FACULTY OF ENGINEERING
AND ARCHITECTURE

Towards Individualized Hearing Profiles Using Deep Neural Nets

Arthur Van Den Broucke
Student number: 01301281

Supervisors: Prof. dr. Sarah Verhulst, Dr. ir. Deepak Baby (Universiteit Gent)

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Biomedical Engineering

Academic year 2018-2019

GHENT
UNIVERSITY

Confidential up to and including 12/31/2019

Important

FACULTY OF ENGINEERING
AND ARCHITECTURE

Towards Individualized Hearing Profiles Using Deep Neural Nets

Arthur Van Den Broucke
Student number: 01301281

Supervisors: Prof. dr. Sarah Verhulst, Dr. ir. Deepak Baby (Universiteit Gent)

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Biomedical Engineering

Academic year 2018-2019

GHENT
UNIVERSITY

Acknowledgements

> *Blindness separates people from things; deafness separates people from people.*
>
> Helen Keller

This dissertation marks the end of my master's in Biomedical Engineering, a feat that wouldn't be possible without the help of many others.

First of all, I would like to thank my supervisors, Prof. dr. Sarah Verhulst and Dr. ir. Deepak Baby, for the opportunity to write this thesis. Prof. Verhulst for the weekly feedback and the smooth incorporation in the Hearing Technology Lab team. Dr. Baby for his daily guidance and help throughout the project, ranging from preliminary work to assisting with the layout of the displayed graphs.

I want to show my gratitude towards my parents for all the years of unconditional love, help and support, I will never take any of this for granted. To my brothers, Jules and Louis, who both serve as an inspiration for me in several aspects of life. The role of my other family members and closest friends should also not be overlooked throughout this journey. Thank you all.

A final word for my grandfather who is, and always will be, serving as an example for me. I am thankful that you can play such a significant role in my life. Opa, bedankt.

Arthur Van Den Broucke, June 2019

Permission of Use on Loan

The author gives permission to make this master dissertation available for consultation and to copy parts of this master's dissertation for personal use. In all cases of other use, the copyright terms have to be respected, in particular with regard to the obligation to state explicitly the source when quoting results from this master's dissertation.

Arthur Van Den Broucke, June 2019

Towards Individualized Hearing Profiles Using Deep Neural Nets

by
ARTHUR VAN DEN BROUCKE

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Biomedical Engineering

Academic year 2018 - 2019

Supervisors: Prof. dr. SARAH VERHULST, Dr. ir. DEEPAK BABY

Department of Information Technology
Chair: Prof. dr. BART DHOEDT
Faculty of Engineering and Architecture
Ghent University

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 (in the 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 (e.g., gammatone, CARFAC and MFCC models) even though there are clear advantages in using more biophysically correct models (e.g., phase). To overcome this, this dissertation combines nonlinear Deep Neural Nets (DNN) with nonlinear TL cochlear models to build a real-time model of the cochlea, able to capture the biophysical properties associated with the TL model. The DNN model was trained using a speech dataset at a fixed sound level, but performed well on a set of basic auditory stimuli of various stimulus levels and frequencies to assess the coupling, tuning and nonlinearity of the new model. The normal-hearing DNN model was afterwards adjusted, by means of transfer learning, to simulate frequency-specific patterns of cochlear gain loss profiles, yielding a set of normal-hearing and hearing-impaired DNN models which can be computed in real-time, are differentiable, and can serve as the next generation of hearing-aid and machine hearing applications.

Keywords

Cochlear models, real-time applications, deep neural networks, hearing-impairment, transfer learning

Towards Individualized Hearing Profiles Using Deep Neural Nets

Arthur Van Den Broucke
Supervisors: Prof. dr. Sarah Verhulst, Dr. ir. Deepak Baby

***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 (in the 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 (e.g., gammatone, CARFAC and MFCC models) even though there are clear advantages in using more biophysically correct models (e.g., phase). To overcome this, this dissertation combines nonlinear Deep Neural Nets (DNN) with nonlinear TL cochlear models to build a real-time model of the cochlea, able to capture the biophysical properties associated with the TL model. The DNN model was trained using a speech dataset at a fixed sound level, but performed well on a set of basic auditory stimuli of various stimulus levels and frequencies to assess the coupling, tuning and nonlinearity of the new model. The normal-hearing DNN model was afterwards adjusted, by means of transfer learning, to simulate frequency-specific patterns of cochlear gain loss profiles, yielding a set of normal-hearing and hearing-impaired DNN models which can be computed in real-time, are differentiable, and can serve as the next generation of hearing-aid and machine hearing applications.**

***Index Terms*—Cochlear models, real-time applications, deep neural networks, hearing-impairment, transfer learning**

I. Introduction

Generally accepted as one of the most complex pathways in the human body, hearing can be seen as a deep and elegant combination of linear and nonlinear aspects. This complexity can be largely attributed to the inner ear's cochlea where frequency selectivity across characteristic frequencies (CF), longitudinal coupling and level-dependent compression are giving rise to a highly nonlinear behaviour. Posing a difficult task to approach the hearing organ by numerical model representations.

These models are being classified in one of two categories: perceptual, functional models or biophysical models. Perceptual models (e.g., gammatone [1] [2], MFCC [3]) reproduce the overall input-output relation of the auditory system while disregarding the underlying biophysical subprocesses [4]. Biophysical models (e.g., transmission-line models [5]), on the other hand, are more focussed on implementing the correct biological processes that can be found in the cochlea.

Literature shows that, although biophysical models are grasping better the full range of above mentioned hearing characteristics, it is the collection of linear perceptual models that is being deployed in various hearing applications (e.g., ASR, noise suppression). This can be related to the fact that they, as opposed to the nonlinear biophysical models, lack computational complexity. It should however be clear that this, ever-present, compromise of computational speed and biophysical correctness is far from ideal.

This paper aims to solve this presented research gap by providing a real-time variant of a biophysically correct, nonlinear cochlear model. This instantaneous character is achieved by applying deep neural network (DNN) techniques to train a fast operating convolutional neural network (CNN) able to account for normal-hearing (NH) cochlear behaviour. Since present models have the ability to include hearing-impaired (HI) profiles as well, another goal is pursued in this paper: an extension of the first (NH) DNN towards a structure that is able to approximate HI cochlear processing as well. The used method for this latter task will be based on transfer learning.

II. Cochlear Mechanics

Inside the field of cochlear modeling it is known that the human cochlea accounts for two essential nonlinearities [6]:

1) Compression at high sound-level: Whereas the cochlear peak response shows linear growth with level for low-to-moderate sound levels, the response grows compressively for higher sound intensities [7].

2) Sharper cochlear tuning for softer sounds: : Basilar membrane (BM) filters have a sharper shape for softer sounds. [8].

The combination of these aspects of *cochlear nonlinearity (i)* together with a correct expression of the *frequency-selective cochlear tuning (ii)* and the ability to capture the natural *longitudinal coupling of the BM filters (iii)*, can be seen as valid criteria for a biophysically correct cochlear model. The combination of these criteria can only be found in the advanced transmission-line (TL) model of Verhulst et al. [9], hence it is this model that will be adopted as reference model in this research. The model uses a cascaded TL to model the cochlear mechanics and travelling waves [5], [10] to correctly account for the above mentioned features [11]. Since this model also holds the possibility to render a HI version of the cochlear processing (by adapting the model parameters responsible for cochlear gain), it is convenient to use it since the second task can be performed on it as well.

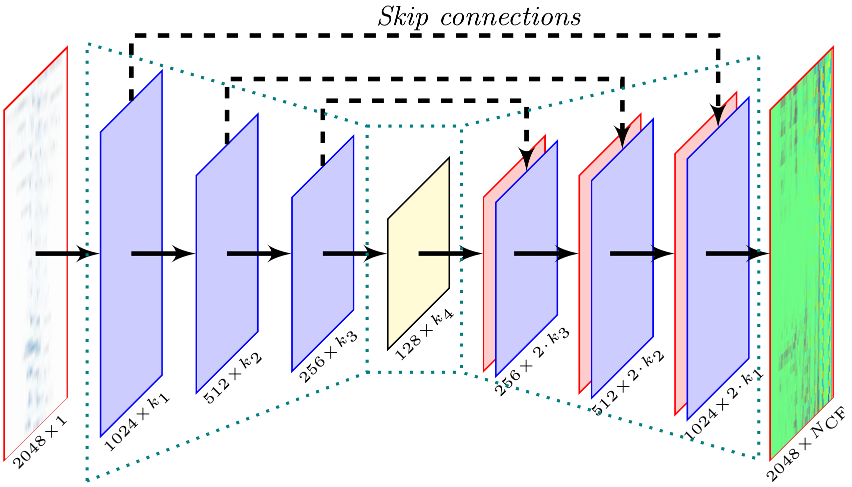


Fig. 1. **The AECNN architecture used in this research (An 8 layer architecture is depicted here as an example):** The architecture makes use of an encoder-decoder [13], [14] structure, where an audio input of sample length 2048 is first processed by an encoder (comprised of a few CNN layers), which encodes the audio signal into a condensed representation. These CNN layers, which were chosen over other neural net approaches since it's most related to cochlear filtering, make use of strided convolutions, meaning the filters are shifted by a time-step of 2, reducing the temporal dimension by half after every CNN layer. Thus, after N CNN layers in the encoder, the audio signal is encoded into a condensed representation which is of size 2048/2^N x k_N , where k_N is the number of filters in the N^{th} CNN layer. This encoded representation is then mapped to the corresponding BM displacements using a decoder which uses deconvolutional or transposed-convolutional layers where the temporal dimension is doubled after every layer. The decoder also contains N deconvolution layers, yielding back the starting temporal dimension of 2048 samples. The number of filters used in the final CNN layer of the decoder is set to be equal to the number of cochlear sections N_{CF} that were available in the reference TL model's output (set to 201 in this research to reduce training time, resembling a frequency range of 100 Hz - 12 kHz [12]). Thus the output of the proposed AECNN model is of size 2048 x N_{CF}. U-shaped skip connections are added as well, since, due to the use of strided convolutions in the CNN layers, the encoder might lose some important information such as temporal alignment and phase information. These connections will hence bypass the relevant information from the encoder to the decoder.

III. Materials and Methods

The cascaded approach of the TL represents the cochlea as a coupled structure, where the response of one cochlear section (CS) is depending on the responses of all previous sections. Since the characterization of cochlear mechanics and travelling waves will be based on the BM displacements of these CS (linked to a certain characteristic frequency (CF) to correctly follow the human tonotopy map [12]), modeled by ordinary differential equations in the TL model, these (displacements across CS) will serve as benchmark values to correctly model the cochlear processing. Hence the evaluation of the performance of the trained DNN will be largely based on how the neural network is predicting these BM displacements. The assumption was made that suppose the DNN would be able to output BM movements similar to the ones predicted by the TL model, when given the same audio input, it would also be able to grasp the underlying cochlear mechanics.

This resembling ability is achieved by deep learning: A DNN can be seen as a combination of (hidden) layers which on their part are characterized by a large number of filter weights, responsible for recognizing patterns and structures in the input data. During a learning phase (further referred to as training), these weights are updated in a way that they are minimizing a certain loss term (based on the difference of the desired and the predicted output for a given input). This training is done by 'showing' a large number of input-output combinations to the network, so the NN can learn what the relation is between the input and the output data. In this particular case: how does the BM displacements (output) look for a certain audio input.

Fig. 1. depicts an example of the DNN architecture that will be used in this project to accomplish this task. How the training of this architecture is done is described in the next section.

A. Preprocessing Pathway

1) Data collection: First, audio (2310 spoken sentences) was collected from the TIMIT dataset [15], which were adjusted such that the RMS energy of the signal had a sound pressure level of 70 dB (resembles best standard conversational speech levels and includes both louder and more silent instantaneous amplitudes).

2) Resampling: The TIMIT dataset has a sampling frequency of 16 kHz, whereas the reference model demands an input sampling frequency of 100 kHz, hence an upsampling was performed on the TIMIT data. Since the TL model will, on the other hand, output a signal with a sampling frequency of 20 kHz, the same TIMIT data needed to be upsampled to 20 kHz as well, this to have the same sampling frequency in the input/output pairs used in the training phase of the AECNN.

3) The reference TL model: Subsequently, the upsampled data was given as an input to the Verhulst et al. model [9].This model predicted the BM displacements across the 201 CF for each speech fragment.

4) Slicing of data: Since the proposed AECNN architecture was set up to only process input data with a sample length of 2048 (see Fig. 1.), both the TIMIT data, as well as the output of the TL model, were sliced in chunks of 2048 samples (102.4 ms) and stored. It's the combination of the TIMIT fragment and accompanying TL output that will form the utterances that are being fed to the network during training.

