Transfer Learning for Ultrasound Image Despeckling

Leandra L. Brickson

***Abstract*—Medical ultrasound imaging forms an image through the measurement of backscattered sound waves in tissue. These images are impaired by speckle noise, a type of noise introduced by sub-resolution scatterers common in tissue. Currently, non-machine learning methods are used for Image denoising in clinical scanners. This project proposes to train an Ultrasound despeckling network which takes the ultrasound raw signal and outputs a despeckled 2D ultrasound image. To achieve this, a simulated training set was made in Field II and used to train a 3D U-net inspired architecture. Results of the 3D network for ultrasound image despeckling shows exciting SNR and CNR improvements on test data, though the network performs less well on edge cases in the test set.**

I. Introduction

Medical ultrasound imaging forms an image through the measurement of backscattered sound waves in tissue. Images formed using this technique are impaired with speckle noise, a type of noise introduced by sub-resolution scatterers common in tissue. An example of this noise can be shown in Figure 1. Currently, non-machine learning methods are used for Image denoising in clinical scanners, and only a few papers have been published on machine learning for speckle reduction with results never surpassing current denoising techniques. This report details the results of a transfer learning problem for the sake of ultrasound image reconstruction to achieve a lower speckle SNR and create cleaner, speckle-free images. This is ongoing work from the Dahl Ultrasound group in the department of Radiology. The dataset we will learn from is a simulated dataset, which takes a clean image, and simulates ultrasound imaging upon it to give a speckled output image. This speckled image can be used as the input to our network as an ultrasound image, and the original image can be used as the clean label. Using this dataset, we will train a U-net [2] inspired architecture to learn to despeckle raw simulated data. The intended use for this trained network is to transfer it to real data, to despeckle actual phantom (fake tissue) and in-vivo ultrasound images. Code for this project can be accessed through github at the following repo: https://github.com/llbricks/CS230Project.git

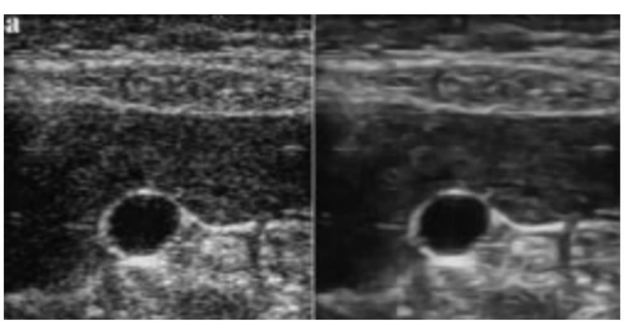

Fig. 1: Left)Example ultrasound image displaying speckle, a noise artifact present in ultrasound images. Right) Current denoising technique to remove speckle [1].

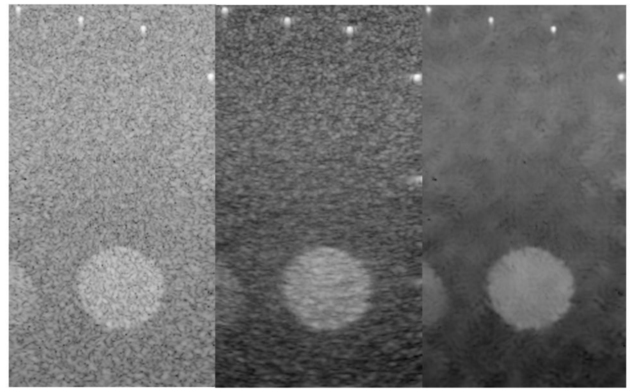

Fig. 2: Demonstration of despeckling methods described in Previous work. Left) Original Ultrasound image, Center) Despeckling using spatial compounding, Right) Despeckling using non-local means.

II. Related Work

Though there exist many methods currently used to despeckle ultrasound images, two methods will be detailed in this report for comparison. The first method, Spatial Compounding (SC) [3], is applied while converting the raw data from the transducer to form the image. This method breaks up the data channels from the transducer and forms many separate images. These images are then averaged to form a speckle reduced image. The second method, Non-Local Means [4], [5], is applied after the image is formed, and won't be detailed in this report. These methods will be used to compare to our despeckling technique results. An example of these images are shown in comparison to their original image in Figure 2. The SNR and CNR of these images are summarized in the results section.

