UNFOLDING MODEL-BASED BEAMFORMING FOR HIGH QUALITY ULTRASOUND IMAGING

Christopher Khan[⋆]   *Ruud J. G. van Sloun*[†]   *Brett Byram*[⋆]

[⋆]Vanderbilt University, Nashville, TN, USA
[†]Eindhoven University of Technology, Eindhoven, Netherlands

ABSTRACT

Aperture Domain Model Image REconstruction (ADMIRE) is an advanced ultrasound beamforming method that uses a model-based approach to suppress sources of acoustic clutter and improve ultrasound image quality. However, it requires solving an ill-posed inverse problem for which regularization is utilized. As a result, the iterative nature of solving this problem is computationally expensive, and the choice of regularization bounds the fidelity of the obtained solution. Therefore, in this work, we pose ADMIRE as a sparse coding problem and unfold the iterations of the iterative shrinkage and thresholding algorithm (ISTA), training it end to end to yield learned ISTA (LISTA). This enables effective tailoring of the solver to the specific data distribution and task at hand, while enjoying higher data efficiency and robustness than generic deep learning methods. Evaluation of our proposed method on both simulated cyst data and *in vivo* liver data demonstrates its potential to outperform conventional ADMIRE.

Index Terms— ADMIRE, Beamforming, Deep Learning, Sparse Coding, ISTA

1. INTRODUCTION

Aperture Domain Model Image REconstruction (ADMIRE) is a model-based ultrasound beamformer that suppresses sources of acoustic clutter [1, 2, 3]. It involves computing the short-time Fourier transform (STFT) of the received ultrasound data and then modeling the aperture domain data (i.e., the data across the transducer elements) for each frequency in a given STFT window as a linear superposition of acoustic wavefronts returning from different scattering locations. This is demonstrated in (1), where y is the aperture domain data for a given frequency, X is a model matrix containing predictors corresponding to the received aperture domain data localized in time and frequency for different scattering locations, and β contains the model coefficients.

$$y = X\beta \tag{1}$$

To estimate β for this inverse problem, linear regression is utilized. However, the inverse problem is ill-posed due to the number of predictors being greater than the number of observations, meaning that regularization must be utilized to obtain a solution. Due to this, an elastic-net penalty was used in previous works [1, 2, 3]. Once $\hat{\beta}$ is obtained, only predictors that are within a region of interest (ROI) are utilized to reconstruct the decluttered data as shown in (2), and the inverse short-time Fourier transform is then computed.

$$y_{ROI} = X_{ROI}\beta_{ROI} \tag{2}$$

Although ADMIRE is able to achieve significant improvements in ultrasound image quality and has been implemented in real time [1, 2, 3, 4, 5, 6, 7, 8], it still suffers from a high computational load and latency due to the iterative nature of solving its inverse problem. Moreover, using an elastic-net penalty provides just one solution of infinitely many, meaning that there are potentially better strategies for performing regularized model fits. Previous work by Gregor and LeCun has shown that the iterative shrinkage and thresholding algorithm (ISTA) for sparse coding (i.e., solving $y = X\beta$ for sparse code β) can be learned in order to perform fast approximations of sparse coding [9]. This learned version of the algorithm (LISTA) was developed by training a sparse encoder using information specific to the target data-distribution. Therefore, in this work, we propose adapting this framework and training a deep neural network (DNN) sparse encoder for performing the model fits of ADMIRE. Deep learning has already been used for a variety of ultrasound imaging applications [10, 11, 12, 13, 14], and this data-driven approach provides the opportunity to both improve the computational efficiency of ADMIRE and obtain a more optimal solution in terms of ultrasound image quality.