B. Machine Learning Parameters

Before training is initiated, some fixed parameters and hyperparameters needed to be chosen. The fixed parameters stayed the same for the entire research and consisted of the amount of filters per layer k_N (128), the batch size (32 samples), the number of epochs (20), the optimizer (Adam [16]) and the type of loss function (L1 loss, mean absolute error). We refer to dedicated literature for a detailed explanation of these parameters [17]. The hyperparameters, having the most overall effect on the performance of the trained models, were the variable settings during this research. These will be discussed in the next section. After the completion of this initialization process, training was started and after 20 epochs, the NH DNN version of the TL model was formed.

This entire architecture and training framework was developed using a Keras [18] machine learning library with a TensorFlow [19] back-end.

IV. Results - NH AECNN

As stated above, the variation of hyperparameter values was investigated in this research, in search of the best performing neural network architecture. These parameters (listed in Table I) are all, in a direct or indirect manner, affecting the time and memory cost of running the DNN training phase, hence making it crucial design parameters.

TABLE I
Investigated hyperparameters (parameters which provided the best performance are in **bold text**).

Hyperparameter	Investigated values
Learning rate	0.001 - 0.0004 - **0.0001**
Layer depth	4 - **6** - 8
Filter length	31 - 63 - **127**
Nonlinear activation function	PReLU - **tanh**

Time and memory cost however, which are general attributes of each trained DNN, don't necessarily reveal how well the model is performing at the required task, hence three additional performance measurements (discussed in the following three sections) were done in the evaluation process. This returned that the *6 layers, 127 filter length, 0.0001 learning rate, tanh model* gave the best overall performance measurements. This model, with 16,955,008 trainable parameters, was trained on 2310 TIMIT training utterances, which gave a total training time of roughly 40 hours (for 20 epochs). After training, a L1 loss term of 0.0148 was achieved (for comparison: the worst performing model returned a 0.0404 loss term).

A. Performance on Basic Auditory Input Stimuli

An audio (or speech) fragment can be seen as a combination of basic components such as click impulses and pure tones varying in frequency. Since the DNN is not trained (the TIMIT corpus only contains speech samples) on those types of basic stimuli, commonly used in cochlear mechanics studies, they are a good performance measure to evaluate how the trained models are performing.

The left column of Fig. 2. depicts the three basic stimuli (input level of 70dBSPL) which will be fed to the trained architecture: a 100 μs click and pure tone stimuli of 1 kHz and 4 kHz. The cochlear dispersion based on the reference TL model is shown in the second column, followed by the predicted response by our trained AECNN. The final plot in every row reveals the difference between the two former ones. The depicted model is the previous mentioned 6 layers, 127 filter length, 0.0001 learning rate, tanh model. For numerical evaluation, the mean square error of the 411,648 values (201 cochlear sections x 2048 samples) was calculated. This relative measure allowed to compare the different models in a numerical manner, based on their predictive performance on the above stated stimuli. Here again, the depicted model performed among the best, as can be seen on the figure: doing an excellent job at predicting the BM displacements for (unseen) basic input stimuli.

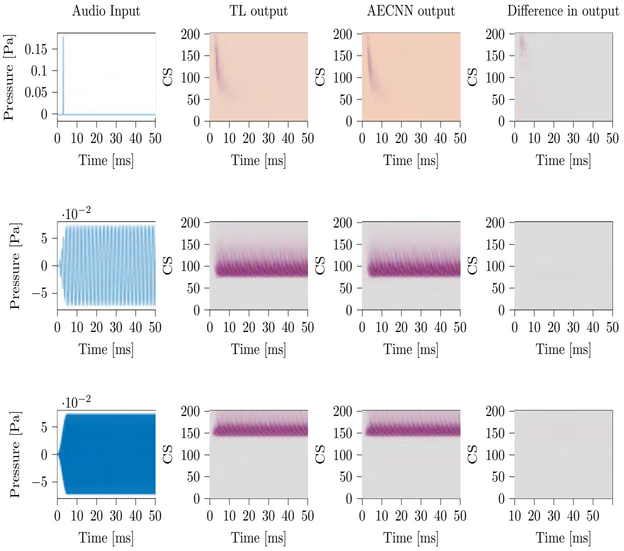


Fig. 2. **Performance best AECNN model on basic auditory input stimuli.** (Left column) Input pressure [Pa] in the time domain [ms] for the three different stimuli (click, pure tone 1 kHz and pure tone 4 kHz). (Middle columns) Output cochlear dispersion of the TL and trained NH AECNN model: BM displacements for the selected 201 CS, for their respective input signal. (Right column) Difference between the two previous depicted outputs.

Starting from these previous plots, the RMS value (in dB), for each of the 201 CS outputs, was calculated. Once all of these RMS values are plotted according to their corresponding CF on a frequency axis, so called excitation patterns are formed. Doing this for multiple sound levels (ranging from 10 dBSPL to 90 dBSPL) will allow to visualize the degree of level-dependency in between those excitation patterns. This level-dependency should follow a nonlinear behaviour, due to cochlear compression, across the sound levels.

Fig. 3. depicts these excitation patterns for both the best performing model with a PReLU as nonlinear activation function and the best performing overall model (tanh activation function). As can be seen, only the tanh-inspired model is able to capture the desired level-dependency. This proves the computational power of DNN structures: the AECNN was only trained on 70 dBSPL sound fragments and hence it would be expected that only 70 dBSPL sounds could be predicted correctly (which was only the case for the PReLU model).

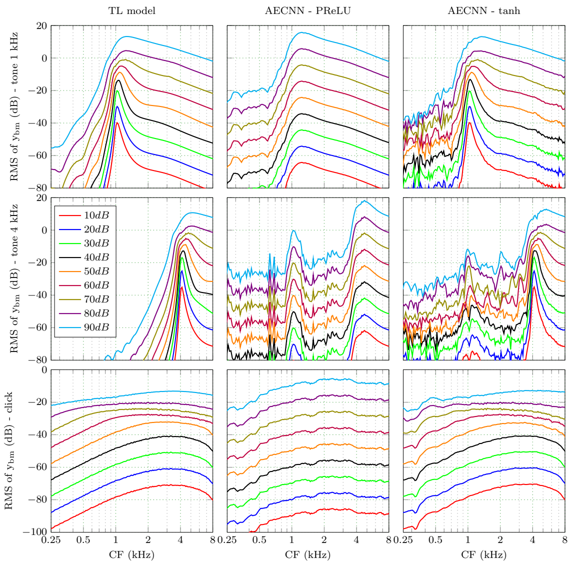


Fig. 3. **Comparison of excitation patterns - Variable nonlinear activation function.** Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model (left) and AECNN architectures varying in nonlinear activation function (PReLU - tanh). It shows that only the tanh model is able to correctly capture the level-dependency characteristic.

B. Performance on Test Set

Since the trained DNN should also be able to correctly predict cochlear outputs for speech fragments that were not part of the training dataset, the models were not only tested on basic input stimuli. A test set, containing 64 unseen speech fragments, was selected from the TIMIT corpus. Thereafter a segment of 2048 samples was chosen from each of the 64 fragments and was fed to both the reference TL model and the trained neural network architecture. Here again, the MSE of all 411,648 values was calculated and used in addressing the overall performance. This again favoured the 6 layers, 127 filter length, 0.0001 learning rate, tanh model.

C. Q_{ERB}

The final performance measurement was the resulting equivalent rectangular bandwidth or the Q_{ERB}. This can be used as a quantification of the sharpness of cochlear tuning [20] as a function of level, one of the cochlear attributes that was demanded to be included in the trained DNN. This Q_{ERB} value as a function of frequency follows a typical curve for humans [20]. This value is described as:

$$Q_{ERB} = \frac{CF}{ERB} \tag{1}$$

Where CF is again the characteristic frequency coupled to a certain cochlear section and ERB the, CF-dependent, equivalent rectangular bandwidth: the bandwidth of a rectangular filter with the same peak response that passes the same total power of a power spectrum that is driven by the same stimulus. This power spectrum is calculated from the fast Fourier transform of the stimulus' impulse response at a specific CF. For the evaluation here, a 100 μs click stimulus [21], [22] was used. Q_{ERB} values for both the PReLU and tanh variant of the best performing model are plotted in Fig. 4. and Fig. 5.

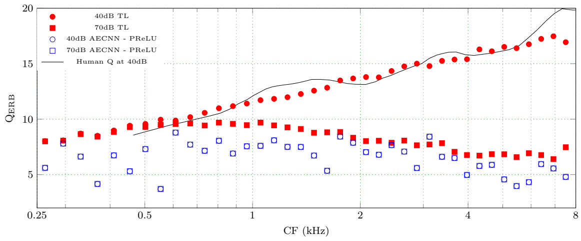


Fig. 4. Q_{ERB} **values for a trained PReLU AECNN.** Q_{ERB} values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the TL model (red), trained PReLU AECNN model (blue) and a literature human Q_{ERB} estimate [20]. The AECNN **is not able** to account for the level-dependency present in cochlear tuning.

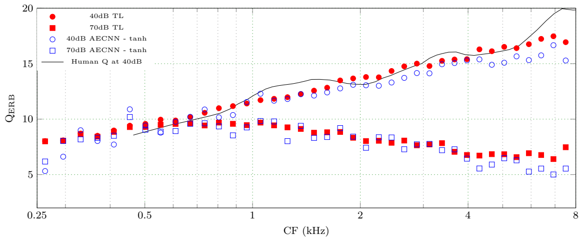


Fig. 5. Q_{ERB} **values for a trained tanh AECNN.** Q_{ERB} values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the TL model (red), trained tanh AECNN model (blue) and a literature human Q_{ERB} estimate [20]. The AECNN **is able** to account for the level-dependency present in cochlear tuning.

D. Context

Although the performance of the AECNN on the Q_{ERB} values is very good for the tanh-inspired model, Fig. 5. also reveals a slightly lower performance for the lower frequency range. A possible explanation can be found in the applied processing pathway, where slicing of both the TIMIT dataset and the TL reference model output to segments of 2048 samples was done. However, to receive the reference TL model output, the full length of a training example was presented at the input, thus including the context (the samples that are proceeding and succeeding) of each sample. This context however is partially lost for the first samples when sliced. This means that the AECNN was trained on examples (the reference model output) that contained information linked to the proceeding context of the cropped audio sample, information that the AECNN cannot see. It is possible that this slightly poorer resemblance for the lower frequency values can be explained by this. This incorporation of context for the data used in the AECNN training phase can be seen as an extension of this research.

V. Results - HI AECNN

Using the best performing NH AECNN model as a starting point to train a HI version of the AECNN, capable of representing hearing loss profiles in its output was the next step in this research.

Training was done, based on input-output combinations of the same TIMIT dataset, but now passed trough the reference TL model that was made hearing-impaired by CF-dependent adjusting of the parameters responsible for simulating cochlear gain. This produced wider cochlear filters, associated with outer hair cell damage [9].

The hearing impaired profiles that were addressed in this research included a 'slope' hearing loss profile, inducing a sloping gain loss starting at a CF of 1 kHz, and a 'flat' hearing loss profile, that has a constant gain loss over the entire frequency spectrum. The most severe 35 dB variant of both profiles was selected [9].

The used training method was transfer learning [23], the 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. Transfer learning assumes that the learned features of the first task, are general and hence transferable to the second task. This approach significantly decreased the number of training utterances since, whereas the NH training needed 2310 training utterances, only 50 additional utterances were used in transfer learning since hearing aspects, that are not altered by cochlear gain loss, were already present in the trained NH AECNN, hence didn't need to be learned again. The HI AECNN variants (slope 35/flat 35) of the NH, 6 layers, 127 filter length, 0.0001 learning rate, tanh-model were trained in only 10 minutes and had an average loss term of 0.0043.

A. Performance on Basic Input Stimuli

The excitation patterns for the HI AECNN versions are depicted on Fig. 6., proving that the trained HI models were able to correctly capture the nonlinear level-dependency present in the HI TL model's excitation patterns.

B. Q_{ERB}

The corresponding Q_{ERB} plots for both the slope 35 and the flat 35 hearing loss profile are depicted in Fig. 7. and Fig. 8. Disregarding the suboptimal low frequency results (see Subsection IV-D), it can be stated that the HI tuning characteristic is correctly grasped by the AECNN.

