3Pb5-15

High-Resolution Ultrasound Imaging by Adaptive Compounding Using Deep Learning

深層学習を用いる適応的なコンパウンドによる高分解能超音波イメージング

Emi Aiura[1†], Norio Tagawa[1] ([1]1Tokyo Metropolitan University”)

相瀬永実 [1†]，田川憲男 [1] ([1] 東京都立大学)

1. Introduction

Ultrasound imaging is widely used in the medical field because it is noninvasive and can be imaged in real time. Typical method of beamforming for generating an ultrasonic image is a delay-and-sum (DAS) beamformer that corrects the delay time due to the difference in the propagation path of the echo received by each tansdiucer and adds them together. Nowadays, minimum variance (MV) beamformer is often used to determine the DAS weights so that the variance of beamforming result is minimized. Compared to the DAS beamformer, the image quality of the MV beamformer can be improved, but it has a problem that the calculation amount is large and it takes time to find the optimum weight for each pixel.

We proposed a high-resolution ultrasound imaging method called FPWC-MVDR (frequency and plane-wave compounding-minimum variance distortionless response) using an adaptive compound of ultrasound transmission angle and subbands [1]. Like the MV beamformer, FPWC-MVDR is a method that determines the weight for each pixel, so that the amount of calculation is large and real-time processing is difficult. Therefore, this study aims to generate images in a short time using deep learning. The result of FPWC-MVDR is used as an annotation for deep learning. Two networks, one that learns the weight calculation of each pixel and the other that directly learns beamforming, are constructed and compared.

2. Method

2.1 FPWC-MVDR

The narrowband chirp signal is transmitted multiple times with the center frequency randomly changed for each transmission angle, and the received echo is delayed by an appropriate time. The subband variance-covariance matrix is estimated for each pixel, and then the weight of the subband compound is determined by the minimum variance criterion. The weight is used to reduce the frequency component for each transmission angle, and then the angular variance-covariance matrix is

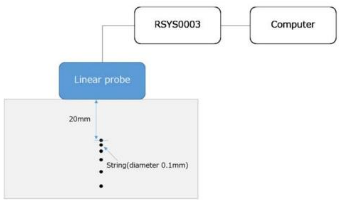

Fig. 1 Experimental setting model.

2.2 Simulation

In this experiment, the transmit and receive sequences used the experimental platform for medical ultrasound equipment (Microsonic RSYS0003) at a sampling rate of 31.25 MHz. The transducer in this experiment has 64 elements and an element pitch of 0.315 mm. A linear array probe (Nihon Dempa Kogyo T0-1599) was also used. The center frequency of this probe is 7.5MHz and the specific bandwidth is 70%. Signal processing was performed offline using MATLAB software.

Figure 1 shows the experimental setting. We present the experimental results obtained using a soft tissue-mimicking phantom (Kyoto Kagaku US-2 multi-purpose phantom N-365), with a speed of sound of 1432 ms^{-1} (25 °C) and attenuation of 0.59 dB cm^{-1} MHz^{-1}. The phantom contains six string wires with the same diameter of 0.1 mm and the distances between these wires are 1.0, 2.0, 3.0, 4.0 and 5.0 mm as measured from the side closest to the phantom.

3. Network Architecture

The networks used in this study were implemented in Python using the Keras API with a tensorflow (Google , CA, USE) backend. For the annotation data, the value of 3300 pixels as a result of FPWC-MVDR shown in **Figure 2** was used. For weight learning, 264 values per pixel (product of 33 angles and 8 sub-bands) were used, and Beamforming learning used one value per pixel. Both are results for RF signals.

Based on [2], in weight learning, the fully connected layers that outputs N nodes and N / 4

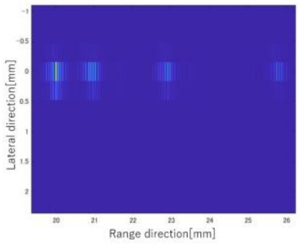

Fig. 2 B-mode images of the result of FPWC-MVDR.

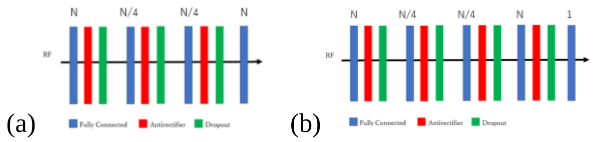

Fig. 3 The overview of network model: (a) weight learning; (b) beamforming learning.