2. METHODS

2.1. Deep Neural Network Sparse Encoder Architecture

Regarding the architecture of our deep neural network sparse encoder, we unfold the iterations of the ISTA algorithm as a feedforward neural network. To obtain the unfolding procedure, we will first derive the model coefficient update corresponding to one iteration of ISTA. Starting from the linear model for ADMIRE shown in (3), we will assume that the

residual errors of the model follow a Gaussian distribution given by $\boldsymbol{\epsilon}_i \sim \mathcal{N}(0, \sigma^2)$, meaning that $\boldsymbol{y}$ also follows a Gaussian distribution given by $\boldsymbol{y}_i \sim \mathcal{N}(\boldsymbol{X}_i\boldsymbol{\beta}, \sigma^2)$. This results in the probability density function of $\boldsymbol{y}_i$ given $\boldsymbol{\beta}$ being expressed as (4). Moreover, for the model coefficients contained in $\boldsymbol{\beta}$, we will assume that the prior distribution follows a Laplacian distribution with a mean of 0 and a scale parameter of $\frac{1}{\lambda}$, giving the probability density function of $\boldsymbol{\beta}_j$ as (5).

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{3}$$

$$p(\boldsymbol{y}_i \mid \boldsymbol{\beta}) = \frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-(\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta})^2}{2\sigma^2}} \tag{4}$$

$$p(\boldsymbol{\beta}_j) = \frac{\lambda}{2}e^{-\lambda|\boldsymbol{\beta}_j|} \tag{5}$$

The maximum a posteriori (MAP) estimate of $\boldsymbol{\beta}$ is given by (6), and using Bayes' theorem along with the fact that $p(\boldsymbol{y})$ amounts to a scaling factor, it can be expressed as (7).

$$\hat{\boldsymbol{\beta}} = \arg\max_{\boldsymbol{\beta}} p(\boldsymbol{\beta} \mid \boldsymbol{y}) \tag{6}$$

$$\hat{\boldsymbol{\beta}} = \arg\max_{\boldsymbol{\beta}} p(\boldsymbol{y} \mid \boldsymbol{\beta})p(\boldsymbol{\beta}) \tag{7}$$

Using (4), (5), and (7), the MAP estimate can be written as shown in (8).

$$\hat{\boldsymbol{\beta}} = \arg\max_{\boldsymbol{\beta}} \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma}e^{\frac{-(\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta})^2}{2\sigma^2}} \prod_{j=1}^{P} \frac{\lambda}{2}e^{-\lambda|\boldsymbol{\beta}_j|} \tag{8}$$

Rather than maximize this formula, we can minimize the negative logarithm and simplify further to obtain (9).

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{N} \frac{(\boldsymbol{y}_i - \boldsymbol{X}_i\boldsymbol{\beta})^2}{2\sigma^2} + \lambda \sum_{j=1}^{P} |\boldsymbol{\beta}_j| \tag{9}$$

Multiplying through by σ^2 and setting $\lambda = \lambda\sigma^2$, (9) is reduced to (10), which is the objective function for least-squares regression with L1-regularization.

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|_1 \tag{10}$$

To minimize this objective function, the proximal gradient descent algorithm can be utilized. The least-squares portion of (10) is convex and differentiable, so we can first obtain a gradient descent update for $\boldsymbol{\beta}$ in the form of (11), where μ is the step size.

$$\boldsymbol{z}^{(k)} = \boldsymbol{\beta}^{(k)} - \mu\boldsymbol{X}^\top(\boldsymbol{X}\boldsymbol{\beta}^{(k)} - \boldsymbol{y}) \tag{11}$$

To account for the L1-regularization term that is convex but not smooth, we can apply a soft-thresholding proximal operator $h_\lambda(z)$ to the gradient descent update.

$$h_\lambda(z) = \begin{cases} z - \lambda & \text{if } z > 0 \text{ and } \lambda < |z| \\ z + \lambda & \text{if } z < 0 \text{ and } \lambda < |z| \\ 0 & \text{if } \lambda \geq |z| \end{cases}$$

Therefore, the update for a single iteration of the ISTA algorithm [9] is given by (12), where $\boldsymbol{S} = \boldsymbol{I} - \mu\boldsymbol{X}^\top\boldsymbol{X}$.

$$\boldsymbol{\beta}^{(k+1)} = h_\lambda(\boldsymbol{S}\boldsymbol{\beta}^{(k)} + \mu\boldsymbol{X}^\top\boldsymbol{y}) \tag{12}$$