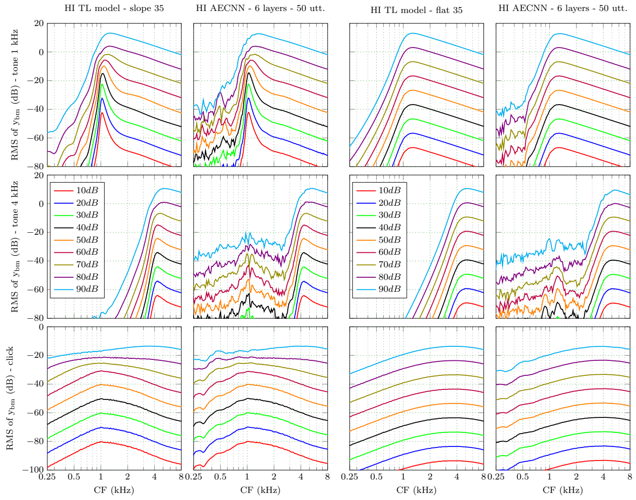


Fig. 6. **Comparison of excitation patterns - HI slope 35 and flat 35 hearing loss profiles.** Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. This for both the best performing model for a slope 35 hearing loss profile (left) and a flat 35 hearing loss profile (right). Each case depicts the reference TL model on the left, and the HI AECNN architecture output, for the considered hearing loss profile, on the right. Both AECNN models were trained via transfer learning on the 6 layer-tanh architecture for an additional 50 (HI) training utterances.

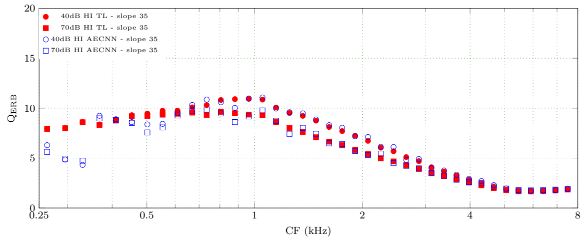


Fig. 7. **Q_{ERB} values for trained HI AECNN - slope 35.** Q_{ERB} values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the HI TL model (red) and trained HI AECNN model (blue) for a slope 35 HL profile. Refer to Subsection IV-D for an explanation of the suboptimal low frequency performance.

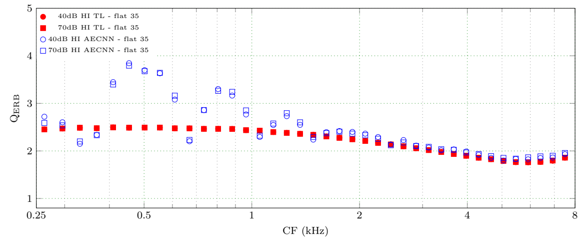


Fig. 8. **Q_{ERB} values for trained HI AECNN - flat 35.** Q_{ERB} values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the HI TL model (red) and trained HI AECNN model (blue) for a flat 35 HL profile. Refer to Subsection IV-D for an explanation of the suboptimal low frequency performance.

C. Fixed layers

The question was asked whether the incorporation of HI profiles in the AECNN was situated only in some hidden layers. This is based on DNN used in image recognition, where certain layers are responsible for the detection of specific structures. This could reduce the total training time even more since only a certain number of parameters will be updated during training.

To verify this, the performance of the 6 layers - 50 utterances - tanh model for a flat 35 HL profile was compared to 4 other models. In the training phase of each of those models only a part of the hidden layers was made trainable. The results showed that the model of which only the last layer's weights were made trainable, had the same resembling performance as the reference model where all 6 layers were trained. The outcome of this fixation of layers was a reduction of 10,420,096 trainable parameters and a time gain of 12 seconds per epoch, returning only a 7 minute training phase. The same result was obtained for a slope 35 HL profile, proving that the accountability for HI in this AECNN architecture can indeed be situated in the last hidden layer.

VI. Conclusion

In this paper, a deep neural network (DNN) architecture was presented to approximate a state-of-the-art, biophysically realistic model of the human cochlea, based on a cascaded nonlinear transmission-line (TL) model. This to remove the ever-existing compromise between biophysically correctness and computational complexity. The reference TL model, on which the DNN architecture was based, possessed also the ability to include hearing-impaired (HI) profiles, based on outer hair cell cochlear gain loss, in its modelling stages. Hence the second objective of this paper consisted of correctly including the auditory processing of a HI cochlea in the DNN as well. This was done by applying transfer learning, where the first NH DNN was functioning as a starting point.

Results showed that, with the correct hyperparameter choices, the desired nonlinear features of the cochlea: longitudinal coupling, frequency-selective tuning and level-dependent compression, could all be found in the performance of the real-time operating DNN. Whereas the process of transfer learning (after the freezing of the filter weights of the hidden layers that showed to interfere with cochlear gain loss) permitted to achieve a HI version of the starting DNN within 7 minutes and trained only on 50 additional HI utterances.

This approach proved its value and the DNN framework can be considered for the replacement of any transmission-line model that incorporates nonlinearities (e.g., brain networks, electronics applications), but also has the ability to be applied into low-power implementations (e.g., ASR, next generation of (smart) hearing-aids, robotics).

VII. Future work

Future work could include: (i) Adding context to the speech fragments in the training phase, to account for the discontinuities in the AECNN output (as mentioned in Section IV-D). (ii) Extending the DNN beyond the cochlear stage, including other hearing stages (e.g., auditory nerve, cochlear nuclei and inferior colliculus) in a machine hearing, real-time framework. This would allow addressing other types of hearing-impairment in auditory modeling (e.g., synaptopathy).

References

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

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

[3] Davis, S. and Mermelstein, P. (1980). Comparison of parametric representations for monosyl- labic word recognition in continuously spoken sentences. IEEE transactions on acoustics, speech, and signal processing, 28(4):357–366.

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

[5] Zweig, G. (1991). Finding the impedance of the organ of corti. The Journal of the Acoustical Society of America, 89(3):1229–1254.

[6] Eguiluz, V. M., Ospeck, M., Choe, Y., Hudspeth, A., and Magnasco, M. O. (2000). Essential nonlinearities in hearing. Physical review letters, 84(22):5232.

[7] Ni, G., Elliott, S. J., Ayat, M., and Teal, P. D. (2014). Modelling cochlear mechanics. BioMed research international, 2014.

[8] Rosen, S., Baker, R. J., and Darling, A. (1998). Auditory filter nonlinearity at 2 khz in normal hearing listeners. The Journal of the Acoustical Society of America, 103(5):2539–2550.

[9] Verhulst, S., Alto'e, A., and Vasilkov, V. (2018). Computational modeling of the human auditory periphery: Auditory-nerve responses, evoked potentials and hearing loss. Hearing research, 360:55–75.

[10] Altoe, A., Pulkki, V., and Verhulst, S. (2014). Transmission line cochlear models: improved accuracy and efficiency. The Journal of the Acoustical Society of America, 136(4):EL302– EL308.

[11] Verhulst, S., Dau, T., and Shera, C. A. (2012). Nonlinear time-domain cochlear model for transient stimulation and human otoacoustic emission. The Journal of the Acoustical Society of America, 132(6):3842–3848.

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

[13] Bourlard, H. and Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. Biological cybernetics, 59(4-5):291–294.

[14] Hinton, G. E. and Zemel, R. S. (1994). Autoencoders, minimum description length and helmholtz free energy. In Advances in neural information processing systems, pages 3–10.

[15] Garofolo, J. S. (1993). Timit acoustic phonetic continuous speech corpus. Linguistic Data Consortium, 1993.

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

[17] Goodfellow, I., Bengio, Y., and Courville, A. (2016). Deep learning. MIT press.

[18] Chollet, F. et al. (2018). Keras: The python deep learning library. Astrophysics Source Code Library.

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

[20] Shera, C. A., Guinan, J. J., and Oxenham, A. J. (2002). Revised estimates of human cochlear tuning from otoacoustic and behavioral measurements. Proceedings of the Na- tional Academy of Sciences, 99(5):3318–3323.

[21] Verhulst, S., Bharadwaj, H. M., Mehraei, G., Shera, C. A., and Shinn-Cunningham, B. G. (2015). Functional modeling of the human auditory brainstem response to broadband stimulation. The Journal of the Acoustical Society of America, 138(3):1637–1659.

[22] Raufer, S. and Verhulst, S. (2016). Otoacoustic emission estimates of human basilar membrane impulse response duration and cochlear filter tuning. Hearing research, 342:150–160.

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

Contents

List of Figures

List of Tables

List of Abbreviations

ABR	Auditory Brainstem Response
AE	Autoencoder
AI	Artificial Intelligence
AN	Auditory Nerve
ASR	Automatic Speech Recognition
BM	Basilar Membrane
CF	Characteristic Frequency
CNN	Convolutional Neural Network
CS	Cochlear Section
dB	Decibel
DNN	Deep Neural Network
EFR	Envelope Following Response
ERB	Equivalent Rectangular Bandwidth
FL	Filter Length
HI	Hearing-Impairment
HL	Hearing loss
IHC	Inner Hair Cell

LSTM	Long Short-Term Memory
LT	Loss Term
MFCC	Mel-Frequency Cepstral Coefficients
ML	Machine Learning
MSE	Mean Square Error
NH	Normal-Hearing
NL	Nonlinear(ity)
NN	Neural Network
OHC	Outer Hair Cell
PReLU	Parametric Rectified Linear Unit
ReLU	Rectified Linear Unit
RMS	Root Mean Square
SPL	Sound Pressure Level
TL	Transmission-Line

Chapter 1

Introduction

The ear is a most complex and beautiful organ. It's the most perfect acoustic, or hearing instrument, with which we are acquainted, and the ingenuity and skill of man would be in vain exercised to imitate it. (Frost, 1838)

181 years ago, this statement of John Frost made in his book *The Class Book of Nature: Comprising Lessons on the Universe, the Three Kingdoms of Nature, and the Form and Structure of the Human Body*, will probably not have received a lot of backlash. However today, in the year 2019, man are not only bound by their own skills. So to state that it would be in vain to try to imitate it, would be underestimating the power of one specific tool that is at the disposal of man today. Machines.

It was David Marr in 1979, who showed in his ground-breaking paper *A computational theory of human stereo vision* that human vision could be imitated by means of numbers and machines, transforming colors, forms, details and motion into numbers and matrices (Marr and Poggio, 1979). 40 years later, the applications of machine vision are immense. The question should be asked if it's too bold to say that, if vision can be represented, this could also be made possible for human hearing? To transform loudness, pitch and timbre into an expression that can be processed and simulated by a machine?

Since hearing can be seen as the most deep and elegant combination of linear and nonlinear aspects, it is generally accepted one of the most complex pathways in the human body. A level of complexity that translates itself in the difficult task of approaching the hearing organ by numerical model representations. This complexity can be largely attributed to the inner ear's cochlea where frequency selectivity across characteristic frequencies (CF), longitudinal coupling and level-dependent compression are giving rise to a highly nonlinear behaviour.

When looking at models of the human auditory periphery -and more specific cochlear models- a clear distinction can be made between perceptual, functional models (e.g., gammatone, MFCC) and biophysical models (e.g., transmission-line). Perceptual models reproduce the overall input-output relation of the auditory system while disregarding the underlying biophysical subprocesses (Saremi et al., 2016). Biophysical models, on the other hand, are more focussed on implementing the correct biological processes that can be found in the cochlea.

Although literature (see chapter 2, Section 2.3) shows that those biophysical models perform better in grasping the full range of above mentioned hearing characteristics, nowadays mainly the perceptual models are deployed in various machine hearing applications. The linear perceptual models (gammatone) have proven their worth by being less computational heavy than the nonlinear and complex biophysical models. It is clear that this ever-present compromise of computational speed and biophysical correctness is far from ideal.

It is the research gap in this trade-off this master's thesis hopes to bridge: Would it be possible to have the best of both worlds? Keep the lack of computational complexity of the 'basic' linear models, but make the final result as biophysically correct as the best performing nonlinear models.

Again referring to machine vision, this field has over the years largely benefited from fast emerging deep learning methods. Methods that make it possible to approach signal processing problems from a new perspective. Deep neural networks (DNN) already have proven their worth in image classification (Cireşan et al., 2012) and object detection (Szegedy et al., 2013), so machine hearing should not stay behind. DNN are currently used as a tool in speech enhancement and noise suppression but not yet as a tool to approximate a nonlinear deterministic system.

The approach of using nonlinear, fast DNN architectures as a tool in replacing the nonlinear, biophysically correct, but slow cochlear models has never been tried before. However, if successful, a real-time cochlear model has applications in automatic speech recognition (ASR), selective noise cancelling or as a front-end in next generation hearing-aids and robotics. Given its broad application area, this project may help to transform hearing applications to be more biophysically realistic.

1.1 Goal of this Master's Thesis

This study aims to apply (recent) machine learning (ML) techniques, more precisely deep neural networks (DNN), in the field of auditory modeling. Since the hearing pathway can be viewed as a cascade of various stages, each with its own specificities, this project is focussed on the replacement of human cochlear mechanics, one of the first stages in the auditory pathway. The choice for a compartmentalization of the hearing cascade -an approach that can also be found in current auditory models- will allow the model to grasp the specific biophysical features that are present in each step of the process.

