INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Hearing-Impaired Bio-Inspired Cochlear Models for Real-Time Auditory Applications

Arthur Van Den Broucke[1], *Deepak Baby*[2], *Sarah Verhulst*[1]

[1]Hearing Technology @ WAVES, Dept. of Information Technology, Ghent University, Belgium
[2]Idiap Research Institute, Martigny, Switzerland

{Arthur.VanDenBroucke, s.verhulst}@UGent.be

Abstract

Biophysically realistic models of the cochlea are based on cascaded transmission-line (TL) models which capture longitudinal coupling, cochlear nonlinearities, as well as the human frequency selectivity. However, these models are slow to compute (order of seconds/minutes) while machine-hearing and hearing-aid applications require a real-time solution. Consequently, real-time applications often adopt more basic and less time-consuming descriptions of cochlear processing (gammatone, dual resonance nonlinear) even though there are clear advantages in using more biophysically correct models. To overcome this, we recently combined nonlinear Deep Neural Networks (DNN) with analytical TL cochlear model descriptions to build a real-time model of cochlear processing which captures the biophysical properties associated with the TL model. In this work, we aim to extend the normal-hearing DNN-based cochlear model (CoNNear) to simulate frequency-specific patterns of hearing sensitivity loss, yielding a set of normal and hearing-impaired auditory models which can be computed in real-time and are differentiable. They can hence be used in backpropagation networks to develop the next generation of hearing-aid and machine hearing applications.

Index Terms: hearing-impairment, real-time auditory modeling, deep neural networks, transfer learning, machine hearing

1. Introduction

The transmission-line (TL) cochlear model used in [1] is an example of a biophysically accurate [2] [3] cochlear model which captures signature cochlear mechanic properties: frequency and level-dependence of cochlear filter tuning, a level-dependent compressive nonlinearity and longitudinal coupling of the cochlear filters. However, solving TL models requires a long computational time (time-domain solution of hundreds of coupled ODE) and this complexity makes these TL models not usable for real-time applications (e.g. machine hearing, robotics, hearing-aids). Real-time applications are hence more drawn towards more basic models of auditory filtering, since these models (e.g., gammatone [4] [5], dual resonance nonlinear [6] and CARFAC [7]) deliver a real-time ($<$10 ms) description of the cochlear processing stage, albeit in a biophysically less correct manner. Because we have recently demonstrated that speech-enhancement becomes more robust when TL models are used as frontends [8], and can assume that hearing-aid signal processing development will be more accurate when using models which resemble the pathological ear accurately, there is a clear need to attempt to solve biophysically-inspired TL cochlear models in real-time.

To this end, we recently developed the CoNNear model [9], which offers a deep convolutional neural network (CNN) description of the computations described by the TL cochlear model. This framework was trained on input-output pairs of 70 dB SPL speech inputs (TIMIT speech corpus [10]) and their corresponding TL reference model output (basilar membrane displacement patterns, or cochlear filter outputs with center frequencies covering the hearing range). Analysis showed that this DNN-based cochlear model can be computed in real-time (7 ms computational time per frame (2560 samples with a sample frequency of 20 kHz) on a NVIDIA GTX1080 GPU) and that it performed well on basic auditory stimuli of various stimulus levels and frequencies which assess the coupling, tuning, nonlinearity and distortion properties of cochlear processing [9].

CoNNear was developed using a TL cochlear model of a normal-hearing (NH) individual and hence returned a "CoNNear" of an audiometrically normal hearing person. However, by adjusting a number of parameters in the reference TL model, it is also possible to render its processing hearing-impaired (HI) to simulate cochlear processing associated with flat and sloping audiograms [1]. This paper investigates to which extent it is possible to obtain HI variants of the CoNNear model using the same architecture as adopted for the NH CoNNear model. This would yield real-time individualized auditory processing frontends which incorporate frequency-specific patterns of outer hair cell damage and which can be used to replace slow to compute nonlinear TL models in several auditory applications (e.g., real-time hearing-aid signal processing or pathological speech recognition).

