Gary Yang
yyang117@jhu.edu

Choice of Papers:
U-Net: Convolutional Networks for Biomedical Image Segmentation
Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. In International Conference on Medical image computing and computer-assisted intervention 2015 Oct 5 (pp. 234-241). Springer, Cham. https://arxiv.org/pdf/1505.04597.pdf

3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation
Çiçek Ö, Abdulkadir A, Lienkamp SS, Brox T, Ronneberger O. 3D U-Net: learning dense volumetric segmentation from sparse annotation. In International conference on medical image computing and computer-assisted intervention 2016 Oct 17 (pp. 424-432). Springer, Cham. https://arxiv.org/pdf/1606.06650.pdf

Project Overview:
I am part of Team 9 that works on "predicting hemorrhage-related outcomes with CT volumetry for traumatic hemothorax." The team consists of Benjamin Albert, Chang Yan, and I. We are mentored by Dr. Dreizin and Professor Unberath. We have about 80 sets of usable axial CT scans. Scans of hemothoratic patients are manually labeled. Scans along with segmentation are then rescaled to a uniform dimension where each voxel corresponds to a determined volume measured in cubic centimeters. Our task is to train deep neural networks that can automatically segment scans. The predictions are evaluated based on the overlap of predicted segmentations to ground truth segmentations, the predicted volume to ground truth volume, and examined in visualization software like 3D Slicer [1].

[1] Fedorov A., Beichel R., Kalpathy-Cramer J., Finet J., Fillion-Robin J-C., Pujol S., Bauer C., Jennings D., Fennessy F.M., Sonka M., Buatti J., Aylward S.R., Miller J.V., Pieper S., Kikinis R. 3D Slicer as an Image Computing Platform for the Quantitative Imaging Network. Magn Reson Imaging. 2012 Nov;30(9):1323-41. PMID: 22770690. PMCID: PMC3466397.

Reason for My Choices:
- We have a volumetric segmentation task, and the most influential model in this domain is 3D U-Net. It is not only the most famous model, but also the necessary baseline for comparison.
- Although 3D U-Net is published in 2016, it is the backbone to many of the state-of-the-art models. In other word, many new model architectures are realized by adding modules to 3D U-Net.
- 3D U-Net paper is closely related to U-Net, a 2D version of the network. Because U-Net is published in 2015, one year earlier than 3D U-Net, it contains more details and rationale about architectures.

Summary of Work:
Ronneberger et al. aim to develop a time-efficient deep network that utilizes few training instances for segmentation tasks. They developed a model consisting of down-sampling for encoding and up-sampling for decoding to tackle the segmentation task. They repetitively

applied convolution operation to gather pixels' contexts (nearby pixels' values, status, etc). Then they used max-pooling operation to down-sample the images into latent spaces. The decoding portion of the architecture is essentially symmetric to the encoding part, just in reversed order. Convolution operations still exist, yet max pooling operations are replaced by transposed convolution. To aid localization accuracy, high-resolution images are concatenated to latent representations. This is the overall flow of their model, U-Net. In order to train deep networks with few examples, they used extensive data augmentation.

Çiçek et al aims to develop a deep neural network model that performs volumetric tasks, which are essentially segmenting a stack of slices of 2D images. They want the model to be capable of handling partially labeled data sets because manual labeling is time impermissible. They hypothesized that a 3D version of the U-Net, where all 2D convolution, 2D max pooling, and 2D transposed convolutions are replaced with their 3D counterparts, can accomplish this goal.