Finally, before this course project, our research group has done previous work to learn ultrasound de-speckling using the fully formed output image. These initial networks mimicked U-net [2] , taking an input 2D image, down sampling to a denser representation space, then up-sampling to the original image size. The loss was the pixel-wise L2 difference between the label and the network output. Initial results seemed promising, showing an increased SNR, however the resulting images are still visually noisy, and additional improvements can be made. This method, along with it's associated original ultrasound image are shown in Figure 4, a) and b). It's metrics are summarized in the results section.

It is known from ultrasound literature that when we take the raw image data from the ultrasound transducer before pre-processing, we are able to gain more information on the speckle [6]. This data includes the individual signals from each transducer element, 128 channels of complex RF data for every image. Since the signals received across the transducer are slightly decorrelated, the data from the individual channels can provide more information on the speckle. This raw data is easily available from the simulated dataset as training data. This project proposes to take the previous architecture one step further to a fully convolutional 3D architecture, which will input a 3D volume of complex ultrasound raw data and output a 2D, speckle-free image.

III. Methods

To simulate the dataset, an ultrasound simulation package called Field II was used. The Imagenet and Places dataset was used to provide content to simulate imaging upon by mapping the image brightness to tissue properties. These datasets provided the network with a wide range of contrasts and shapes to train on. In addition to the simulation, a random amount of thermal and reverberation noise was added to the images. These are two common types of noise encountered in ultrasound imaging.

Since we are working with a transfer learning problem, and we are using a simulated dataset, it is important that the training set covers as wide a range of imaging conditions as possible. Therefore this training set should be simulated in many different conditions, imaging shapes, tissue properties and imaging settings, such as the transmission frequency. In doing this, we can generate a dataset which spans a wide range of possible inputs in hopes that our test set lies somewhere in that range.

The 3D network which has been made for this project again mimicks the U-net architecture. At all stages in the network, the data is a 3D volume and this network remains fully convolutional in all 3 dimensions, so we are learning on 3D convolution kernels. For this project, the complex data is treated as two separate values of real and imaginary, and a complex data type isn't used. The loss functions to be evaluated are L2 and L1 difference loss, structural similarity and total variation prior.

Hyperparameter tuning was performed on the network by manually tuning network depth, number of hidden layers, type of loss function and filter size. After this, random hyperparameter tuning of learning rate, dropout, batch size, L1 and L2 regularization was done. Additionally, when ultrasound images are viewed by humans, it is first log compressed, since it has too wide of a dynamic range. Because of this, both the log-compressed and uncompressed versions were used to calculate the loss, to see which domain the network performs best in. These networks were evaluated by rating the performance on a few images from the test set. Each resulting image was given a rating from 1 to 10 on the perceptual improvement in

Fig. 3: Example training data, the left column shows the original image before simulation, this will be used as a label. The right column shows the simulated speckled image, this will be used as the input

image quality. This manual evaluation is to ensure that the loss function correctly represents the perceptual transformation we would like. The network with the highest rated images were chosen.

To evaluate this network, a test set of multiple phantom and in-vivo images have been gathered. These images have no reference labels, so evaluation must be done on metrics which don't require a label. Due to the varied preferences of ultrasound sonographers and radiologists, utrasound despeckling methods are a bit difficult to evaluate with a single metric. For this report, we will choose to evaluate test images on both the image signal to noise ratio (SNR) and the contrast to noise ratio (CNR), which ensures that contrast differences are preserved. These metrics are defined to be:

$$\text{Contrast} = 20\log_{10}\left(\frac{\mu_t}{\mu_b}\right) \tag{1}$$

$$\text{CNR} = \frac{\mu_t - \mu_b}{\sqrt{\sigma_t^2 + \sigma_b^2}} \tag{2}$$

$$\text{SNR} = \frac{\mu_b}{\sigma_b}, \tag{3}$$

IV. Results

Sample training images before and after simulation are shown in Figure 3. These images were patched, reshaped and cropped as data augmentation to make 20k training samples of size (32,32,128,2) where 32x32 are the height and width of the image, 128 is the number of raw data channels, and 2 is the real and imaginary component of the data. As work on this project continues, this dataset will continue to increase as more simulations are made, to increase the breadth of the training dataset.