2. Methods

2.1. CoNNear architecture

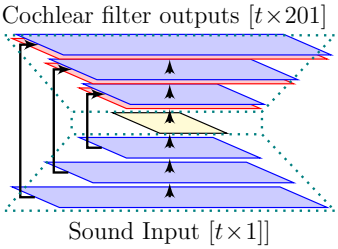

Figure 1: ***CoNNear architecture [9].*** *The CoNNear architecture is an auto-encoder, convolutional neural network framework which is connected using strided convolutions between layers, and skip-connections. It maps the time-domain sound input (bottom) to 201 time-domain cochlear filter outputs of different center frequencies (CF) (top). The depicted model has four encoding and decoding layers and uses a tanh activation function between the layers.*

http://dx.doi.org/10.21437/Interspeech.2020-2818

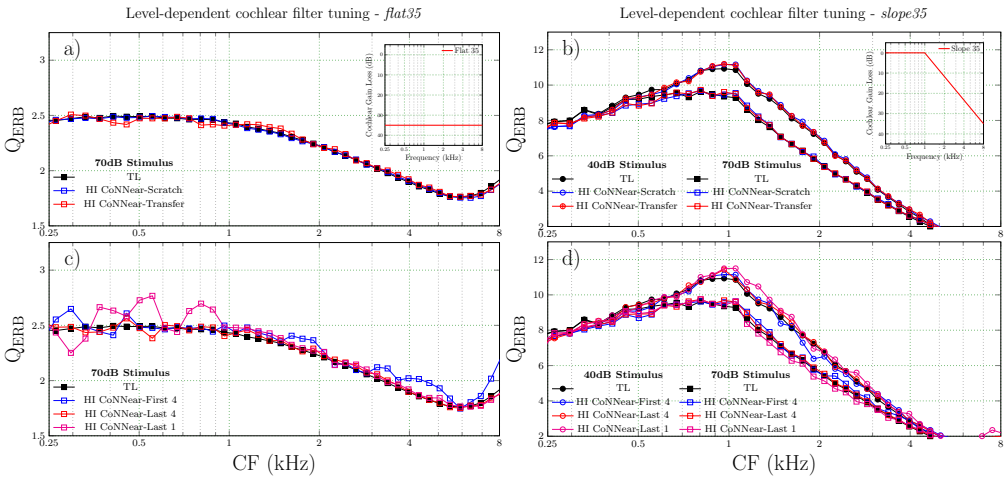

Figure 2: ***Level-dependent cochlear filter tuning (Q_{ERB}) for trained HI CoNNear models - flat35 and slope35 hearing loss profiles.*** *Q_{ERB} curves determined from the equivalent-rectangular bandwidth (ERB) [2], i.e., the bandwidth of a rectangular filter, at a certain characteristic frequency (CF), that passes the same energy underneath as the power spectrum of the impulse response to a 100 μs click of different intensities (40 and 70 dB peSPL).* ***(top)*** *Simulations are shown for the HI TL model (black) and two trained HI CoNNear models (scratch and transfer learning) for a flat35 HI profile (left) and a slope35 HI profile (right).* ***(bottom)*** *Tuning curves for HI CoNNear models obtained by transfer learning using only a selection of layers which is made trainable.*

The architecture we will use to develop HI CoNNear models will correspond to that of the NH CoNNear [9] (Fig.1). CoNNear makes use of an encoder-decoder structure: an audio input is first processed by an encoder (comprised of four CNN layers), which encodes the audio signal into a condensed representation. These CNN layers reduce the temporal dimension by half after every layer using strided convolutions. The encoded representation is then mapped to the corresponding BM displacements (i.e. time-domain outputs of the cochlear filters with CFs spanning the human hearing range) using a four layer decoder which restores the temporal dimension to the starting dimension. The original 401 channel output of the reference TL model [1] was downsampled by a factor of 2 to yield 201 CoNNear cochlear filter outputs with CFs between 100 Hz and 12 kHz [11] in the final CNN layer. U-shaped skip connections, that bypass the encoder and decoder layers, are also added to prevent the loss of temporal alignment and phase of the speech, which is important for speech intelligibility [12]. Other relevant network hyperparameters are [9]: the number of filters per layer (128), the filter length (64) and the nonlinearity between the layers (tanh).