Using (12), we can represent our network architecture as shown in Fig. 1, where the number of layers corresponds to a fixed number of ISTA iterations. The input to this network is $\boldsymbol{X}^\top\boldsymbol{y}$, and the output is $\hat{\boldsymbol{\beta}}$, which contains the predicted model coefficients for a given model fit. Note that in this diagram, $\boldsymbol{S}$ represents a fully connected linear layer with weights that can be learned rather than representing $\boldsymbol{I} - \mu\boldsymbol{X}^\top\boldsymbol{X}$, and instead of using $\mu\boldsymbol{X}^\top\boldsymbol{y}$, we simply use $\boldsymbol{X}^\top\boldsymbol{y}$. Moreover, a sigmoidal soft-thresholding function with a learnable threshold is used as the nonlinear activation function. The equation for this function is shown in (13), where z is the input to the function and γ is the learnable threshold. We use this function as a smooth approximation to the soft-thresholding function $h_\lambda(z)$. By using this approximation, the function can be differentiated at all points, which is important when using gradient descent methods for training a neural network.

$$\frac{z}{1 + e^{-(|z|-\gamma)}} \tag{13}$$
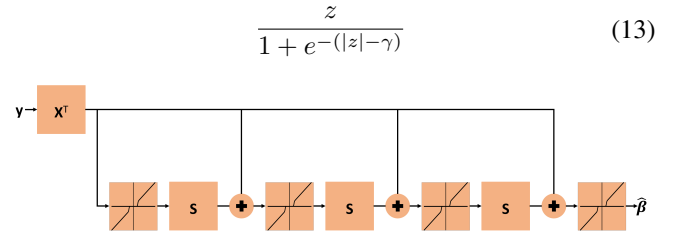
Fig. 1: Diagram of the deep neural network sparse encoder architecture for performing the model fits of ADMIRE.

2.2. Training Strategies

To train the network to learn the ISTA algorithm, one frame of human *in vivo* liver data acquired using a Verasonics (Verasonics, Kirkland, WA) Vantage 128 ultrasound research system and a C5-2 curvilinear transducer array was used. This data was processed with ADMIRE, and the model fits were performed by using the ISTA algorithm. For a given model fit, ISTA was performed until the maximum model coefficient change across all of the model coefficients became less than 1×10^{-4}. Several thousand model fits were performed, and this data was used to create training and validation data sets. In terms of the training scheme, a mean-squared error (MSE) loss function between the predicted model coefficients of the network for a given training iteration and those produced by the ISTA algorithm was used.

Training was performed using PyTorch [15], and the Adam optimizer [16] was used with a learning rate of 0.001. In addition, different training cases were performed in which the network architecture was modified to change the number of fixed DNN ISTA iterations, where one iteration involves applying a fully connected linear layer S, adding $X^{\top}y$, and applying (13). Once trained, the different cases were evaluated on five test frames of *in vivo* liver data to compare them. Fourth-order blind identification independent component analysis (FOBI-ICA) [7, 8] was applied to the ADMIRE model matrices to reduce their sizes.

In addition, rather than learning to mimic ISTA, our framework can be adapted to learn unique, data-driven model fitting schemes. Specifically, we can train with targets that are better tailored to achieving the end goal of ADMIRE, which is suppressing acoustic clutter. To obtain these targets, we used ultrasound data simulated using the Field II simulation package [17, 18]. This data consisted of hypoechoic and anechoic cysts with added reverberation clutter (simulated using a pseudo nonlinear approach [19]) and noise, and the benefit of using simulated data is that the ground-truth clean and clutter aperture domain STFT signals are known. Using this data, we trained the network using the following schemes.

Scheme 1: The network was trained to best reconstruct the clutter signal. This was done by taking the predicted model coefficients of the network for a given training iteration, using them along with the ADMIRE model matrix X to compute the predicted clutter signal as $\hat{y}_{clutter} = X_{clutter}\hat{\beta}_{clutter}$, and then computing the MSE loss between the predicted clutter signal and the ground-truth clutter signal. For the purposes of training, the reverberation clutter and noise were considered to be the clutter signal.

Scheme 2: The network was trained to best reconstruct the clean signal using steps similar to Scheme 1.

