Program Digest, 2019 IEEE International Ultrasonics Symposium (IUS)
Glasgow, Scotland, October 6-9, 2019

Compact Convolutional Neural Networks for Ultrasound Beamforming

Zhanwen Chen
Department of Electrical Engineering and Computer Science
Vanderbilt University
Nashville, USA
zhanwen.chen@vanderbilt.edu

Adam Luchies and Brett Byram
Department of Biomedical Engineering
Vanderbilt University
Nashville, USA
{adam.c.luchies, brett.c.byram}@vanderbilt.edu

***Abstract*—We trained convolutional neural networks (CNNs) to suppress off-axis scattering in the short-time Fourier Transform (STFT) domain. Our training data were point target responses from simulated anechoic cysts. We used random neural architecture search to build CNN models with variable input formulations, layer sizes, and training hyperparameters. Our results showed that CNNs were easier to train, as they required fewer network weights to match the performance of fully-connected networks (FCNs). The best CNN models achieved comparable phantom CNRs with with two to three orders of magnitude fewer weights.**

***Index Terms*—ultrasound, beamforming, convolutional neural networks, deep learning**

I. Introduction

Recently, deep neural networks have been used for ultrasound beamforming by our group [1] and others [7] [9]. Our method applies deep neural network beamforming in the short-time Fourier Transform (STFT) domain in order to avoid having to train for different pulse shapes, depth dependent attenuation, and other pulse parameters that may vary across patients and even across probes as they age. Our early approach used classic fully connected deep networks (FCNs) trained with synthetic data. These beamformers are convolutional in nature insofar as the networks, including their weights, are reused through depth; however, fully connected layers are used to span the aperture dimension. We demonstrated that these models could work well [1].

Convolutional Neural Networks (CNNs) have seen widespread applications in Computer Vision, Natural Language Processing, and Medical Imaging alike. By taking advantage of localized parameter sharing, CNNs require fewer parameters and thus reduce the risk of overfitting. In our studies, we use CNNs for the same STFT-domain beamforming with similar performance but many fewer parameters, which makes them easier to train and less resource-intensive to deploy.

By applying CNNs to the same STFT data, we introduce convolution to the aperture dimension as well. CNNs could have the additional benefit of detecting local features such as aperture shapes. In addition, CNN beamformers, with fewer parameters, may be easier to train as shown in other application domains.

II. Methods

A. Training Data and Evaluation Scans

The FCN beamformer studied here was proposed previously by members of our group [6]. A collection of FCN is trained to operate on channel data in the frequency domain. Training data was generated from FieldII simulated point target responses [5]. The simulated ultrasonic array was based on the L7-4 (38 mm) linear array transducer. Point targets were randomly placed in an annular sector centered at the focal depth of the transducer array, using a process that we have described previously [6]. For the point targets inside the main lobe of the beam, the corresponding output was the same as the input; for those outside the main lobe of the beam, the corresponding output was a vector of zeros [6]. We used 10^5 examples for training and 10^4 for validation.

An ATL L7-4 (38 mm) linear array transducer was operated using a Verasonics system to scan a physical phantom. A cylindrical cyst having 5mm in diameter located at 7cm depth was scanned using a cross-sectional view at five different positions along the axial dimension. The image quality metrics used for evaluation purposes included contrast-to-noise ratio (CNR), contrast ratio (CR), and speckle SNR.

B. Convolutional Neural Networks

Since there were no existing CNNs for our task, we had to design custom CNN architectures. The CNN architectures that we designed were based on LeNet [4] and AlexNet [8] because these architectures commonly serve as baselines for other applications. We implemented CNNs with configurable input, output, layer sizes, with and without pooling, kernel size, number of kernels, number of fully-connected layers, and many others, using similar layer types and layer orders.