During the training phase of CoNNear, the L1-loss (MAE) was being minimized by adapting the 11.5 million weights of the filter kernels. This L1 loss-term compares the CoNNear outputs with the BM displacements predicted by the TL reference model when presented with the same TIMIT audio sample at the input. 2310 training utterances were used in the weight optimization phase, a phase that roughly took two days to complete 20 epochs on the full training set.

2.2. Training phase

To develop a HI version of CoNNear, two methods were followed: One where training started from scratch using a HI reference TL model and a second one where transfer learning was applied on the NH CoNNear model. The architecture and training framework was developed using a Keras [13] machine learning library with a TensorFlow [14] back-end.

2.2.1. Starting from scratch

In this method, the CoNNear architecture depicted in Fig.1 was initialized using random weights. Training was performed using input-output pairs of the TIMIT speech inputs and the respective outputs from the reference HI TL model. The HI TL outputs reflected how a cochlear gain loss profile associated with a specific audiogram shape affects cochlear filtering [15]. Despite the difference in the reference model, training was performed similarly to the NH TL variant: using 20 epochs and 2310 training utterances. Both *flat35* and *slope35* hearing-loss profiles were considered, which correspond to the most severe expressions of hearing damage in the reference model [1]. The audiograms (cochlear gain loss profiles) of the two implemented HI models are depicted in the insets of the top plots of Fig.2 and include a *flat35* profile with a constant 35 dB gain loss across all frequencies. The sloping profile *slope35* shows frequency-dependent gain loss from 1 kHz to 8 kHz, where it reaches its concluding value of -35 dB gain loss.

2.2.2. Transfer learning

Transfer learning [16] on the other hand, is a machine-learning technique where a model, trained on one task, is reused as a starting point to train a model on a second, related, task. It was assumed here that, although including frequency-dependent gain loss in the cochlear stage, many specific auditory features are similar for both NH and HI profiles and hence might not have to be relearned in the training phase of the HI CoNNear model.

Here, the NH CoNNear model served as a starting point to construct the HI variant: the structure, weights and parameters of the fully trained, NH model were used to develop the HI CoNNear models for the two considered hearing loss profiles. The use of transfer learning saves time on the feature extraction of the speech corpus since, compared to the HI scratch model training, a reduced number of training utterances is needed from the slow to compute HI TL model (50 vs. 2310). Also the train-

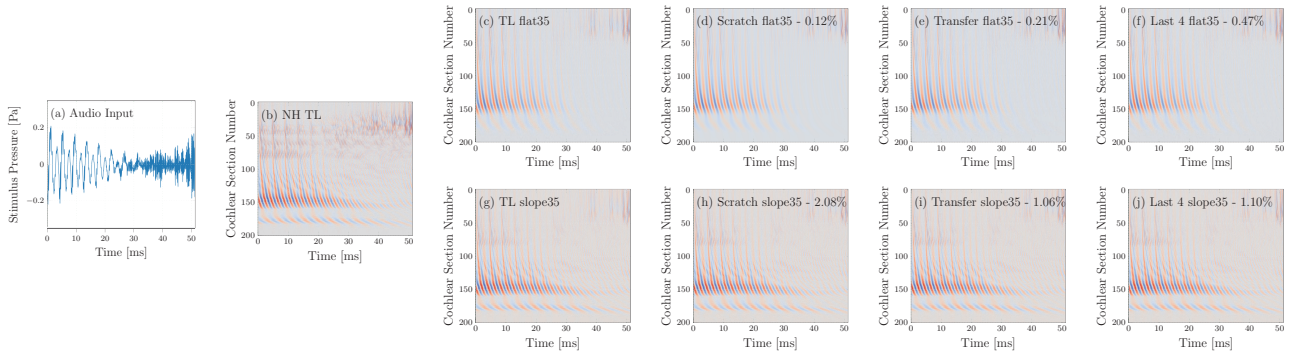

Figure 3: ***HI CoNNear performance evaluated on an unseen speech fragment.*** *Panel (a) shows the stimulus pressure of an audio input file of the TIMIT testset, which was unseen during training. As a benchmark, this stimulus is first passed through the NH TL model [1] (b). Panels (c)-(f) depict the outputs of a flat35 hearing loss profile and display the performance of the reference HI TL model (c) and three HI CoNNear models that were trained using different methods. Panels (g)-(j) show results for a slope35 profile. Panels (b)-(j) show the instantaneous intensities of the considered cochlear filters between 100 Hz (cochlear section 200) and 12 kHz (cochlear section 0). For the trained HI CoNNear models, the mean squared error (MSE) was calculated across all predicted HI CoNNear samples with respect to the reference HI TL outputs for 56 unseen speech fragments of 102.4 ms. The average MSE, normalized by the squared maximum value of each respective reference HI TL model, is included in the HI CoNNear panels.*

ing time itself will be reduced. As part of our evaluation, we will test whether transfer learning can yield the same performance as the training-from-scratch approach.