Vocabularies:

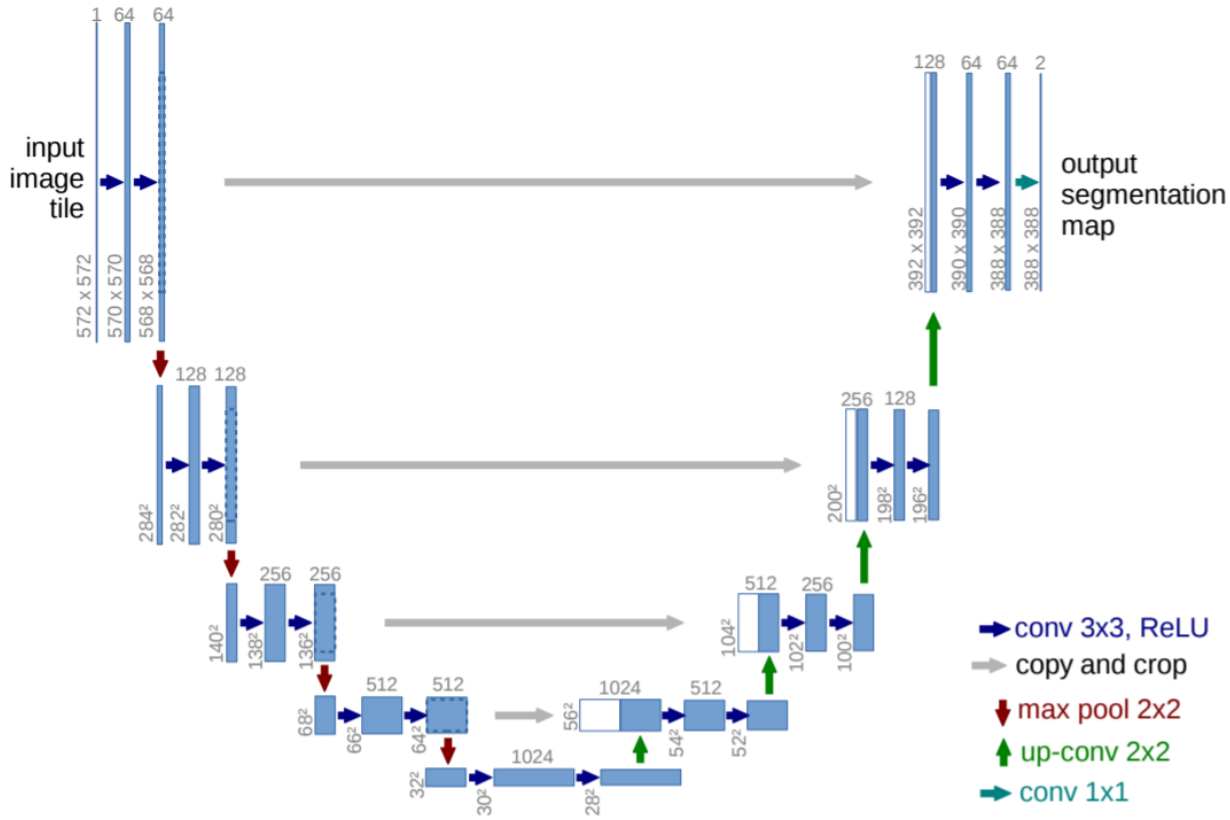
- Convolution: A matrix to matrix element-wise multiplication, followed by an overall summation.
- Max pool: An operation that selects the greatest value within the region of interest.
- Transposed convolution: A reversed convolution that up-samples. Multiple of the kernel filter matrix summed in a sliding window fashion.
- Data augmentation: Artificially increase the number of examples by performing rigid transformations, change in contrast, and elastic deformable transformations.
- Context: The values of surrounding elements. Segmentation is kind of like asking, what is the chance that this current element is belongs to the positive class, given the surrounding elements have the values they have. Sort like the idea of conditional probability.
- Encoder: The process of transforming matrices from the ijk coordinates of the image frame to an unknown frame
- Decoder: The process of transforming matrices from an unknown frame back to the original image frame.
- Batch Normalization: For each batch of training examples, pixel-wise normalization.