We also varied the optimizers and loss functions. Optimizers were chosen between Stochastic Gradient Descent (SGD) and Adaptive Moment Estimation (Adam). Each model could have either Mean Squared Error (MSE), Mean Absolute Error (L1), or Smooth Mean Absolute Error (Smooth L1) as its loss function.

Our neural architecture search (NAS) consisted of 1000 trained models. They were random instead of heuristic, as current studies show that sophisticated NAS algorithms generally fail to outperform random search [2]. In addition, studies

Hyperparameter	Value
Architecture Type	LeNet-like
Input Formulation	1x130x1
Using Max Pooling	True
Using Batch Normalization	True
Adding Gaussian Noise	True
Conv1 Kernel Size	17
Conv1 Number of Kernels	45
Conv1 Stride	1
Conv1 Dropout	0
Pool1 Kernel Size	2
Pool1 Stride	2
Conv2 Kernel Size	12
Conv2 Number of Kernels	35
Conv2 Stride	1
Conv2 Dropout	0.5149
Pool2 Kernel Size	2
Pool2 Stride	2
Fully-Connected (FC) Layers	2
FC Layers Width	109
Loss Function	Smooth Mean Absolute Error
Optimizer	Adam
Learning Rate	1.803e-4

TABLE I
ARCHITECTURE AND HYPERPARAMETERS FOR THE BEST CNN MODEL

show that random hyperparameter search outperformed grid search [3]. Since popular NAS libraries such as AutoKeras and NNI lacked support for regression tasks, we created our own framework for random NAS using constraint satisfaction with Prolog. The models were created and trained with PyTorch [10].

We investigated the effects of treating real and imaginary components as a flat array (1 by 130 by 1), as separate channels (1 by 65 by 2), and as a height dimension (2 by 65 by 1). We first concatenated the real and imaginery parts to have a single-channel 1D convolution. We then stacked the two parts as separate channels, resulting in a two-channel 1D convolution. Lastly, we treated inputs as narrow gray images that we convolve in both aperture and I-Q dimensions.

III. RESULTS

We found that LeNet-like CNNs featuring two convolutional layers and two or three fully-connected layers were effective. We also found that adding Gaussian noise in training was useful. Furthermore, it was better to concatenate the I and Q components as a single-channel 1D array or stack them as a two-channel 1D array than to stack them as a one-channel 2D array. Furthermore, almost all top-performing models used Adam instead of SGD as their optimizers. In terms of loss functions, L1 was ineffective compared with MSE and Smooth L1.

For phantom targets, the CNR was 5.46±0.45 dB, 5.57±0.20 dB, and 4.24±0.38 dB for the best CNN, the best FCN, and DAS, respectively. A t-test was used to compare the best CNN to the best FCN and the difference was not statistically significant (p-value=0.45). Our results suggest that CNNs produce equivalent results to FCNs and qualitative assessment suggests they may have a larger depth of field.