Finding a DNN-representation that can account for the highly nonlinear cochlea should be a first step in the quest for a computationally fast (real-time), biophysically correct, auditory model. Even more, since several hearing-impaired profiles (mainly based on outer hair cell loss) have been proven to intervene in the working of the cochlea, one should aim for a DNN architecture that is both capable of capturing the normal-hearing profiles while having the capability of being made hearing-impaired to process those profiles as well. To achieve this, two research goals are presented:

- **A DNN replacement of a human cochlear model, for normal-hearing (NH) people**: A state-of-the-art auditory model that reaches the desired level of biophysical correctness, while including the cochlear nonlinearities and the frequency selectivity, will be selected first. Next, the optimal DNN architecture which replaces the NH human cochlear model will be developed. The NN will be trained using a combination of human speech fragments and reference model outputs. The suggested NN architecture should be able to reproduce the features that the reference model captures when presented with the same input speech stimuli or stimuli which are commonly adopted in cochlear mechanical studies.

- **A DNN replacement of a human cochlear model, for hearing-impaired (HI) people**: Since one of the criteria in selecting the reference model will be to allow for the inclusion of hearing-impairment in its computational stages, other DNN architectures, which account for the variety of hearing-impaired profiles, should be modeled. The second proposed HI DNN architecture will develop from the NH DNN using transfer learning techniques.

1.2 Thesis Outline

In this first chapter, the auditory modeling field and the thesis subject of this master's dissertation were introduced. The remaining chapters are organized as follows:

- Chapter 2 provides some more general context to the topic. Starting with a biological view on our hearing system, it is introduced how the structure and function of the ear can be modeled in practice. Next, a summary will be provided of computational cochlear models. The choice to select a reference model for this project is justified and reference model simulations -both for NH as HI people- will be shown.

- In chapter 3, the focus will be on artificial intelligence (AI). Several machine learning frameworks are defined and we show how they can be implemented in this thesis. Transfer learning in the scope of this project will also be touched upon.

- Chapter 4 provides the methodology in search for the optimal DNN structure for NH and HI profiles: starting from preprocessing the data, extracting features and selecting the general ML hyperparameters towards the crucial, variable model choices for the entire architecture.

- Chapter 5 lists the results of the different trained models that will be based on those parameter choices. Using different performance metrics, the best performing model is selected and will serve as a starting point for the second phase of this project. The results of the transfer learning procedure are also described here.

- A conclusion of this project, as well as looking at future work, will end this master's dissertation in chapter 6.

Chapter 2

From Human to Machine Hearing

2.1 From Sound to Meaning

Hearing can be seen as the process by which the ear transforms sound vibrations of the environment into auditory spikes that are directed towards the brain, where they are interpreted as sounds. However, in order for a sound to be transmitted to the central nervous system, the sound pressure undergoes different transformations that take place inside the human hearing pathway (the components of that pathway are depicted on Figure 2.1):

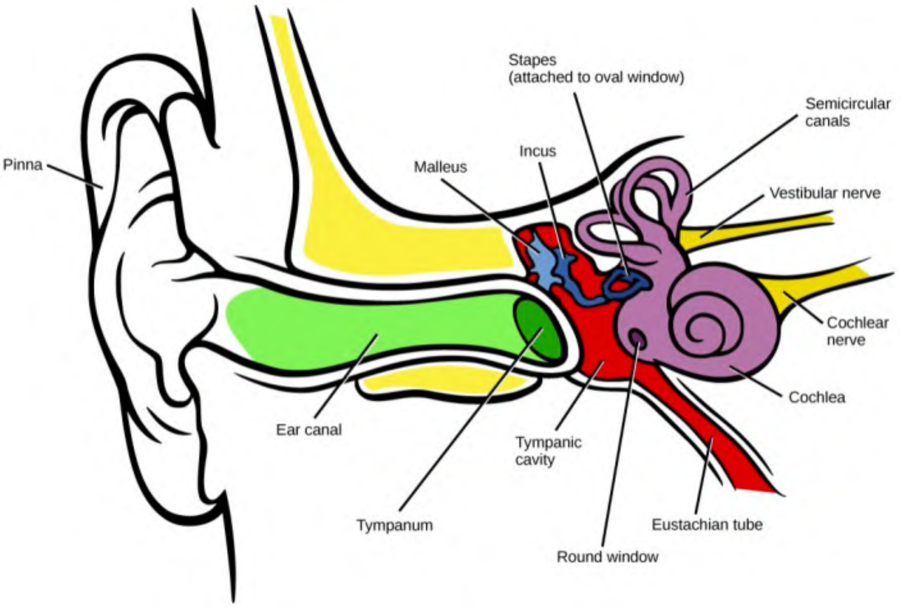


Figure 2.1: The hearing pathway (Daniel Rothmann, 2018)

Sound energy is first gathered by the visible pinna. This thin plate of elastic cartilage funnels the sound waves into the ear canal, where it, via air vibration, will reach the tympanum (or tympanic membrane). The portion of the sound that is absorbed by the membrane will vibrate the umbo, the central portion of the membrane, resulting in inwards and outwards bending. Since the umbo is linked to the handle of the malleus, the first ossicle, the sound wave undergoes a transformation from air vibration into mechanical vibration. This vibration is further passed onto the second (incus) and third (stapes) ossicle. The middle ear can be seen as an impedance matching device since the ossicles convert the lower-pressure tympanum sound vibrations into higher-pressure vibrations at the membrane on the other side of the tympanic cavity: the oval window.

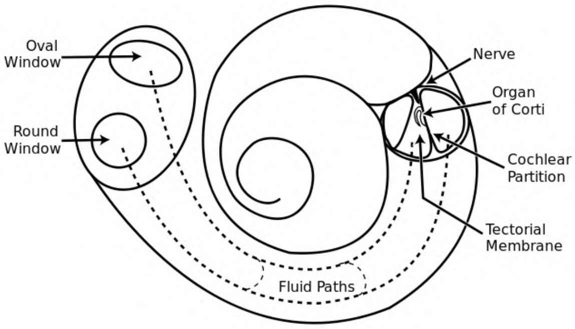


Figure 2.2: The cochlea (Daniel Rothmann, 2018)

Sound is concentrated onto the small oval window which is located at the base of the cochlea as can be seen on Figure 2.2. The same figure depicts also another window: the round window. This window is not attached to one of the ossicles, so it is able to 'freely' vibrate. This is a necessary feature hence in that way the, essentially incompressible, cochlear fluid is allowed to move. It is inside this inner ear structure that sounds makes its final transition: from the mechanical vibration in the middle ear to electrochemical transmission in the cochlear nerve. To complete the auditory pathway, the cochlear nerve fibers will lead towards the auditory cortex of the brain, where meaning is extracted.

2.2 The Cochlea

Figure 2.2 already showed the typical coiled structure that resembles a snail shell that is the cochlea. Once this small, yet complex, structure is cross-sectioned, the presence of three canals is revealed: the scala vestibuli, which is directly driven by the middle ear ossicles, the scala media and the scala tympani (Moller, 1994). All three channels contain a fluid (Pickles, 2013), with the scala vestibuli and the scala tympani containing perilymph, a fluid similar to the cerebrospinal fluid. The scala media contains endolymph, which resembles intracellular fluid. It is the movement of these fluids, upon vibrations of the oval window, that will cause basilar membrane movements.

The Basilar Membrane

The basilar membrane (BM) separates the scala tympani from the scala media. Once sound enters the fluid-filled cochlea, it causes deflection of the BM due to pressure differences between the scalae. Those pressure differences will be the origin of the formation of travelling waves. It was Von Békésy in the 1940s who carried out pioneering work revealing those waves in the cochlea (Von Békésy and Wever, 1960). He noticed that once a travelling wave, generated by a pure tone excitation, is propagating along the BM, the wave amplitude gradually increases until it reaches a peak at a certain location. Resonance occurs on this location after which a quick decay of the vibration happens. It is the frequency of the input tone that determines on which location along the BM this resonance occurs, making this whole process frequency specific. The BM is thus operating as a frequency analyser in the hearing pathway and responds to frequencies ranging from 20 kHz at the base of the cochlea to 20 Hz at the apex. This behaviour became quickly one of the most critical evaluation criteria for cochlear models (Ni et al., 2014).

At first, it was believed that frequency selectivity followed a linear pattern, depending on stimulus level, along the BM. This was not doubted until Rhode, in 1971, pointed out that the response of the BM is less frequency selective for higher level stimuli (Rhode, 1971). Over the years and with the availability of more sophisticated measurement systems, the theory of an active and nonlinear cochlea became more and more plausible. The active character of the structure was first raised by Gold (Gold, 1948) and was later confirmed by Kemp (Kemp, 1978) and is related to the outer hear cells in the organ of Corti.

The Organ of Corti

The translation of the movements of the basilar membrane into electrical impulses occurs in the organ of Corti, seen as the receptor organ of the ear (Moller, 1994). The organ is located on top of the basilar membrane and roughly contains 16,000 receptor cells, also known as hair cells:

- **Inner hair cells (IHC)**: One of which is present each 10 μm-long cross section of the organ of Corti (Elliott and Shera, 2012). This will convert motion into chemical signals that excite adjacent nerve fibers. Those will account for generating neural impulses which are sent to the brain via the auditory pathway.
- **Outer hair cells (OHC)**: Those exists, as is showed on Figure 2.3, in rows of three within the cochlear cross-section. The OHC play a more active role in cochlear dynamics. These types of hair cells will be important in this project since they are partially responsible for the nonlinear and compressive growth of BM vibrations with level. Also, OHC are closely related to hearing-impairment as their damage can result in cochlear gain loss, as will be discussed later.

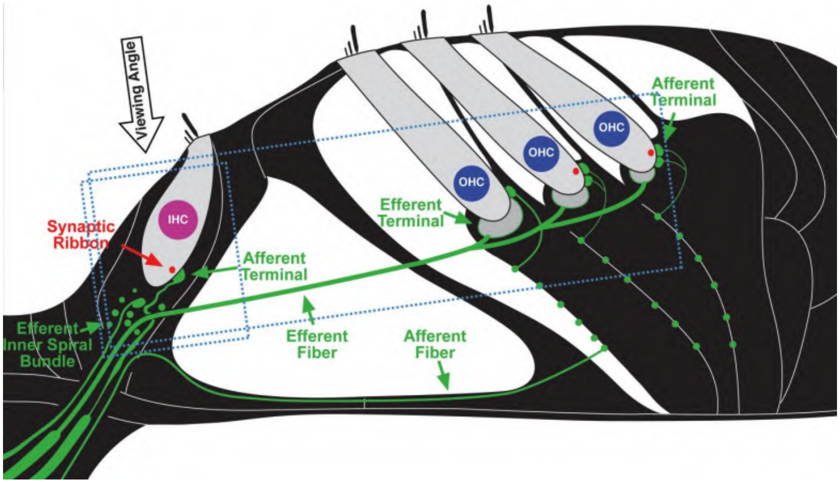


Figure 2.3: The Organ of Corti (Kujawa and Liberman, 2009)

Upon basilar membrane motion, the reticular lamina moves upward or downward, resulting in shear forces between the reticular lamina and the tectorial membrane that will bend the hair cells and will cause an ionic reaction that will depolarize the hair cell. It is this depolarization of the IHC that will lead to the releasing of neurotransmitters and the accompanying propagation of the auditory signal (Moller, 1994) towards the auditory nerve and ascending auditory pathways.

2.2.1 Nonlinearity

The cochlea can be seen as a highly nonlinear structure, since it accounts for two well-documented essential nonlinearities(Eguiluz et al., 2000):

- **Compression at high sound-level**: Whereas the cochlear response at the peak shows linear growth with level for low-to-moderate sound levels, the response grows compressively for high sound intensities. This is the most significant nonlinearity, and in engineering technology, it is said that the cochlea performs automatic gain control (Ni et al., 2014).
- **Sharper cochlear tuning for softer sounds**: Research (Rosen et al., 1998) shows that for softer sounds (sounds that are perceived less loud), the BM has a sharper filter shape, resulting in a sharper tuning.

The combination of these aspects of cochlear nonlinearity (i) together with a correct expression of the frequency analyser-role of the cochlea (ii) and the ability to capture the natural longitudinal coupling of the BM (iii), can be seen as valid criteria for a biophysically correct cochlear model. It is the accountability for these three criteria that will be demanded in the search of a cochlear model that can serve as a reference model in this project.

2.3 Auditory Models

Going from sound towards human perception spans the whole auditory pathway. However, it is often assumed in auditory models that the cochlear contribution towards auditory processing is the most important transformation in the pathway (Rhode, 1971). A cochlear model can be thought of as a tool with which, by using 'numerical experiments', researchers can obtain or predict cochlear output responses to different stimuli. But how to describe and replicate what the cochlea does? Should models be derived from the underlying physics (mechanical/biophysical models), or be validated against a range of measurements on real human auditory systems (perceptual models) (Lyon, 2018)?