2.3. Evaluation

Two cochlear mechanics evaluation metrics [9] will be used to evaluate the HI CoNNear models, as well as an additional metric which considers its speech processing power: *(i)* the resulting equivalent rectangular bandwidth or the Q_{ERB}, which quantifies the sharpness of cochlear tuning [2] as a function of stimulus sound pressure level (SPL). *(ii)* The RMS of each of the 201 cochlear filter channel's outputs in response to basic pure tone sound stimuli. These RMS values are plotted according to the corresponding center frequency of the cochlear filter to yield the excitation pattern. Doing this for multiple sound levels can visualize the level-dependent properties of the simulated cochlear excitation patterns. *(iii)* A speech fragment (unseen during training) will be presented to the different models and their waveform outputs compared.

3. Results

The top panel of Fig.2 depicts simulated Q_{ERB}-curves, using procedures described in [9], for the *flat35* and *slope35* hearing loss profiles. Both training methods (scratch and transfer learning) yielded Q_{ERB}-curves which matched those of the reference HI TL models, indicating that both training methods correctly capture the HI cochlear filter tuning characteristics. The performance of both methods is also displayed in Fig.3, where the instantaneous cochlear filter outputs (y_{bm}) across different trained HI CoNNear models are shown. Also for this task, training procedures show similar outcomes with MSE percentages within 2% of the squared maximum reference values.

Table 1 shows the number of trainable model weights for the trained HI CoNNear models, the elapsed time per epoch and the final returned loss term. Comparing the scratch and transfer learning implementations, equally low loss terms are achieved after 20 epochs. However, the transfer learning approach was 300 times faster and reduced the training phase from 2 days to 9 minutes.

Table 1: ***Comparison of trained HI CoNNear.*** *Characteristics showing the number of trainable model weights, time per epoch and the L1 loss term obtained after completing 20 epochs for every trained variant of the HI CoNNear model. Training was performed on a NVIDIA GTX1080 GPU.*

HI CoNNear	Weights	Time/epoch	Final L1 loss
Scratch	11,689,984	7627 s	2.7019e-4
Transfer Learning	11,689,984	26 s	4,4592e-4
First 4 layers	3,154,920	19 s	0,0035
Last 4 layers	8,536,064	23 s	6,9481e-4
Last layer	3,294,184	17 s	0,0016

3.1. Fixed layers

So far, when transfer learning was applied, we updated the weights of all network layers. However, we wanted to know whether the expression of a HI profile in the CoNNear model was situated in specific hidden layers. If so, the total training time could even further be reduced since only a low number of weights would need to be updated during the training phase.

Table 1 shows results for three additional HI CoNNear models where only a part of the layers was made trainable. We conclude that a network which considers only the last 4 layers as trainable will yield a decrease of roughly 3 million trainable parameters while yielding a lower training time per epoch and a comparable loss term with similar performance (Fig.2 and 3). A further decrease in trainable layers yielded an even-faster training time, but was accompanied by a drop in performance (Fig.2) and hence not further considered.