Using this dataset, different loss functions were tested. It was found that the L2 difference loss without log compression provided the best output images after about 20 epochs, so it was chosen as the best loss function for this task. The total variation prior was experimented with, but it degraded image quality, even with small regularization values, so it was omitted.

Hyperparameter tuning evaluation results are shown in Figure 6 (located on last page of this report). From this evaluation, the best rated network was chosen. The result of a test sample inferred on this network is shown in comparison to it's original image in Figure 4. This test sample is the same as those shown in Figure 2 for other despeckling techniques. The SNR and CNR values for the original image, the spatial compounded (SC) image, the non-local means (NLM) despeckled, 2D network(2DN) and 3D network(3DN) results are compared in the Table below. Additional test samples inferred on the best chosen network are shown with their original images in Figure 5.

	Original	SC	NLM	2DN	3DN
SNR	1.91	3.14	15.4	4.0	9.34
CNR	0.57	0.62	0.752	2.3	3.3

V. Discussion

The resulting images for the 3D network show significant improvement on the image quality and smoothness of bulk regions, while still preserving sharp edges of features. The network seems to fail the most in dark regions of the image, and around the point targets in the right image in Figure 5. We suspect that the poor performance may be due to the calculation of loss before log compression. In this regime, a difference in bright pixels would be orders of magnitude larger than an identical perceived difference in dark pixels. It is surprising that the log compressed loss calculations lead to very poor network output images, its possible there were errors in implementation that prevented the log compression loss to work properly. The point targets probably didn't turn out well because the training set didn't contain many small bright regions. This feature should be included in the next simulated batch of training data.

The results of the hyperparameter tuning are very interesting. the strong relationship between high ratings and low loss is very promising, and indicates that our loss is properly representing the transformation we want. It is also interesting to see no real importance of the weightings on the L1 and L2 weight regularizations. As a sanity check, this range was increased, and the rating started dropping off when the regularization was too high. The ratings for the dropout rate and batch size were displayed using violin plots, which widen when many points overlap in the same place. This is a good way to visualize data where both axes are discretized. From these violin plots, it appears that the network also doesn't seem to very dependent on batch size or dropout rate. A very important parameter, however, is the learning rate. there is a strong relationship between network rating and learning rate. The network performs poorly when the learning rate is both too high and too low.

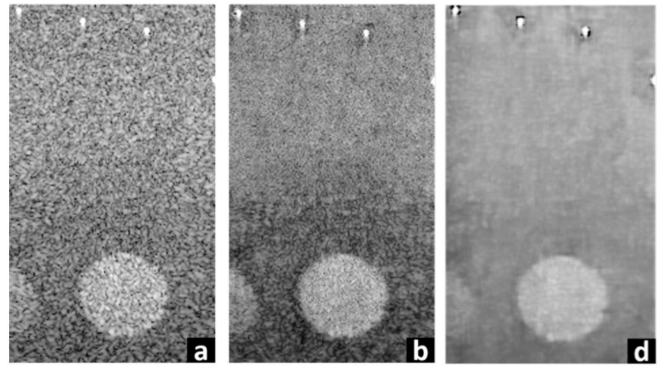

Fig. 4: Demonstration of network results on real phantom ultrasound data. a) original image, b) Initial results on 2D U-net architecture. d) Results on 3D U-net architecture.