2.3.1 Early Work

In general, auditory models have been developed to simulate characteristics of the human auditory system and to be used as realistic sound processors for machine hearing applications (Saremi et al., 2016). It was Von Helmholtz in 1875 who performed ground-breaking work on independent resonator theory and tuned filters (Von Helmholtz and Ellis, 1875). Based on these results, the critical band was introduced as a perceptual representation of the auditory filtering process (Fletcher, 1940). This was an inspiration to modelers to create filterbanks that consisted of several discrete filters to reproduce the available psychoacoustic data (Green, 1958). In this view, the cochlea was seen as a frequency analyser, comparing the working of the basilar membrane to a bank of highly tuned resonators, as could be found in musical instruments. This view however, total disregarded the role of the cochlear fluid and was not including longitudinal coupling inside the cochlea (Allen, 2001). Over the years, due to further advances in psychoacoustics (Stevens, 2017), the approach shifted and by means of connecting underlying hydrodynamics and calibrated parameters on human performance data, the models of today are able to represent a wide range of both linear and nonlinear aspects of the physiology of hearing with a rather basic and elegant set of circuits or computations (Lyon et al., 2010).

2.3.2 The Range of Auditory Models

There are two types of auditory models: Perceptual or functional models, that phenomenologically reproduce the overall input-output relation of the auditory system. However, since they lack one (or multiple) aspects of the criteria mentioned above, they are not explicitly modeling all of the underlying biophysical subprocesses (Saremi et al., 2016). Still, these models are the go-to models in applications today due to their low computational cost. The second class consists of biophysical models, which are focussed on implementing the correct biological processes at the cochlea and hence disregard computational simplicity for biophysical correctness.

Perceptual Models

To include frequency selectivity, perceptual models are based on linear or nonlinear filters that will be put in parallel or in a cascade to account for the tonotopic regions in the BM.

Gammatone filterbank A gammatone filterbank consists of a set of parallel filters that approximate the shape, sharpness and bandwidth of human auditory filters. This set of bandpass filters has a decreasing center frequency and an increasing sharpness as sound travels from the cochlear base (close to the middle ear) to the apex (Baby and Verhulst, 2018a). Figure 2.4 depicts the visualisation of such a filterbank. Gammatone filters (De Boer, 1975; Aertsen et al., 1980) were developed to yield an efficient realizable filter for applications. A shortcoming of this model is the fact that it lacks accuracy, which can be blamed on the absence of a structure that incorporates the nonlinear effects of the cochlea and thus does not emulate level-dependent characteristics of auditory filters.

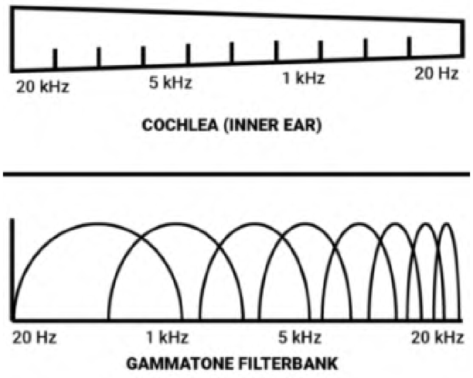


Figure 2.4: Gammatone Filterbank (Daniel Rothmann, 2018)

Mel-scale filterbank The mel-scale was developed based on results from human pitch-perception experiments from the 1940s. The sole purpose was to describe the human auditory system on a linear scale. A filterbank based on this mel-scale, as showed on Figure 2.5, can be used to derive the MFCC which are time-frequency energy bins applied in current ASR applications (Davis and Mermelstein, 1980). This spectrogram-like presentation of the cochlea is however not biophysically correct.

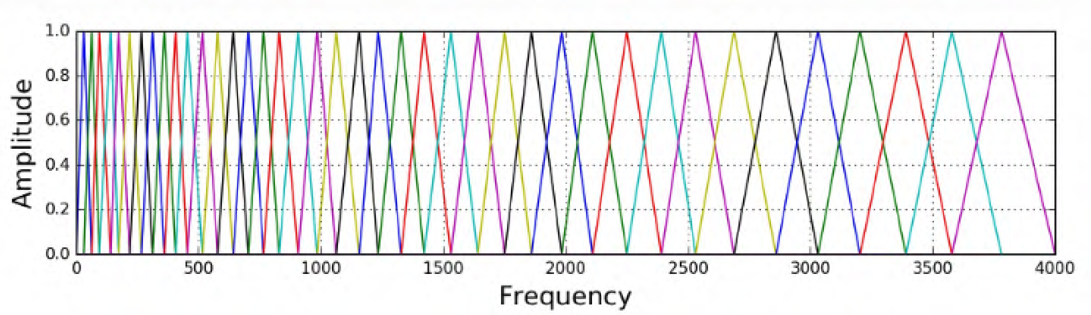


Figure 2.5: Filterbank based on a mel-scale (Haytham Fayek, 2016)

In the description of the gammatone filterbank, it was mentioned that there is an absence of nonlinear characteristics in these models. This belief grew when experimental evidence for the cochlear nonlinear character emerged (Kemp, 1978; Rhode and Robles, 1974), thus shifting the cochlear modeling towards nonlinear adaptations of the already available filterbank models.

As could be expected, those nonlinear extensions of basic filter models were more complicated and it was Lopez-Poveda, in 2005, who observed that users were forced to make a compromise between the complexity of a model and its ability to account for a wide range of physiological phenomena (Lopez-Poveda et al., 2005). This is evidently a suboptimal state where the model is forced to either lack a fast computation, that allows implementation in applications, or to lack important nonlinear characteristics of the human hearing pathway.

Biophysical Models

This trade-off is something that can also be found in biophysical models, since these models pay the price of a high computational load for their more accurate prediction of the cochlear nonlinearities and cascaded architecture.

Transmission-line The transmission-line (TL) model is founded on the Wegel and Lane model (Wegel and Lane, 1924). A transmission-line model discretizes the space along the basilar membrane length and describes this system in terms of coupled mass-spring-damper elements (Zweig, 1991). Those elements are put in a basic structure and it is a cascaded version of this structure (Figure 2.6) that forms the TL architecture. The model approximates the cochlear processing as a cascade of serial impedances and shunt admittances, respectively modeling the fluid coupling and mechanical filter properties in the cochlea (Verhulst et al., 2012). The numerical solution of these models are computed by means of differential equations (Altoe et al., 2014).

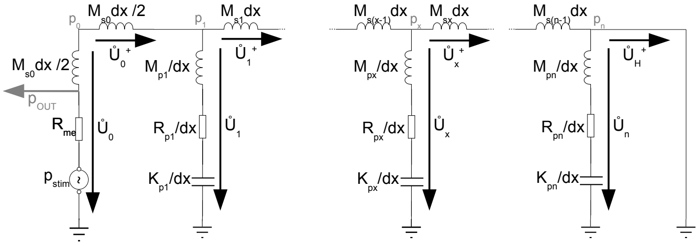
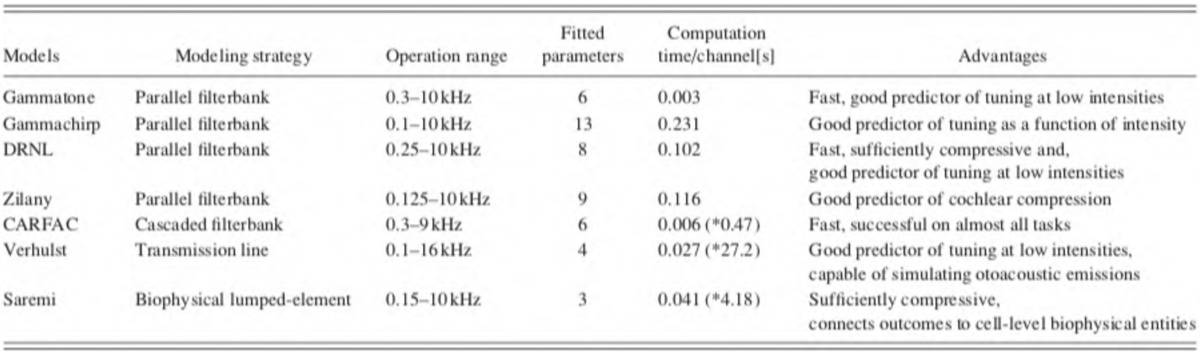


Figure 2.6: Basic element of the transmission-line model (Verhulst et al., 2010)

2.3.3 Model Selection

As put in the research goals, this study aims to replace a cochlear model with a deep neural network (DNN). But which model to chose? If the assumption is made that the nonlinear character of the DNN will be able to grasp every aspect of nonlinearity present in the reference model, the chosen model should be most resembling to the working of the human cochlea without taking computational effort into account. Saremi made a comparative study investigating seven types of auditory filter models (Saremi et al., 2016) and investigated the influence of model architecture on cochlear filtering by comparing their outputs on a fixed set of stimuli. Figure 2.7 displays the conclusion table of this investigation which shows that the TL model of Verhulst has the largest operation range (0.1 kHz - 16 kHz) and is a good predictor of tuning at low intensities. Additionally, as coupling was part of the initial quality criteria, the TL model architecture is preferred since only TL models take physical coupling between system elements into account, whereas the other class of models have independent channels and coupling is fully determined by the common input (Duifhuis, 2004).

The only current drawback of the Verhulst TL model is the default number of cochlear channels for the model. Although a larger number of channels can be seen as a more accurate



Models	Modeling strategy	Operation range	Fitted parameters	Computation time/channel[s]	Advantages
Gammatone	Parallel filterbank	0.3–10 kHz	6	0.003	Fast, good predictor of tuning at low intensities
Gammachirp	Parallel filterbank	0.1–10 kHz	13	0.231	Good predictor of tuning as a function of intensity
DRNL	Parallel filterbank	0.25–10 kHz	8	0.102	Fast, sufficiently compressive and, good predictor of tuning at low intensities
Zilany	Parallel filterbank	0.125–10 kHz	9	0.116	Good predictor of cochlear compression
CARFAC	Cascaded filterbank	0.3–9 kHz	6	0.006 (*0.47)	Fast, successful on almost all tasks
Verhulst	Transmission line	0.1–16 kHz	4	0.027 (*27.2)	Good predictor of tuning at low intensities, capable of simulating otoacoustic emissions
Saremi	Biophysical lumped-element	0.15–10 kHz	3	0.041 (*4.18)	Sufficiently compressive, connects outcomes to cell-level biophysical entities

Figure 2.7: Results of the Saremi investigation (Saremi et al., 2016)

description of the BM, the Verhulst model has a high relative computation time per channel as compared to, for example, the CARFAC model, which is almost 5 times faster. Taking into account the large difference in total channel numbers, this will render the CARFAC model substantially faster and hence explaining why current implementations are more drawn to this type of model instead of the computational heavy, but more biophysical TL model.

However, this project aims to reduce this required computational time by offering a model solution which incorporates the various features of the reference model, and for that, the TL model, more specifically the Verhulst model (Verhulst et al., 2018) will be selected as the reference model for this investigation.

2.4 The Adopted TL Model

The adopted TL model (Verhulst et al., 2018) depicted in Figure 2.8 accounts for the whole human auditory periphery, ranging from the middle ear towards the brainstem. The transmission-line architecture and elements are explained in detail in (Verhulst et al., 2015) and (Verhulst et al., 2018). The focus of this thesis will be on the TL cochlear model part of the architecture.

2.4.1 Overview of Cochlear Characteristics

Tonotopy The TL model divides the BM into 1000 sections, where the CF of each section was determined by the Greenwood map (Greenwood, 1961).

