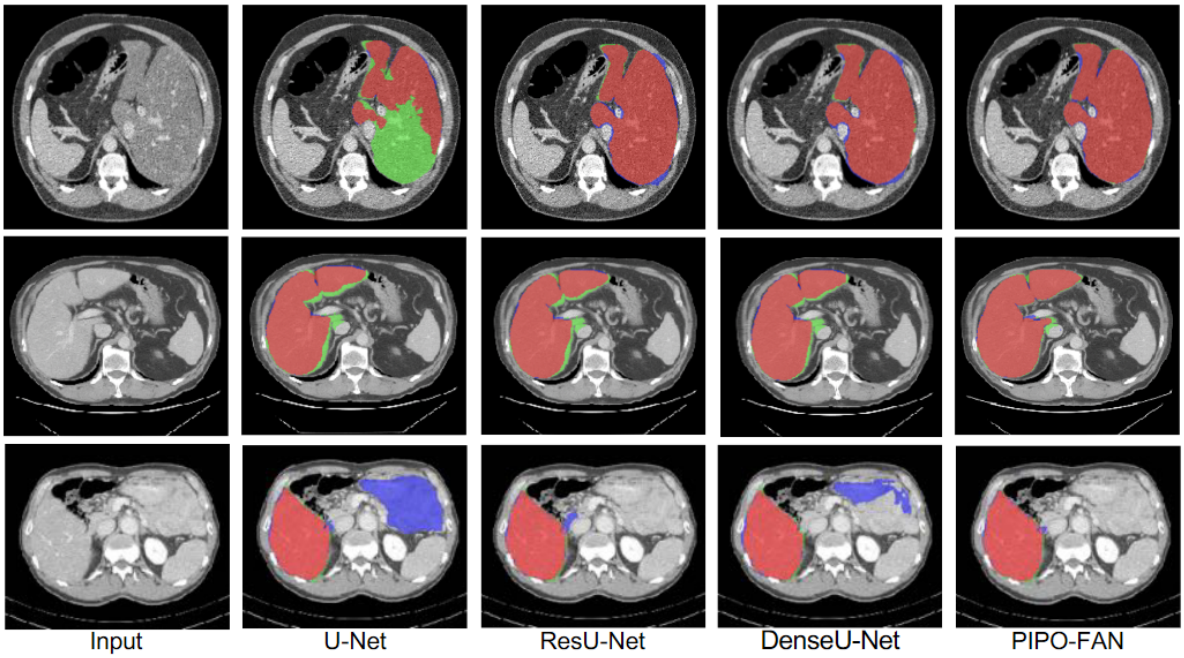

Fig. 6. Segmentation examples of different methods on LiTS data. From left to right are the raw image, results of U-Net, ResU-Net, DenseU-Net and our proposed PIPO-FAN, the red depicts correctly predicted liver segmentation, the blue shows false positive, green shows false negative.

Good Parts:
The main good parts of the paper are the dice scores, which generally surpass those of the benchmark networks. The authors also conduct an ablation study to assess the effects of the FAN and ECD, demonstrating that PIPO without ECD is comparable to PIPO-FAN without ECD. However, the combination of FAN and ECD are demonstrated to give slightly better results, improving by 0.3% average Dice.

Major Criticisms:
The authors list the contributions of their work as:

1. Novel Pyramid-Input Pyramid-Output Feature Abstraction Network (PIPO-FAN) to address the semantic gap that arises in multiscale features
2. Adaptive weighting layer to combine multiscale features
3. Adaptive loss to enable learning from partially labeled datasets
4. Good performance on public datasets

Although the first contribution is valid and good, the other three are not. Specifically, an adaptive weighting layer to combine multiscale features is simply an attention mechanism; although the authors state this, it is not novel and should not be considered a contribution. The adaptive loss to enable learning from partially labeled datasets at face-value is interesting, but the actual implementation of it is simply a branching schema similar to how multiclass SVMs operate; hence, it too is not novel. The fact that the results of PIPO-FAN outperform the benchmark models should be considered a prerequisite to publication rather than a contribution.

The authors do, however, open-source their PyTorch implementation of PIPO-FAN. Unfortunately, it is entirely undocumented and uncommented. The few comments that exist are simply commented out blocks of code with no explanations.

The authors also make many claims that they do not support. For example, the authors write that "DPS can help relieve the problem of gradient vanishing in deep neural networks and learn deep level features with hierarchical contexts. It also enforces the outputs in all scales to maintain structural information." Although deep supervision is demonstrated to relieve the gradient vanishing problem in deep neural networks in the original paper by Lee et al., the authors do not support the claim that their Deep Pyramid Supervision method "enforces the outputs in all scales to maintain structural information". Moreover, the authors need an in-text citation to Lee et al. to establish the fact that deep supervision alleviates the gradient vanishing problem (Lee CY, Xie S, Gallagher P, Zhang Z, Tu Z. Deeply-supervised nets. In Artificial intelligence and statistics 2015 Feb 21 (pp. 562-570). PMLR.)

In total, PIPO and PIPO-FAN were benchmarked against four state-of-the-art networks on four separate challenge datasets. However, the authors selectively chose a subset of these four networks against which to compare for each challenge dataset. For example, Table III

benchmarks against U-Net, ResU-Net, and DeepLabV3, but table V benchmarks against U-Net, ResU-Net, and DenseU-Net. For consistent evaluation criteria and benchmarking, PIPO and PIPO-FAN should have been evaluated against all networks for all datasets. Otherwise, it leaves room for manipulation of results by omission.

The authors assess statistical significance by using the t-test. They claim that PIPO-FAN significantly outperforms the benchmark models due to the p-values returned by the t-test. However, the authors do not demonstrate nor state that the underlying data distribution is normally distributed, which is one of the assumptions of the test. It would have been better for them to have used the Wilcoxon signed rank test because the data are paired and the Wilcoxon signed rank test does not assume normal distribution of the data.

Minor Criticisms:

Table VII, which enumerates the total number of parameters in PIPO-FAN, does not compare the number of parameters to those in the benchmark models. Some acronyms are used before defined (e.g. GAP/GMP, meaning global average pooling and global max pooling respectively, in Fig. 4). Lastly, the materials section does not describe nor list the GPUs, but the authors eventually mention them in the results and acknowledgement sections.