3.2. Timing

Although the performance of the HI CoNNear networks was comparable to the reference HI TL models, validation is necessary to show that these models are indeed operating in a real-time manner and are significantly speeding up computation. To this end, Table 2 shows the execution time of the reference HI

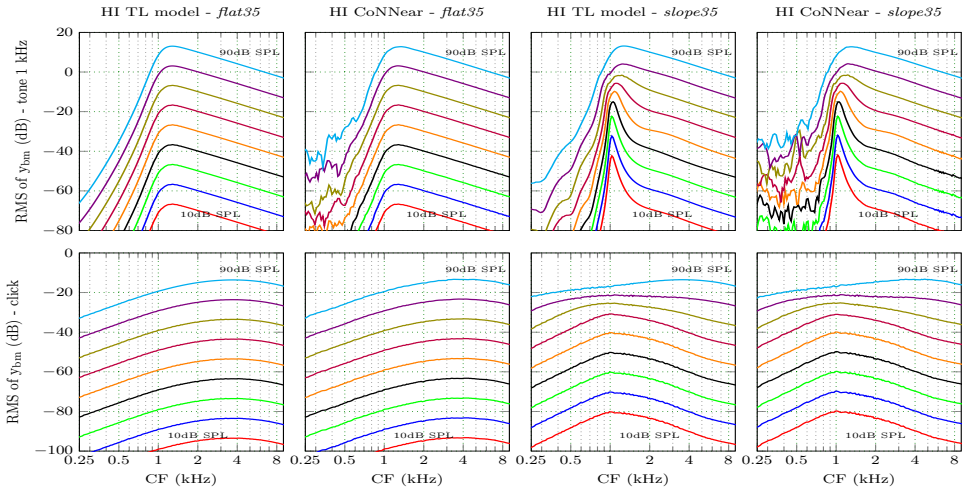

Figure 4: ***Comparison of cochlear excitation patterns across model implementations - HI flat35 and slope35 profiles.*** *Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear filter output for a stimulation with a 1kHz pure tone (top row) and click stimulus (bottom row) of intensities between 10 and 90 dB SPL. Both the best-performing HI CoNNear model for a flat35 hearing loss profile (column 2) and a slope35 hearing loss profile (column 4) are shown. The reference HI TL model simulations for each condition are shown in column 1 and 3. The displayed HI CoNNear models were trained by transfer learning the last 4 layers of the 8-layer NH CoNNear architecture using 50 additional (HI) training utterances.*

TL model and compares it to the trained HI CoNNear networks when presented with the same 1.4 s speech fragment. The input was given to the CoNNear models in 2048 sample length windows (20 kHz sampling frequency). The first sample window was not taken into account due to the initialization time of CoNNear (<1 s). On average, the CoNNear models were 100 times faster than the TL model on a CPU and 2240 times faster on a GPU. Consequently we can conclude that real-time (<10 ms) latencies were obtained in the GPU computation.

Table 2: ***Execution time of trained models.*** *Execution time of trained models of the reference HI TL model compared to trained versions of the HI CoNNear models. Computations were performed on a CPU (Apple MacBook Air, 1.8 GHz Dual-Core processor) as well as on a GPU (NVIDIA GTX1080). The reported time corresponds to the time it took to compute an input fragment of 102.4 ms, averaged over the, all but first, 2048 sample length windows of a 1.4 seconds speech fragment. This fragment was not seen during the training phase.*

	flat35		***slope35***	
	CPU	**GPU**	**CPU**	**GPU**
TL model	25.156 s	NA	21.016 s	NA
Scratch	0.231 s	0.0079 s	0.233 s	0.0077 s
Transfer Learning	0.242 s	0.0079 s	0.234 s	0.0079 s

4. Discussion

Since both the Q_{ERB}-behaviour and the speech performance were comparable between the scratch and the transfer learning HI CoNNear models, we conclude that further training a NH CoNNear model towards a HI model using transfer learning is desired when incorporating a specific hearing-impaired profile in the CoNNear architecture. Our study shows that with 50 additional training examples and 9 minutes of training time (on a GPU), the desired HI CoNNear can be obtained. If needed, the time of the training phase, as well as the number of trainable weights, can further be reduced (without losing significant performance) when only updating the last 4 layers of the NH CoNNear during training. To prove the latter, Fig.4 depicts the excitation patterns of the HI TL reference model next to a HI CoNNear obtained by only updating the weights of the last 4 layers.