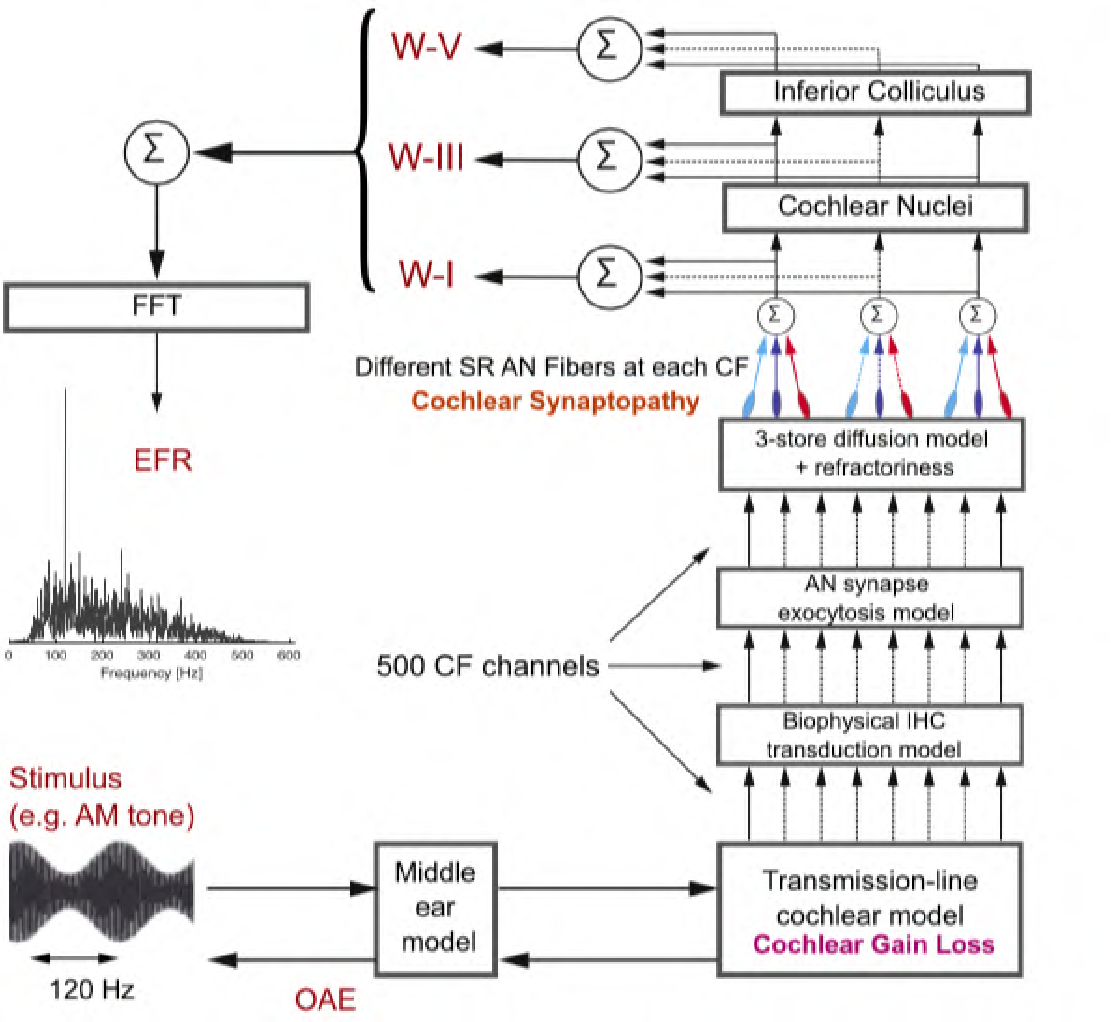


Figure 2.8: The architecture of the reference TL model (Verhulst et al., 2018)

Frequency tuning and longitudinal coupling Whereas most human auditory models use a functional parallel filterbank to capture cochlear frequency-tuning (Q_{ERB}), this model uses a TL architecture for the prediction of BM vibrations (Verhulst et al., 2015). In this way, they account for phenomena emerging from the coupled architecture of the BM and surrounding fluids in cochlear travelling waves (Verhulst et al., 2018). These phenomena include: two-tone suppression (Ruggero et al., 1992), asymmetrical filter shapes (Von Békésy, 1970) and phase changes in BM responses (Ruggero et al., 1997). TL front-ends benefit from their natural cascaded architecture, resulting in the desired coupling phenomena without introducing a second filter stage (Verhulst et al., 2018).

Compression An instantaneous nonlinearity was included in the model to account for the compressive behaviour observed in measured BM impulse responses (Recio and Rhode, 2000). This can be connected to the placement of the poles of the BM admittance in the frequency domain, since the pole location, relative to the imaginary axis of the complex plane, will determine the stability of the model (Verhulst et al., 2012). In this model, the concept of a pole movement depending on the BM motion is implemented, accounting for the compressive nonlinearity.

2.4.2 Hearing-Impairment

Recent studies are trying to understand how cochlear nonlinearities are affecting sound perception and how they are linked to hearing-impairment. People with damage to the OHC have issues with those nonlinear aspects of hearing: poorer audiometric thresholds, loudness recruitment and reduced frequency selectivity. This type of hearing loss is associated with stereocilia damage, the actual loss of OHC bodies or a metabolic reduction of the gain properties of OHC, which are all known to reduce the cochlear gain.

Crucial in the scope of this project is the possibility to render the reference TL model hearing-impaired. One of the model parameters which simulates cochlear gain is the above mentioned pole of the BM admittance, which can be adjusted in a CF-dependent manner to simulate wider cochlear filters associated with OHC damage (Verhulst et al., 2018). The different degrees of cochlear gain loss that can be simulated are depicted in Figure 2.9.

Two types of profiles are displayed: the sloping profiles that induce a sloping gain loss starting at a CF of 1 kHz and are common among the ageing population, and the flat hearing-impaired profiles (e.g., flat 35 dB gain loss), that has a constant gain loss over the entire frequency spectrum. The reference model implementation applies a table which relates cochlear gain reductions as in Figure 2.9 to values of the pole of the considered BM admittance at each CF, such that any desired gain loss, within the audiometric frequency range, can be translated into an associated pole location and can be used for HI simulations.

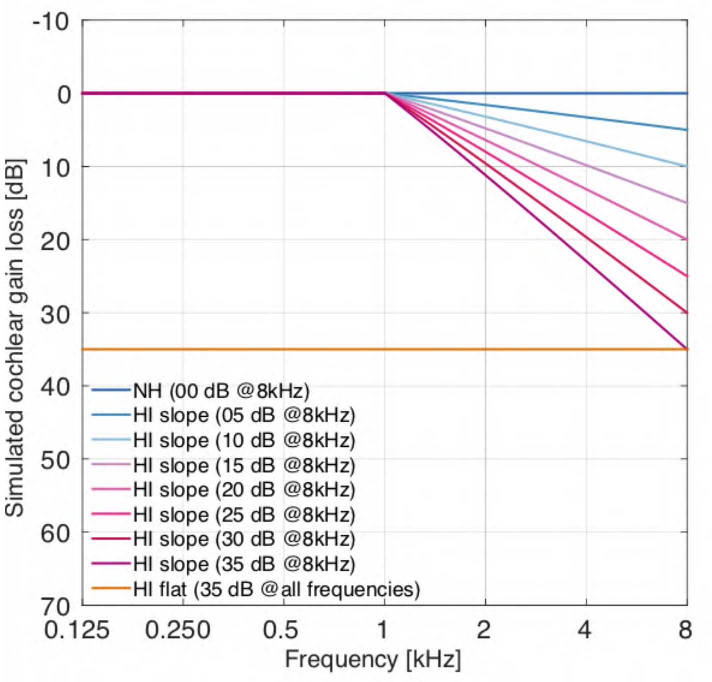


Figure 2.9: Possible HI profiles that can be simulated with the Verhulst model (Verhulst et al., 2018)

This chapter gave a general view of the human hearing system and showed how different auditory models are incorporating cochlear mechanics and nonlinear features. The choice for the Verhulst et al. (2018) model, as a reference model for this project, was explained and justified. The next chapter will focus on some general machine learning frameworks and how these are implemented in the quest of making a nonlinear NN representation of the cochlear TL model for people with NH or HI profiles.

Chapter 3

Neural Networks

Artificial neural networks or neural networks (NN) can be seen as a set of algorithms which are based on how the human brain recognizes patterns. NN can interpret various sensory data (e.g., images, sounds, time series), as long as this data can be transformed into a numerical representation. This data is then provided as an input to a first layer where it flows into a network based on machine perception. The inspiration of the formation of those NN is primarily drawn from cognitive science which attempts to combine perspectives of biology, neuroscience, psychology and philosophy to gain a greater understanding of the human cognitive faculties (Daniel Rothmann, 2017).

3.1 Deep Neural Networks

A returning issue of conventional NN is the fact that they are limited in their ability to process natural data in their raw form. Constructing a feature extractor that is able to transform this raw data into a suitable representation, from which the subsystem can recognize patterns, requires careful engineering and considerable domain expertise (LeCun et al., 2015). Once it became clear that NN could serve as a feature extractor themselves, and could be used for more complex functions, when the number of hidden layers was expanded -since different layers may perform different kinds of transformations (Hinton et al., 2006)-, deep neural networks (DNN) were formed (Heaton, 2017).

The key aspect of deep learning is that these hidden layers of features, are not designed by humans: they are learned from data using a general-purpose learning procedure, a procedure that allows to represent data with multiple levels of abstraction via the multiple available processing layers (LeCun et al., 2015). These intricate structures in large datasets are discovered using a backpropagation algorithm. This algorithm indicates how a machine should adapt its internal parameters in each layer of the network. It brings a concept of depth into the NN and hence makes it possible to form a hierarchical representation of a certain problem (Cevora, 2019). A hierarchical structure which is known to be a prominent feature of information processing in the human brain (Riesenhuber and Poggio, 1999).

3.1.1 Applications in Hearing Field

Deep learning methods are making major advances in solving problems that have resisted the best attempts of the AI community for many years (LeCun et al., 2015). Over time, DNN were -successfully- implemented in machine vision and considering the benefits of getting inspired by human processes in vision, we stand to gain from a similar approach for machine hearing with neural networks (Daniel Rothmann, 2018).

Some examples of current applications in the hearing field:

- Recent advances in deep neural network-based learning architectures are shown to outperform most of the conventional **speech enhancement** approaches (Xu et al., 2015; Lu et al., 2013; Sun et al., 2017). Thanks to their nonlinear representations (multiple hidden layers) these architectures are enabled to model the complex degradations in the captured speech signal.

- Conditional generative adversial networks (cGAN) (Baby and Verhulst, 2018b), based on the SEGAN model (Pascual et al., 2017), have showed to provide an alternative framework to yield promising **noise suppression** performance (Michelsanti and Tan, 2017; Donahue et al., 2018).

- Recurrent neural networks, which employ LSTM cells, are used, leveraging upon its memory structure, to capture **temporal contexts** within a larger training data set (Baby and Verhulst, 2018a).

- Using a DNN, MIT researchers created a model able to replicate **human performance on auditory tasks** such as identifying a musical genre (Kell et al., 2018).
- Algorithms are formed with an aim of **speech separation** of multiple speakers (Yu et al., 2017).

The rise of these projects can be accounted to several factors like an ever-increasing computational power, an increase in the amount of big data available for training these algorithms and constantly emerging training methods. But the applications are not limited to those mentioned above.

3.2 Model Architecture

The DNN architecture proposed in this master's thesis can be seen as a combination of two machine learning frameworks: convolutional neural networks (CNN) and autoencoders (AE).

3.2.1 Convolutional Neural Networks

Convolutional neural networks, initially developed by Yann LeCun in 1989 (LeCun et al., 1989), are based upon a structure known as the neocognitron, developed by Fukushima (Fukushima, 1988) as an attempt to build a functional artificial visual system. The name CNN indicates that the network employs a mathematical operation called convolution (Goodfellow et al., 2016), a specialized kind of linear operation that essentially means moving a filter across the data to identify features in the input (Cevora, 2019). CNN can also be placed under the definition of artificial neural networks since they are formed by neurons that take in weighted sums of inputs and output a certain activity level, a nonlinear function of the input value.

The typical architecture of a CNN consists of a series of different stages, of which the majority are formed by two types of layers: convolutional and pooling layers. Units in a convolutional layer are organized in so called feature maps. Feature maps that are connected with local patches in the previous layers via a set of weights called a filterbank. This organisation is also shown in Figure 3.1. In combination with the filterbank operation, a weighted sum is taken and then passed through a nonlinearity (e.g., PReLU). This filtering operation is called a discrete convolution, hence the name. The role of this layer is to detect local conjunctions of features in the previous layer.

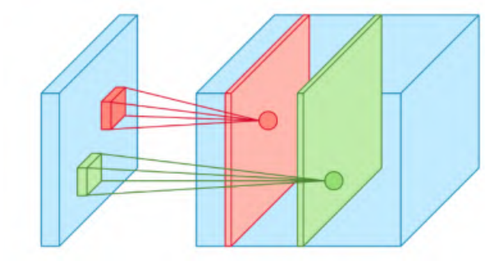


Figure 3.1: Example of a convolutional operation: the patches on the left are the kernels, partitions of the filterbank, the coloured layers on the right are the respectively accompanying feature maps (Arden Dertat, 2017b)

Pooling layers on the other hand, have the goal to merge semantically similar features. In a max pooling operation, the maximum activity of a certain local patch is selected and placed in another, smaller feature map. This will compress the previous layer and create an invariance to small shifts and distortions.

Typically to form a complete CNN architecture, two or three stages of convolutional, nonlinearity and pooling layers are stacked, followed by some more convolutional and fully-connected layers, which are layers that can be seen as standard feedforward networks, without spatial layout or restricted connectivity. The weights of the various filterbanks are the trainable parameters, which can be updated by means of backpropagation of gradients.

The Choice for CNN

Convolutional neural networks were chosen in this project over other NN approaches since the convolutional (filter) architecture is most closely related to the task we want it to perform: cochlear filtering. Furthermore there are four key ideas behind the usage of CNN that take advantage of the properties of natural signals (LeCun et al., 2015): Local connections, pooling, the use of several network layers and shared parameters. Parameter sharing also has other advantages, since it does not affect the runtime of forward propagation through the network but does reduce the storage requirements of the model (Goodfellow et al., 2016).