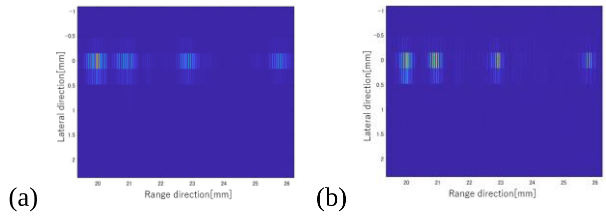

Fig. 4 B-mode images: (a)outputs of weight learning; (b) outputs of beamforming learning.

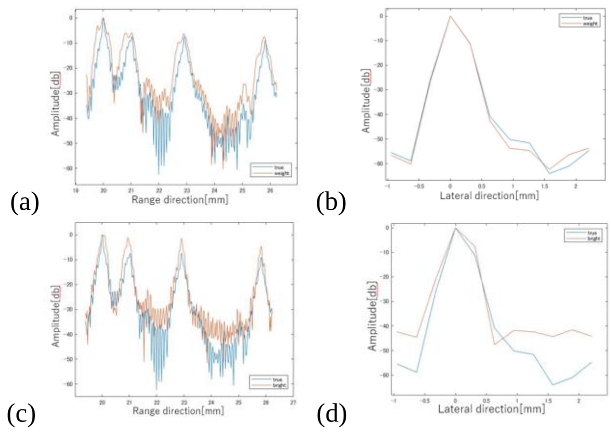

Fig. 5 Amplitude distribution profiles: (a) range direction of Fig. 4 (a); (b) lateral direction of Fig. 4 (a); (c) range direction of Fig. 4 (b); (d) lateral direction of Fig. 4 (b).

nodes are used and in beamforming learning, it adds a layer that outputs 1 node at last. The overview of each network is shown in **Figure 3**. To prevent network overfitting, a dropout layer is applied between each fully connected layer with a probability of 0.2. Training optimization uses the Adam optimizer with a learning rate of 0.001.

In deep learning, the output value can be limited by using various activation functions. The ReLU function that outputs 0 when the input value is 0 or less and outputs the input value as it is when it is positive value is often used in the field of deep learning. Since the gradient disappearance may occur when the ReLU is used, an activation function called an anti-rectification layer, which is a combination of the ReLU and L2 normalization, is used in this study. By combining the ReLU with the normalization term, gradient disappearance can be prevented and overfitting of the constructed model can be suppressed.

During learning, a loss function is used to reduce the error between the predicted value and the correct value. The loss function is a function used to improve the prediction accuracy, and the network parameters are updated to minimize the loss function. In this study, the parameters are updated using the mean-squared-error (MSE), which is a typical loss function. In weight learning, a function that brings the sum of weights closer to 1 is combined with MSE.

4. Result and Conclusion

We output an image using the data used for training and check whether the network is training. The output results are shown in **Figure 4** and the prpfiles are shown in **Figure 5**. **Table 1** shows the average value of the full width at half maximum values. From these, it can be confirmed that the network can be learned, and that the beamforming learning has a narrower half-value width than the weight learning, and the image is clear. In terms of time, it took about 15038 seconds to create an image of FPWC-MVDR, about 142 seconds to learn weights, and about 166 seconds to learn brightness. This time, the learning time was short because the amount of data was small, but the weight learning was even shorter.

Table. 1 The average of the full width at half maximum values of FPWC-MVDR, output of weight learning, beamforming learning

	FPWC	brightness	weight
range	0.173	0.21	0.378
lateral	0.205	0.22	0.214

In the future, we will increase the amount of data to improve the accuracy of the networks and confirm the versatility of the network by using the data not used for learning.

References

1. J. Zheng, N. Tagawa, M. Yoshizawa, and T. Irie: Jpn. J. Appl. Phys., **60** (2021) SDDB08
2. B. Luijten, R. Cohen, F. J. de Bruijn, H.A.W. Schmeitz, M. Mischi, Y.C. Eldar and R.J.G. van Sloun: ICASSP, (2019) 1333-1337.