Scheme 3: The network was trained to best reconstruct both the clutter and clean signals. This was done by computing the predicted clutter and clean signals as well as computing the predicted combined signal given by $\hat{y} = X\hat{\beta}$ for a given iteration and then using a joint loss function consisting of the addition of the clutter signal MSE loss, the clean signal MSE loss, and the combined signal MSE loss.

For each scheme, the DNN architecture was set to three fixed DNN ISTA iterations as shown in Fig. 1. Once trained, the networks were evaluated on five test cyst simulation data sets with added reverberation clutter and noise as well as one test frame of *in vivo* liver data. Other methods including delay-and-sum (DAS) beamforming, ADMIRE using cyclic coordinate descent to perform linear regression with elastic-net regularization ($\alpha = 0.9$), ADMIRE using ISTA to perform linear regression with L1-regularization, and a DNN that was trained to mimic the ISTA algorithm were also evaluated. FOBI-ICA was applied to the ADMIRE model matrices. As an additional study, the different DNN methods were trained again, but the number of fixed DNN ISTA iterations was var-

ied. Once trained, the normalized image envelope reconstruction error between each method and the ground-truth clean signal for each test cyst data set was computed and averaged as a function of the number of fixed DNN ISTA iterations.

3. RESULTS

3.1. Learning the ISTA Algorithm (LISTA)

Fig. 3 shows the MSE between the coefficients predicted by the DNN and those produced by the ISTA algorithm for the test *in vivo* liver data. It also shows the MSE between the reconstructed sets of normalized envelope data for the DNN and the ISTA algorithm. Based off this plot, it appears as though the number of fixed DNN ISTA iterations does not make a significant difference in terms of model coefficient prediction accuracy. Moreover, it shows that producing the lowest MSE for coefficient prediction does not necessarily result in the lowest MSE for normalized envelope data reconstruction. This is most likely because other operations for ADMIRE are performed after model fitting, such as reconstructing the decluttered aperture domain frequency data and computing the inverse short-time Fourier transform. In terms of computational efficiency, performing the model fits using the ISTA algorithm required an average of approximately 6,927 iterations. As a result, using a DNN can dramatically reduce the computational load of model fitting. Note that the step size used for the ISTA algorithm for each fit was $\frac{1}{L}$, where L was set to be slightly larger than the largest eigenvalue of $X^{\top}X$.

3.2. Learning Custom, Data-Specific Model Fitting

Table I shows image metrics obtained with different methods as well as the normalized image envelope reconstruction error between each method and the ground-truth clean signal for the test cyst data sets. The results for the one frame of *in vivo* liver data are shown in Fig. 2. For the DNN trained to mimic ISTA, it was trained using only the simulated data, and this was compared to the DNN with three fixed ISTA iterations that was trained using one frame of *in vivo* liver data in the previous section. The latter had better performance, so it is the one that is shown in Table I and Fig. 2. For the cyst results, using the DNN that was trained to best reconstruct the clutter signal produced image metrics that had the overall greatest similarity to the ground-truth clean signal. It also produced the highest contrast ratio value for the *in vivo* liver data frame. For this method as well as the DNN with three loss functions, the performance was better when reconstructing the clutter signal and then subtracting it from the aperture domain frequency data to obtain the decluttered signal rather than directly reconstructing it from the ROI predictors. This is similar to iterative ADMIRE [20]. Moreover, Fig. 4 shows the normalized image envelope reconstruction error between each DNN method and the ground-truth clean signal for the