3.2.2 Autoencoders

An autoencoder (Bourlard and Kamp, 1988; Hinton and Zemel, 1994) (Figure 3.2) is the combination of an encoder function, which converts the input data into a different representation, and a decoder function, which turns the new representation back into the original format (Goodfellow et al., 2016), meaning that the targeted output of the autoencoder is the input itself. If an input can be reconstructed at the output without having a great loss term (the difference between the desired and the received output), the network has learnt to encode it in such a way that the internal representation contains enough meaningful information. The use cases of this framework are data denoising and dimensionality reduction.

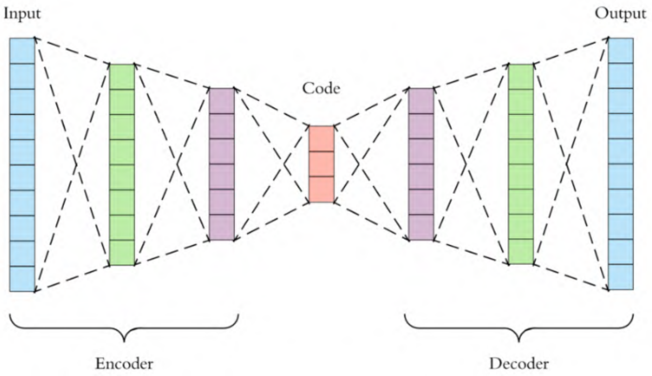


Figure 3.2: Autoencoder architecture (Arden Dertat, 2017a)

3.3 AECNN

The two previous frameworks can be combined to form an autoencoder convolutional neural network (AECNN), the type of DNN that will be used in this project as a starting framework. Figure 3.3 displays this architecture.

On the left, the audio input data (sample length 2048) is displayed. This data is first processed by an encoder architecture, comprised of a few CNN layers. Each CNN layer consists of a set of filterbanks followed by a nonlinear operation on the obtained filter outputs. The filter weights used in this CNN model are the trainable parameters and are updated in the training phase. As can be seen, the temporal dimension of the audio is reduced by a factor of two in every layer, resulting in a condensed representation of the input after the encoding part of the architecture (denoted in yellow on the figure). The halving of the temporal dimension can be assigned to the usage of strided convolutions in the CNN layers, these are resulting in a filter operation every two samples.

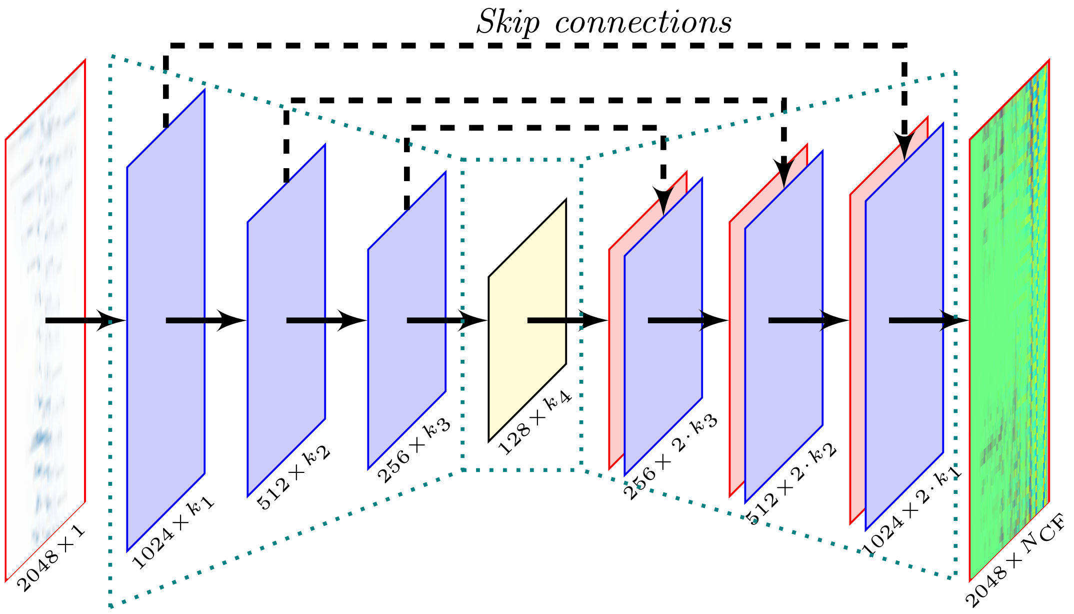


Figure 3.3: The AECNN architecture used in this project (note that normalization and activation layers are omitted)

In general, after N CNN layers in the encoder, the audio signal is compressed into a representation of size L/2N x k_N. With L representing the starting sample length and k_N being the number of kernels in the N^{th} CNN layer.

This encoded representation is then mapped to the output using a decoder system, which uses transposed-convolutional layers, where the temporal dimension is now doubled every layer. The decoder also consists of N deconvolution layers, yielding back the starting temporal dimension of L samples. The number of filters used in the final CNN layer of the decoder is set to be equal to the number of cochlear sections (N_{CF}) that are provided in the cochlear reference model. Thus returning an output of the proposed AECNN model of size L x N_{CF}.

Since the encoder uses strided convolutions, it might lose some important information such as temporal alignment and phase information. To account for this, the proposed architecture uses U-shaped skip connections to bypass this information from the encoder to the decoder layers (He et al., 2016). In the specific scope of this research, the use of the skip connections can be seen as pivotal since the phase information, that is transferred via these connections, coupled to an audio sample is crucial for speech intelligibility in noisy listening conditions. These connections also facilitate several direct paths between the input and the output, which in turn helps the architecture to combine different levels of nonlinearities which is essential in approximating the nonlinear coupling and the level-dependent tuning of the human cochlea.

3.4 Transfer Learning

Transfer learning (Pan and Yang, 2009) 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. Due to the fact that both tasks are related, transfer learning assumes that the learned features of the first task, are general and hence transferable to the second model.

In this project, this technique is used when an already developed and trained DNN model serves as a starting point to construct another DNN. More specifically, the structure, weights and parameters of the first obtained AECNN for NH people will be used to form a AECNN capable of predicting the HI cochlear response for the different hearing-impaired profiles. Since people with hearing-impairment will still posses hearing characteristics that resemble the ones of the normal-hearing profiles, the responsible features for capturing these characteristics can be transferred.

The use of transfer learning will save time on both the feature extraction, since only a limited number of training utterances compared to the starting model will need to be collected and processed, and on the training time itself.

Chapter 4

Methodology

4.1 Data

4.1.1 Dataset

The speech material that will be used in the training and testing phase stems from the TIMIT training dataset:

"The TIMIT corpus of read speech is designed to provide speech data for acoustic-phonetic studies and for the development and evaluation of automatic speech recognition systems. TIMIT contains broadband recordings of 630 speakers of eight major dialects of American English, each reading ten phonetically rich sentences. The TIMIT corpus includes time-aligned orthographic, phonetic and word transcriptions as well as a 16-bit, 16kHz speech waveform file for each utterance." (Garofolo, 1993)

The choice of the TIMIT corpus over other datasets (e.g., Wall Street Journal; Paul and Baker, 1992) was made considering the presence of various naturally spoken speech combinations, something that is absent in quality filtered databases, that limit the whole database to a partition with only the 64,000 most frequently occurring words (Sakti et al., 2009).

4.1.2 Data Processing

Figure 4.1 displays the pathway that will be used to obtain the desired NH AECNN capable of replacing the TL model.

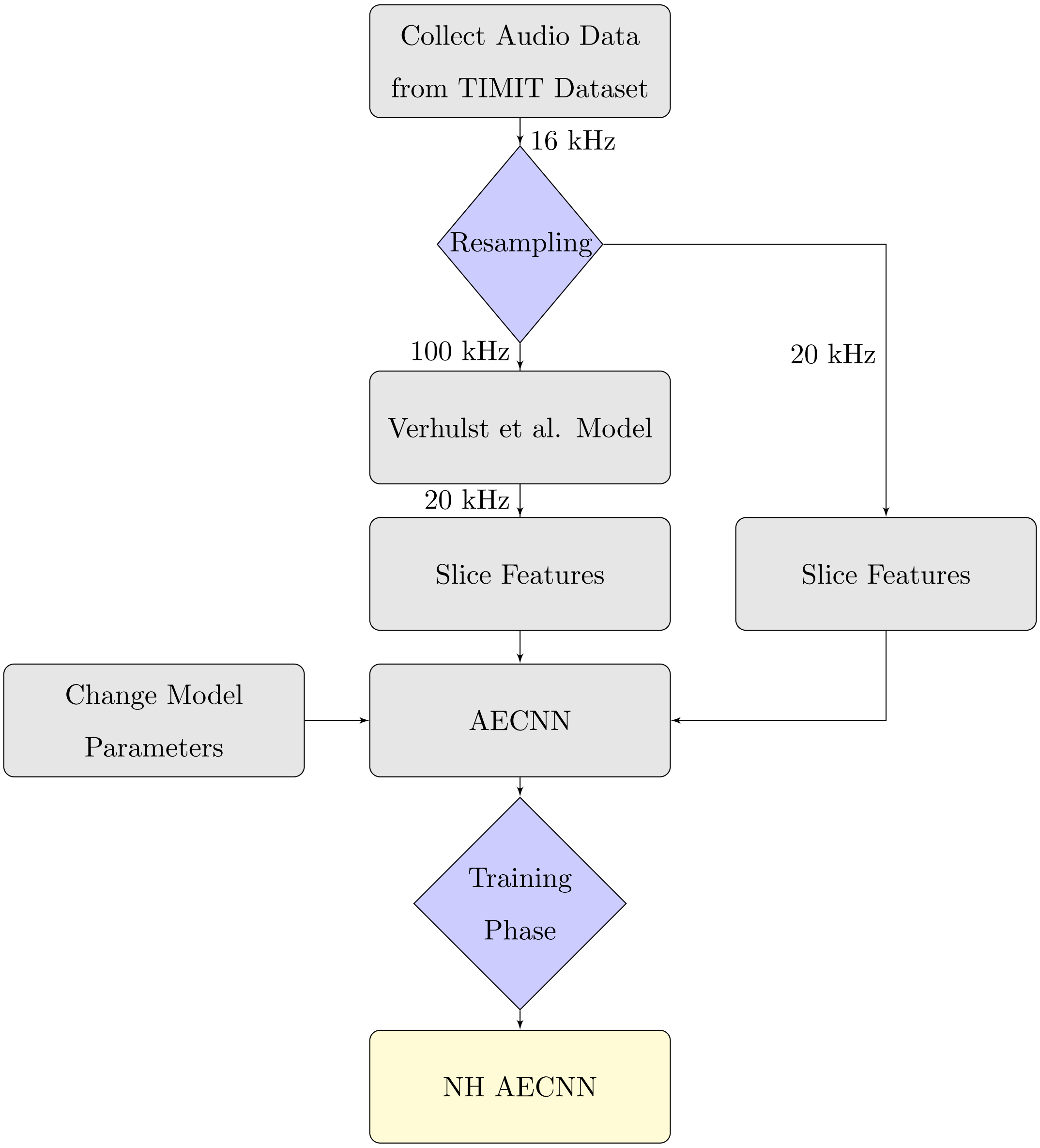


Figure 4.1: The data and training pathway used in this project to form a NH AECNN

Initial Steps

Data collection The audio collected from the TIMIT dataset (2310 sentences) is adjusted such that the root mean square energy of the signal has a sound pressure level of 70 dB. This sound level is best resembling standard conversational speech levels and includes both louder and more silent instantaneous amplitudes. Those amplitude variations need to be present in the training phase for the model to learn to capture the nonlinearities associated with cochlear processing.

Resampling The TIMIT dataset has a sampling frequency of 16 kHz, whereas the TL model demands an input sampling frequency of 100 kHz, hence an upsampling is performed on the TIMIT data. Since the TL model will, on the other hand, output a signal with a sampling frequency of 20 kHz, the same TIMIT data needs to be upsampled to 20 kHz as well, this to have the same sampling frequency in the input and output data for the AECNN.

The reference TL Verhulst et al. model The upsampled data is given as an input to the Verhulst model. The model will predict the basilar membrane displacements across CF. Although the model is able to return 1000 CS, the choice is made to only output 201 sections. This to reduce training time in further stages. The CF of these 201 sections are based on the Greenwood map Greenwood (1961) and span the frequencies between 113 Hz and 12,010 Hz. Appendix A gives the entire list of CS with their accompanying CF.

Slice Features Since the proposed AECNN architecture is set up to only process input data with a sample length of 2048, both the TIMIT data, as well as the received output of the TL model is sliced in chunks of 2048 samples and stored.