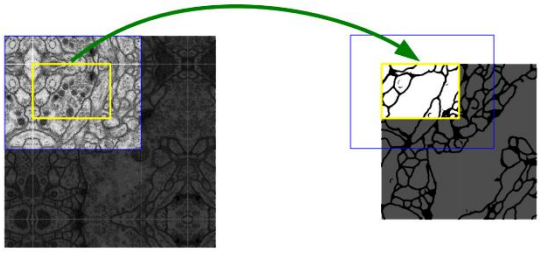
Implementations:

- Each blue arrow is a convolution operation following by ReLU activations.
- Each red arrow is a max-pooling operation, halving x, y dimension of images.
- Each grey arrow is a concatenation operation, where the matrix on the left joins the one on the right.
- Each green arrow is a transposed operation, doubling the x, y dimensions of images.
- The cyan-colored arrow gives the probability of pixel-wise prediction.

The model architecture can be decomposed into encoder and decoder. The left half of the down-sampling is the encoder portion, where images are being transformed into latent representations. Convolution summaries the nearby context. It gives us more condensed information, considering a broad range of elements. Each max pool can effectively halve the range of consideration; however, much spatial information is lost in this process. In this case, because each max pool selects one element from a 2*2 range. After 4 rounds of max pool, each value is effectively in

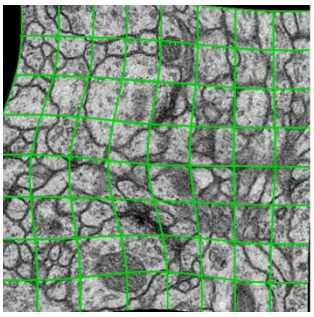
charge of representing a 16*16 matrix (thus, 1 in 256). This representation, therefore, sacrifices the localization accuracy, although rich context information is gained. To combat this problem, the authors choose to concatenate the original high-resolution images to the up-sampling path. Comparatively, fewer number of max-pooling is performed to the matrices on the left of each gray arrow, so they should contain more accurate spatial cues. Combined with the matrices on the right, which is richer in information, authors believe that good localization and the use of context can be ensured simultaneously.

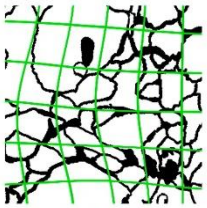




Segmentation: To ensure that corner pixels also have access to a full range of context. The authors performed padding in the manner of mirroring.

Data Augmentation: The authors performed rigid transformation and mild elastic deformation on each training instance. This action promotes the network to learn truly discriminative features rather than things like absolute positioning.





Results:

For U-Net, Ronneberger et al. claim to have the smallest warping error and reasonable rand error on ISBI electro-microscopic neuronal structure segmentation challenge.

Table 1. Ranking on the EM segmentation challenge [14] (march 6th, 2015), sorted by warping error.

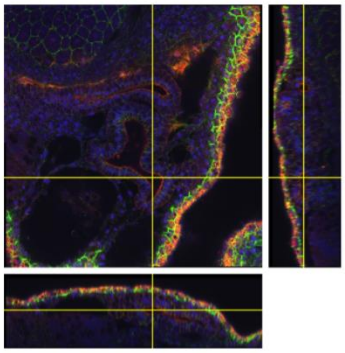
Rank	Group name	Warping Error	Rand Error	Pixel Error
	** human values **	0.000005	0.0021	0.0010
1.	u-net	**0.000353**	0.0382	0.0611
2.	DIVE-SCI	0.000355	0.0305	0.0584
3.	IDSIA [1]	0.000420	0.0504	0.0613
4.	DIVE	0.000430	0.0545	**0.0582**
⋮				
10.	IDSIA-SCI	0.000653	**0.0189**	0.1027

In addition, U-Net earned the highest Jaccard coefficient on ISBI cell tracking challenge. Jaccard coefficient is computed as true positives/(true positives + false negatives + false positives).

Table 2. Segmentation results (IOU) on the ISBI cell tracking challenge 2015.