8684

| Method	Contrast Ratio (dB)	Contrast-to-Noise Ratio (dB)	Generalized Contrast-to-Noise Ratio	Mean-Squared Error
Ground-Truth Clean Signal	25.75 ± 0.92	4.73 ± 0.70	0.98 ± 0.01	—
Delay-and-Sum (DAS)	19.07 ± 1.02	4.16 ± 0.74	0.88 ± 0.04	0.001 ± 0.000
ADMIRE (Cyclic Coordinate Descent Algorithm)	23.53 ± 1.04	3.29 ± 0.91	0.92 ± 0.03	0.006 ± 0.002
ADMIRE (ISTA Algorithm)	23.03 ± 1.06	3.19 ± 0.89	0.92 ± 0.03	0.006 ± 0.002
ADMIRE (ISTA DNN)	24.06 ± 1.21	2.60 ± 0.82	0.92 ± 0.03	0.008 ± 0.002
ADMIRE (DNN Clutter Signal Loss Function)	25.63 ± 1.33	3.17 ± 0.75	0.94 ± 0.03	0.006 ± 0.002
ADMIRE (DNN Clean Signal Loss Function)	20.46 ± 1.26	3.72 ± 0.72	0.89 ± 0.03	0.005 ± 0.001
ADMIRE (DNN Three Loss Functions)	22.42 ± 0.94	3.16 ± 0.80	0.91 ± 0.03	0.006 ± 0.001
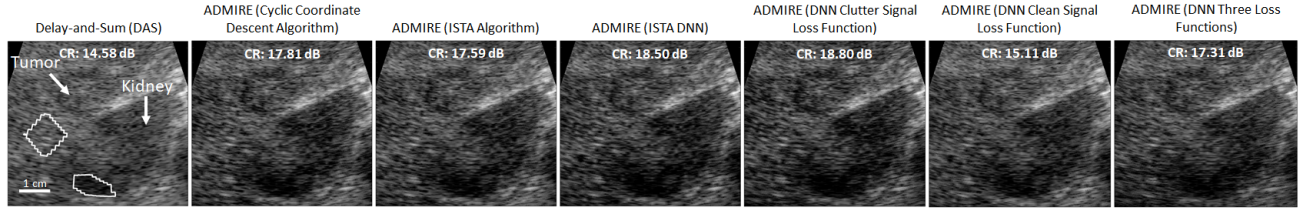
Fig. 2: *In vivo* liver data images produced using different methods. All images are displayed with a dynamic range of 60 dB. The masks used for computing the contrast ratio values are displayed.
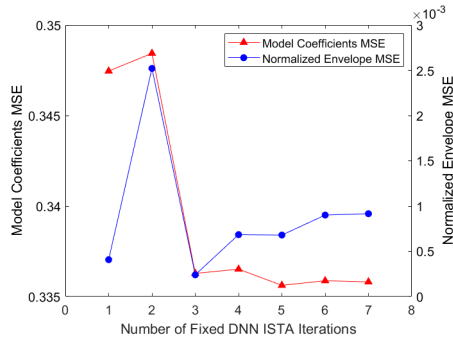

Fig. 3: Model coefficient prediction error and envelope reconstruction error between the DNN and the ISTA algorithm as a function of the number of fixed DNN ISTA iterations. For each iteration case, the MSE values were calculated for five test frames of *in vivo* liver data and then averaged.

Fig. 4: Envelope reconstruction error between each DNN method and the cyst ground-truth clean signal as a function of the number of fixed DNN ISTA iterations. The MSE was computed for each cyst data set and then averaged over the five test cyst data sets. The ISTA algorithm result is also provided. For the ISTA DNN case, the results are for training with simulated data only.

five test cyst data sets as a function of the number of fixed DNN ISTA iterations. This plot shows that increasing the number of fixed DNN ISTA iterations does not necessarily decrease the normalized image envelope reconstruction error.

4. DISCUSSION

The results show that using a DNN can produce computationally efficient approximations of coefficients for the model fits of ADMIRE. Moreover, this approach can be tailored using data-specific information in order to learn custom model fitting schemes that have the potential to outperform conventional ADMIRE in terms of ultrasound image quality. In addition, compared to an end-to-end DNN approach, our approach maintains the model-based intuition of ADMIRE because we are only using a DNN to determine an optimal linear combination of features produced by our own physics-based model.
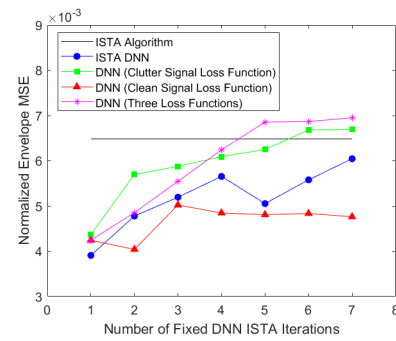
5. CONCLUSION

Using a deep neural network sparse encoder provides a computationally efficient and data-driven approach to solving the ADMIRE inverse problem, and it has the potential to outperform conventional ADMIRE in terms of ultrasound image quality. Future work includes utilizing a larger and more variable training data set and optimizing hyperparameters.