5. Conclusions

In this paper a hearing-impaired version of the normal-hearing CoNNear model [9] was formed as a real-time replacement of the biophysically correct, but slow to compute, transmission-line cochlear model of [1]. This HI variant showed to correctly grasp the main cochlear mechanics for different profiles (*flat35* and *slope35*) of outer hair cell cochlear gain loss in a real-time manner, making this a viable option for real-time auditory applications. Secondly, it can be included within backpropagation networks given the differentiable nature of the CoNNear architecture. Possible future work consists of the extension of the CoNNear model beyond the cochlea: the inclusion of other hearing stages (e.g., auditory nerve, cochlear nuclei and inferior colliculus) in a machine hearing, real-time framework. This could prove to be useful in the task of accounting for other types of hearing-impairment (e.g., synaptopathy). Also, it should be tested to which degree the trained HI CoNNear models can be used to design hearing-loss compensation algorithms for the next-generation of hearing aids.

6. Acknowledgements

Work supported by European Research Council ERC-StG-678120 (RobSpear). Code of the CoNNear models can be found on: *https://github.com/hearingtechnology*.

7. References

[1] S. Verhulst, A. Altoè, and V. Vasilkov, "Computational modeling of the human auditory periphery: Auditory-nerve responses, evoked potentials and hearing loss." *Hearing research*, 360:55-75, 2018.

[2] C. A. Shera, J. J. Guinan, and A. J. Oxenham, "Otoacoustic estimation of cochlear tuning: Validation in the Chinchilla." *J. Assoc. Res. Otolaryngol*, 11, 343–365, 2010.

[3] A. Saremi, R. Beutelmann, M. Dietz, G. Ashida, J. Kretzberg, and S. Verhulst, "A comparative study of seven human cochlear filter models." *The Journal of the Acoustical Society of America*, 140(3):1618–1634, 2016.

[4] A. Aertsen, P. I. Johannesma, and D. Hermes, "Spectro-temporal receptive fields of auditory neurons in the grassfrog." *Biological Cybernetics*, 38(4):235–248, 1980.

[5] E. De Boer, "Synthetic whole-nerve action potentials for the cat." *The Journal of the Acoustical Society of America*, 58(5):1030–1045, 1975.

[6] R. Meddis, L. P. O'Mard, and E. A. Lopez-Poveda, "A computational algorithm for computing nonlinear auditory frequency selectivity." *The Journal of the Acoustical Society of America*, 109(6), 2852-2861, 2001.

[7] R. F. Lyon, "Cascades of two-pole–two-zero asymmetric resonators are good models of peripheral auditory function." *J. Acoust. Soc. Am.* 130(6), 3893–3904, 2001.

[8] D. Baby, and S. Verhulst, "Biophysically-inspired features improve the generalizability of neural network-based speech enhancement systems." In *19th Annual Conference of the International-Speech-Communication-Association (INTERSPEECH 2018)*, (pp. 3264-3268), ISCA, 2018.

[9] D. Baby, A. Van Den Broucke, and S. Verhulst, "A convolutional neural-network model of human cochlear mechanics and filter tuning for real-time applications." *arXiv preprint* arXiv:2004.14832, 2020.

[10] J. S. Garofolo, "Timit acoustic phonetic continuous speech corpus." *Linguistic Data Consortium*, 1993.

[11] D. D. Greenwood, "Critical bandwidth and the frequency coordinates of the basilar membrane." *The Journal of the Acoustical Society of America*, 33(10):1344–1356, 1961.

[12] C. Lorenzi, G. Gilbert, H. Carn, S. Garnier, and B. C. Moore, "Speech perception problems of the hearing impaired reflect inability to use temporal fine structure." *Proceedings of the National Academy of Sciences*, 103(49), 18866-18869, 2006.

[13] F. Chollet, et al., "Keras: The python deep learning library." *Astrophysics Source Code Library*, 2018.

[14] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al., "Tensorflow: A system for large-scale machine learning." In *12th USENIX Symposium on OSDI 16*, pages 265–283, 2016.

[15] S. Verhulst, A. Jagadeesh, M. Mauermann, and F. Ernst, "Individual differences in auditory brainstem response wave characteristics: Relations to different aspects of peripheral hearing loss." *Trends in hearing*, 20, 2331216516672186, 2016.

[16] S. J. Pan, and Q. Yang, "A survey on transfer learning." *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[17] P. Kummer, T. Janssen, P. Hulin, and W. Arnold, "Optimal L1-L2 primary tone level separation remains independent of test frequency in humans." *Hearing research*, 146(1-2), 47-56, 2000.