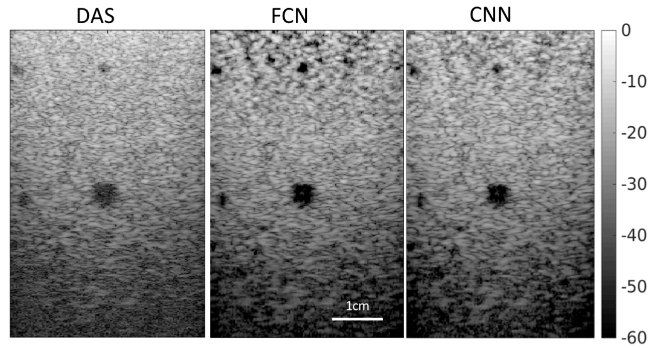

Fig. 1. For a phantom target, DAS has a CNR of 4.3994, FCN 5.5403, CNN 5.2946

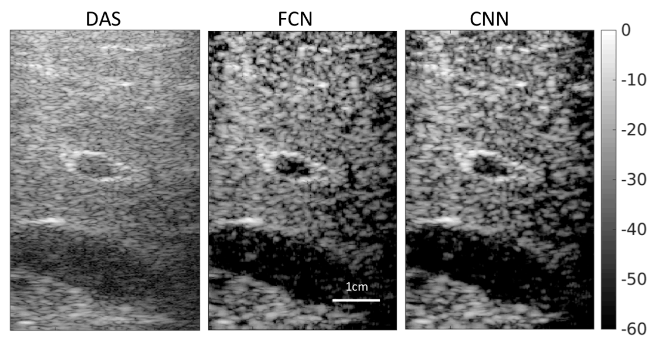

Fig. 2. For an in vivo target, DAS has a CNR of -14.982, FCN -0.80402, CNN -2.8361

IV. CONCLUSION

Based on the evaluation scans, we conclude that CNNs can match the performance of FCNs in suppressing off-axis scattering. In addition, CNNs have the added benefit of having fewer parameters, making training and deployment easier. Lastly, beamformed images indicate that CNNs may have a larger depth of field.

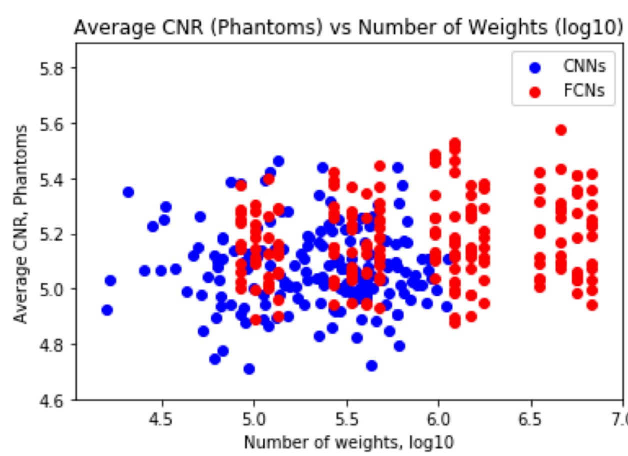

Fig. 3. The best CNNs tend to have fewer weights than the best FCNs

REFERENCES

[1] A. C. Luchies and B. C. Byram, Deep Neural Networks for Ultrasound Beamforming, IEEE Trans. on Med. Imag., vol. 37, pp. 20102021, 2018.
[2] L. Li and A. Talwalkar, "Random Search and Reproducibility for Neural Architecture Search,", Conference on Uncertainty in Artificial Intelligence, 2019.
[3] J. Bergstra and Y. Bengio, "Random Search for Hyper-Parameter Optimization," Journal of Machine Learning Research, vol. 13, pp. 281-305. 2012.
[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," Proceedings of the IEEE, Nov. 1998.
[5] J. A. Jensen, Field: A Program for Simulating Ultrasound Systems, in 10th Nordic-Baltic Conference on Biomedical Imaging, vol. 4, sup. 1, part 1, pp. 351353, Citeseer, 1996.
[6] A. C. Luchies and B. Byram, "Training Improvements for Ultrasound Beamforming with Deep Neural Networks," Physics in Medicine and Biology.
[7] L, L. Brickson, D. Hyun, and J. J. Dahl, "Reverberation Noise Suppression in the Aperture Domain Using 3d Fully Convolutional Neural Networks," 2018 IEEE International Ultrasonics Symposium (IUS)
[8] A., Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet Classification with Deep Convolutional Neural Networks," Advances in Neural Information Processing Systems, pp. 1097-1105, 2012.
[9] Y. H. Yoon, S. Khan, J. Huh and J. C. Ye, "Efficient B-Mode Ultrasound Image Reconstruction From Sub-Sampled RF Data Using Deep Learning," in IEEE Transactions on Medical Imaging, vol. 38, no. 2, pp. 325-336, Feb. 2019.
[10] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, Alban, L. Antiga, and A. Lerer, "Automatic Differentiation in PyTorch, " NIPS Autodiff Workshop, 2017.