6. ACKNOWLEDGEMENT

7. REFERENCES

[1] B. Byram and M. Jakovljevic, "Ultrasonic multipath and beamforming clutter reduction: a chirp model approach," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 61, no. 3, pp. 428–440, 2014.

[2] B. Byram, K. Dei, J. Tierney, and D. Dumont, "A model and regularization scheme for ultrasonic beamforming clutter reduction," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 62, no. 11, pp. 1913–1927, 2015.

[3] K. Dei and B. Byram, "The impact of model-based clutter suppression on cluttered, aberrated wavefronts," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 64, no. 10, pp. 1450–1464, 2017.

[4] C. Khan, K. Dei, S. Schlunk, K. Ozgun, and B. Byram, "A real-time, gpu-based implementation of aperture domain model image reconstruction," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 68, no. 6, pp. 2101–2116, 2021.

[5] C. Khan and B. Byram, "Genre (gpu elastic-net regression): A cuda-accelerated package for massively parallel linear regression with elastic-net regularization," *Journal of Open Source Software*, vol. 5, no. 54, pp. 2644, 2020.

[6] K. Dei, S. Schlunk, and B. Byram, "Computationally efficient implementation of aperture domain model image reconstruction," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 66, no. 10, pp. 1546–1559, 2019.

[7] J. Shlens, "A tutorial on independent component analysis," *CoRR*, vol. abs/1404.2986, 2014.

[8] J. F. Cardoso, "Source separation using higher order moments," in *International Conference on Acoustics, Speech, and Signal Processing,*. IEEE, 1989, pp. 2109–2112.

[9] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proceedings of the 27th international conference on international conference on machine learning*, 2010, pp. 399–406.

[10] R. J. G. van Sloun, R. Cohen, and Y. C. Eldar, "Deep learning in ultrasound imaging," *Proceedings of the IEEE*, vol. 108, no. 1, pp. 11–29, 2020.

[11] A. C. Luchies and B. C. Byram, "Deep neural networks for ultrasound beamforming," *IEEE Transactions on Medical Imaging*, vol. 37, no. 9, pp. 2010–2021, 2018.

[12] J. Tierney, A. Luchies, C. Khan, J. Baker, D. Brown, B. Byram, and M. Berger, "Training deep network ultrasound beamformers with unlabeled in vivo data," *IEEE Transactions on Medical Imaging*, pp. 1–1, 2021.

[13] B. Luijten, R. Cohen, F. J. de Bruijn, H. A. W. Schmeitz, M. Mischi, Y. C. Eldar, and R. J. G. van Sloun, "Adaptive ultrasound beamforming using deep learning," *IEEE Transactions on Medical Imaging*, vol. 39, no. 12, pp. 3967–3978, 2020.

[14] D. Hyun, L. L. Brickson, K. T. Looby, and J. J. Dahl, "Beamforming and speckle reduction using neural networks," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 66, no. 5, pp. 898–910, 2019.

[15] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds., pp. 8024–8035. Curran Associates, Inc., 2019.

[16] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[17] J. A. Jensen, "Field: A program for simulating ultrasound systems," in *10TH NORDICBALTIC CONFERENCE ON BIOMEDICAL IMAGING, VOL. 4, SUPPLEMENT 1, PART 1: 351–353*. Citeseer, 1996.

[18] J. A. Jensen and N. B. Svendsen, "Calculation of pressure fields from arbitrarily shaped, apodized, and excited ultrasound transducers," *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, vol. 39, no. 2, pp. 262–267, 1992.

[19] B. Byram and J. Shu, "A pseudo non-linear method for fast simulations of ultrasonic reverberation," in *Medical Imaging 2016: Ultrasonic Imaging and Tomography*. International Society for Optics and Photonics, 2016, vol. 9790, p. 97900U.

[20] S. Schlunk, K. Dei, and B. Byram, "Iterative model-based beamforming for high dynamic range applications," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 68, no. 3, pp. 482–493, 2020.