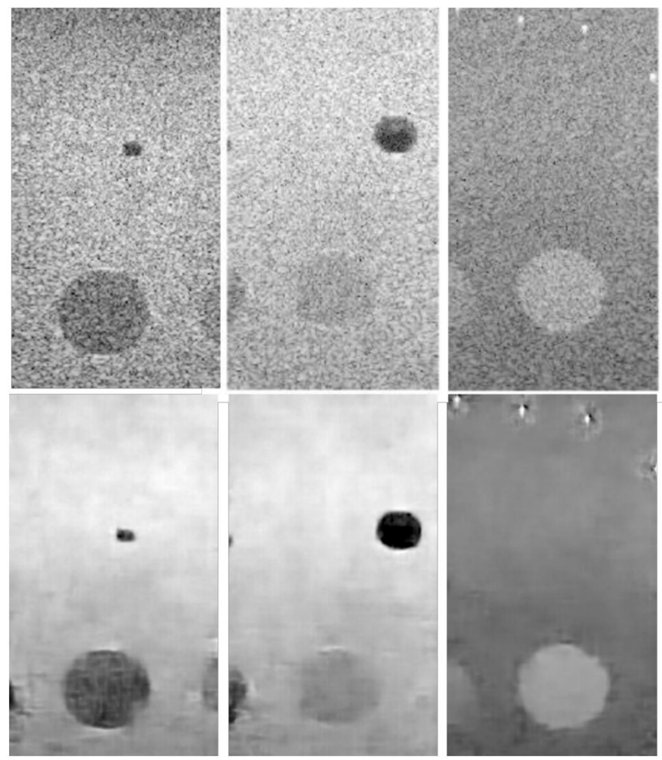

Fig. 5: More results on the 3D U-net architecture. Top row shows original ultrasound images, constructed from raw data, bottom row shows the images formed from raw data through the 3D network.

For the SNR and CNR metrics, we can see a significant SNR improvement in both the 3D network results and non local means, while the 2D and 3D networks provide significant CNR improvement. From this evaluation, it seems that the 3D network performs comparably to the conventional methods currently used in ultrasound despeckling.

VI. Conclusion

A simulated training set was made in Field II and used to train a 3D U-net inspired architecture. This network was tuned manually to narrow down good network structures and loss functions, then hyper parameter tuning was implemented with manual evaluation of the network. Network performance showed high dependence on learning rate. There was agreement of high evaluation with a low loss, indicating that the loss function successfully represented the desired transformation. Results of the 3D network for ultrasound image despeckling shows exciting SNR and CNR improvements on test data, though the network performs less well on edge cases in the test set, such as dark regions and point targets. An improvement for this could be continuing to expand the range of the simulated training set.

VII. Contributions

The author did the majority of the work on this project. It should be mentioned that this project is built off of the 2D input despeckling project, which was a project started by Dr. Dongwoon Hyun in the Jeremy Dahl lab. For this project, the initial dataset was simulated by Dr. Dongwoon Hyun, and the machine learning was done by both Dr. Hyun and Leandra. The 3D input network, however, was designed and implemented entirely by Leandra.

References

[1] K. Z. Abd-Elmoniem, A. B. M. Youssef, and Y. M. Kadah, "Real-time speckle reduction and coherence enhancement in ultrasound imaging via nonlinear anisotropic diffusion," *IEEE Transactions on Biomedical Engineering*, vol. 49, no. 9, pp. 997–1014, Sept 2002.

[2] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *CoRR*, vol. abs/1505.04597, 2015. [Online]. Available: http://arxiv.org/abs/1505.04597

[3] G. E. Trahey, S. W. Smith, and O. T. von Ramm, "Speckle Pattern Correlation with Lateral Aperture Translation: Experimental Results and Implications for Spatial Compounding," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 33, no. 3, pp. 257–264, 1986.

[4] C. Kervrann, J. Boulanger, and P. Coupé, "Bayesian non-local means filter, image redundancy and adaptive dictionaries for noise removal," *Proc. Conf. Scale-Space and Variational Meth. (SSVM' 07)*, pp. 520–532, 2007.

[5] P. Coupé, P. Hellier, C. Kervrann, and C. Barillot, "Nonlocal means-based speckle filtering for ultrasound images," *IEEE Transactions on Image Processing*, vol. 18, no. 10, pp. 2221–2229, 2009.

[6] R. F. Wagner, S. W. Smith, J. M. Sandrik, and H. Lopez, "Statistics of speckle in ultrasound b-scans," *IEEE Transactions on Sonics and Ultrasonics*, vol. 30, no. 3, pp. 156–163, May 1983.