Name	PhC-U373	DIC-HeLa
IMCB-SG (2014)	0.2669	0.2935
KTH-SE (2014)	0.7953	0.4607
HOUS-US (2014)	0.5323	-
second-best 2015	0.83	0.46
u-net (2015)	**0.9203**	**0.7756**

3D U-Net did compete in any competition and therefore is evaluated with 3-fold cross-validation. Because Çiçek et al. wanted to study the capacity of 3D U-Net learning partially labeled images, of the three instances of *Xenopus* kidney images, only 77 slices are labeled in total. These slices may be in any of the three views.



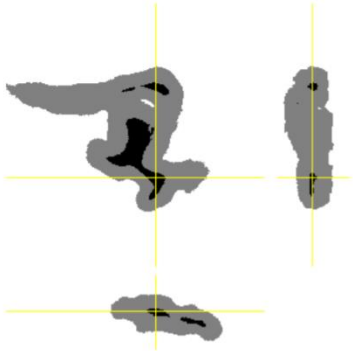


Table 1: Cross validation results for semi-automated segmentation (IoU)

test slices	3D w/o BN	3D with BN	2D with BN
subset 1	0.822	0.855	0.785
subset 2	0.857	0.871	0.820
subset 3	0.846	0.863	0.782
average	0.842	0.863	0.796

On average, the 3D network outperforms the 2D network, suggesting that spatial relationship is helpful. In addition, the batch normalization helps a little bit.

Criticism:

1. In the EM challenge, Ronneberger et al. only applaud themselves for achieving the highest wrapping error and a good rand error, without acknowledging what is each of the error measuring and the significance of three errors http://brainiac2.mit.edu/isbi_challenge/evaluation. This is no explicit mentioning of the reason why they choose to sort the errors based on wrapping error, rather than the other two metrics, which will place U-Net at a lower rank. Because it is a challenge, they did not perform k-fold cross-validation.

2. 3D U-Net authors state that only very few examples are needed for training segmentation models for many biomedical applications due to the intrinsic repetitive structures across instances. First of all, the statement does not hold. Many, if not all, of the diseased samples, are prone to belong to the distinct underlying distribution. Take our project as an example; the hemothoratic locations across subjects vary significantly.

3. For the 3D U-Net paper, although the author acknowledges that there are not many applications of 3D convolutional networks applied to volumetric tasks, I think they should still compare 3D U-Net to those models (just for the sake of completeness).

4. The U-Net code was published along with the paper, back in 2015. It was written in Caffe. Although it is not the authors' fault that Caffe is not as popular now as the other Deep Learning framework such as Tensorflow and PyTorch, for the sake of reproducibility and fair comparison, they should probably consider re-implement U-Net and 3D U-Net in the newer language. Several GitHub repos have U-Net implementations in PyTorch, yet they all differ from each other slightly. Any paper that uses U-Net as a baseline will inevitably be compared against different models. Even if the difference is minor, I think this problem should be addressed.

Relevance:

Lastly, I choose to review U-Net and 3D U-Net papers because they are relevant and fundamental. We used 3D U-Net as the baseline for our hemothorax volumetry task. It achieved 0.366 ± 0.096 for the right lung hemothorax and 0.463 ± 0.104 for the left lung hemothorax. A survey of related literature suggests that both U-Net and 3D U-Net fails to provide highly accurate localization during segmentation, even though high-resolution matrices are concatenated to latter layers with skip connections. Besides, U-Nets cannot adequately address potential class imbalance problems. In the case where the target of interest is a very tiny structure, the overwhelming number of negative instances might result in suboptimal learning. A weighted loss function can combat this problem, but only to some extent [2]. These observations lead to an attention-based, multi-scale network for our project, which we are still implementing.

[2] Dreizin, D., Zhou, Y., Zhang, Y., Tirada, N., & Yuille, A. L. (2020). Performance of a Deep Learning Algorithm for Automated Segmentation and Quantification of Traumatic Pelvic Hematomas on CT. Journal of digital imaging, 33(1), 243–251. https://doi.org/10.1007/s10278-019-00207-1