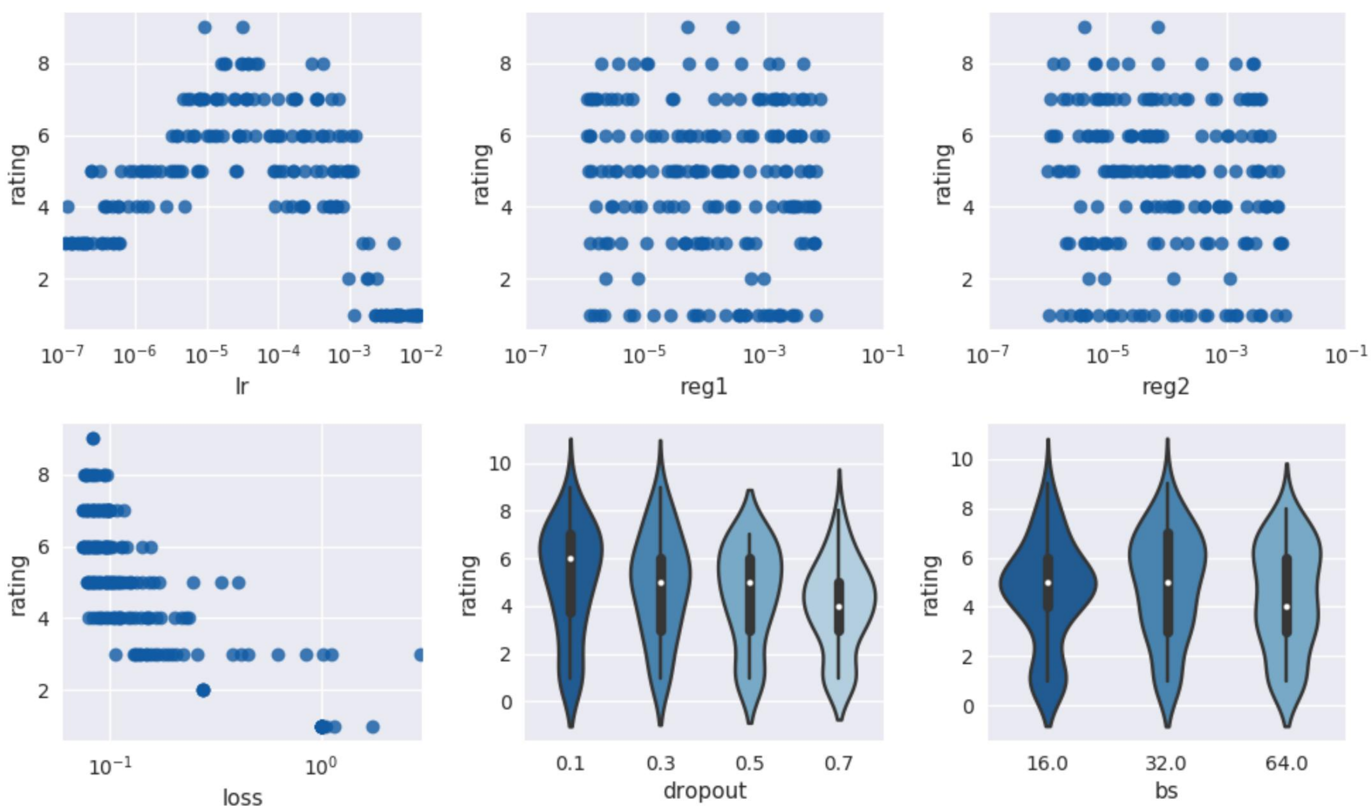

Fig. 6: Results of manual evaluation of hyperparameter tuning. About 60 networks were trained with randomized hyperparameters and were manually evaluated on perceptual improvement on 2 test set images. This was used to choose the best network after hyper parameter tuning, and to verify that the loss function accurately represents the desired transformation. From left to right on the top the ratings are plotting against: learning rate, weighting on the L1 regularization term and weighting on the L2 regularization term, on the bottom: the final loss, the dropout rate and the batch size. Dropout and Batch size plots are plotted using violin plots.