4.2 AECNN for NH

The sliced TIMIT data is given as an input to the AECNN whereas the sliced features of the TL model function as reference data in the training phase. Before entering this training phase, the exact AECNN architecture should be constructed. A distinction should be made between fixed AECNN parameters, that will be kept constant in this investigation (Table 4.1), and the variable parameters, also referred to as hyperparameters which generally affect the time and memory cost of running the algorithm (Goodfellow et al., 2016) and can thus be seen as crucial design parameters. The addressed hyperparameters in this investigation are the learning rate, layer depth, filter length and NL activation function of the NN architecture.

Table 4.1: Fixed parameters AECNN framework

Parameter	**Value**	**Summary**
Window length	2048	Length of the input audio sample. Based on the nature of envelope fluctuations in speech signals and context between words.
Input features dimension	1	1D signal input.
Stride	2	A step size of 2 for the filter operation. Downsizing and resizing the temporal component by (roughly) a factor of 2.
Padding	Same	Add zero padding to fit the filters on the input shapes.
Kernels/Filters per layer	128	Design choice.
Output features dimension	201	Number of cochlear sections (CS) in the TL output data and thus resembling AECNN output features. Each CS corresponds with a CF between 100 Hz and 12 kHz (see Appendix A).

4.2.1 Learning Rate

The learning rate is perhaps the most important hyperparameter since it controls the effective capacity of the model in a more complicated way than other hyperparameters (Goodfellow et al., 2016). Plotting the training error for a model in function of the learning parameter will give a U-shaped curve, meaning that both too small, and too high learning rates will yield suboptimal solutions.

In this research, the investigated learning rates are: **0.001 - 0.0004 - 0.0001**

4.2.2 Layer Depth

The layer depth is coupled to the number of hidden layers that make up the DNN. Increasing this number will increase the representational capacity of the model but will evidently also increase the time and memory cost of every operation of the model. This trade-off should be monitored and the design choice should be made accordingly.

In this research, the investigated layer depths are: **4 layers - 6 layers - 8 layers**

4.2.3 Filter Length

A fixed parameter of the framework is the number of filters per layer but the width/length of these filters can be adapted. Again, a kind off trade-off is taking place where increasing the filter length will increase the number of parameters in the model. This increase in tuneable parameters will increase the runtime and computational cost but also the overall performance of the model.

In this research, the investigated filter lengths are: **31 - 63 - 127**

4.2.4 Nonlinear Activation Functions

Most NN approaches to model biophysical systems use a rectified linear unit (ReLU) as activation function. However to correctly capture the cochlear nonlinearities, the shape of the activation function is crucial in this design. In essence, the function should cross the x-axis to capture zero-crossings, and in standard activation functions (e.g., sigmoid, ReLUs) this is not the case and hence the model is unable to capture level-dependent compression and the level-dependency of cochlear tuning. The two activation functions used here do have this zero-crossing, as shown in Figures 4.2 and 4.3.

In this research, the investigated nonlinearities are: **PReLU - tanh**

4.2.5 Training Phase

Once the AECNN architecture is chosen, the training phase, of which the fixed parameters are depicted in Table 4.2, will start.

In short, the models are trained such that the mean absolute error (L1 loss) between the predicted and reference TL cochlear outputs, to the same stimulus, is minimized.

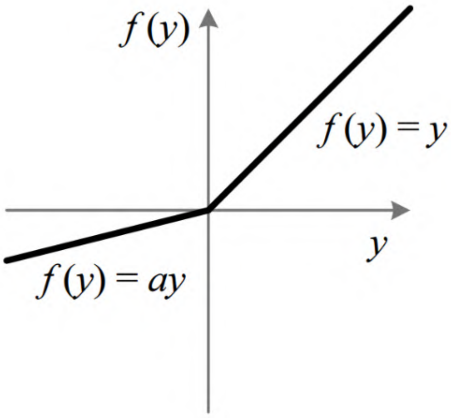


Figure 4.2: PReLU activation function (He et al., 2015)

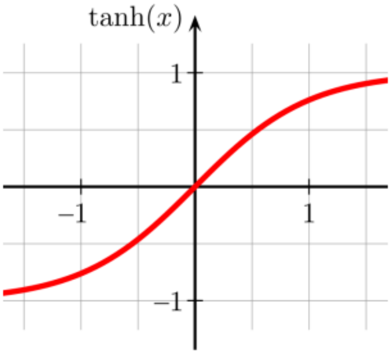


Figure 4.3: Tanh activation function (Kishan Maladkar, 2018)

In the very first run of the training phase, the weights of the filters in the tuneable layers are randomized. After the completion of the first batch, where 32 input-output combinations were showed to the model, the weights are updated via stochastic gradient descent based on backpropagation (Ganin and Lempitsky, 2014) to minimize the L1 loss term. The same procedure is repeated until the entire training set is presented to the model: this marks one epoch. The epoch number is incremented and the training phase is resumed for the indicated number of epochs. After completing the last epoch, the best performing AECNN architecture (based on the lowest L1 loss) is stored together with its weights and can be used afterwards for evaluation or as a starting point to develop the HI version of the AECNN (see further).

Table 4.2: Fixed parameters training phase

Parameter	**Value**	**Summary**
Optimizer	Adam (Kingma and Ba, 2014)	An adaptive learning rate optimization algorithm, fairly robust to the choice of hyperparameters.
Number of epochs	20	Number of times that the learning algorithm will work through the entire dataset. Design choice to keep the total training time reasonable.
Batch size	32	Number of training utterances that the model has to work through before the tuneable parameters are updated.
Shuffle	Batch	The shuffling of the training data before each epoch is done in batch-sized chunks.
Loss function	L1 loss	the mean absolute error between the desired and predicted output is used as loss term.

The entire architecture and training framework is developed using a Keras (Chollet et al., 2018) machine learning library with a TensorFlow (Abadi et al., 2016) back-end. Appendix B depicts an example Keras model summary output for an 8 layer AECNN model.

4.3 AECNN for HI

The different types of cochlear gain loss were already mentioned in Section 2.4.2. This project will focus on four of the most severe HL profiles that can be implemented in the reference model: slope 25/35 and flat 25/35.

The pathway depicted on Figure 4.4 is similar to the NH pathway explained above, one crucial difference however is the rendering of the HI version of the reference TL Model model based on the selected HL profile.

The AECNN is a selected NH model from the first part of this project. The architecture is exactly the same and the weights of the model are put as starting values. One possible

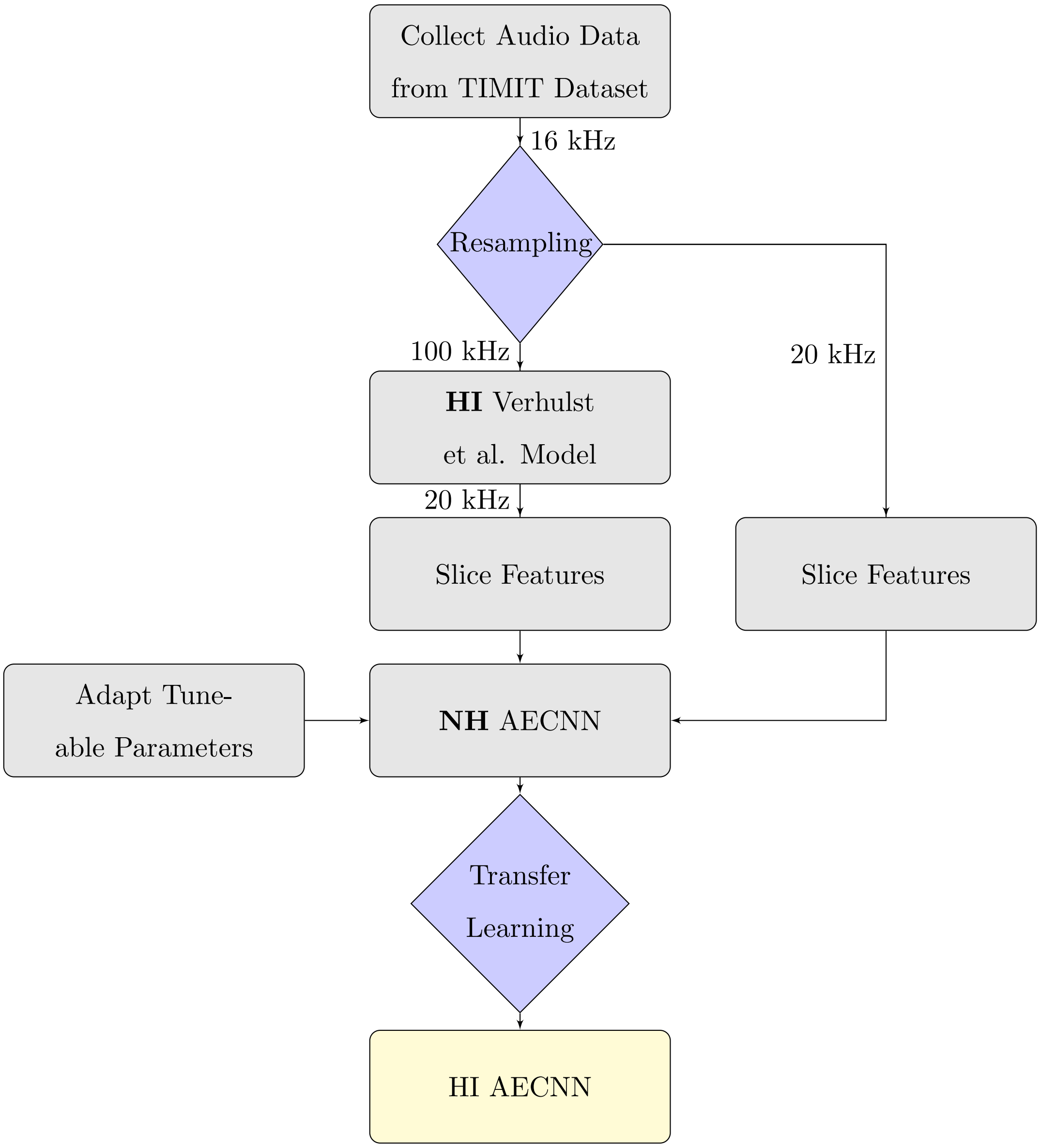


Figure 4.4: The data and training pathway used in this project to form a HI AECNN

variation in the DNN architecture is the choice of tuneable parameters by freezing the weights of particular hidden layers and hence not updating these during the transfer learning phase.

This fixation will not only decrease the number of adjustable parameters but also the computational time needed to perform the training phase. The challenge, however, lies in finding a compromise between enough tuneable parameters to grasp the HI profile specificities and an acceptable computational time, keeping further implementations in mind. After 20 epochs, where again the L1 loss of the model performance is minimized, the best model is exported and saved.

4.3.1 Transfer Learning Utterances

The biggest benefit of transfer learning is the drop of training utterances needed to correctly train the DNN architecture. Whereas the NH training phase used 2310 training utterances, a much lower number suffices here since hearing aspects, that will not be altered by cochlear gain loss, will already be present in the trained NH AECNN.

In this research, the investigated number of utterances are: **50 - 100**

Chapter 5

Results

5.1 Performance Measurements

The previous chapters explained the AECNN model architectures, their parameters and how they are being implemented in this research. This chapter will evaluate and compare the performance of the trained NH and HI models. The evaluation and selection metrics for the 'best performing model' will be described first.

5.1.1 General Attributes

Before the start of the training phase, and directly after the completion of the training cycle, some generic attributes will be available for each model:

- **Number of (trainable) parameters**
- **Loss term**
- **Time per epoch**

The number of trainable parameters plays a role in the general performance, since a greater number will permit the DNN to capture more specific characteristics that are present in the audio input. Linked to this resembling ability, the loss term also experience variations, since this indicates the mean absolute error (L1 loss) during training. A lower loss term indicates a better performance on the training set. Upfront it seems desirable to keep the time per epoch (the time for one training iteration on the entire dataset) as low as possible since an increase in time will require a higher computational power and more storage room for the entire model. Since every depicted model was trained for 20 epochs, the time per epoch is 5% of the total training time for a model.

5.1.2 Performance on Basic Auditory Stimuli

An audio (or speech) fragment can be seen as a combination of basic components such as click impulses and pure tones varying in frequency. Since the DNN is not trained (the TIMIT corpus only contains speech samples) on those types of basic stimuli, commonly used in cochlear mechanics studies, they are a good performance measure to evaluate how the trained models are performing.

Click Stimulus

The models will be presented with a click impulse lasting for 100 μs, of which the shape -only a narrow pressure spike- can be seen on the left part of Figure 5.1. The sound pressure of this stimulus can be calculated by means of Equation 5.1, where p_0, the reference pressure for the decibel (dB) scale, is equal to $2e^{-5}$ Pa. L denotes the sound pressure level (SPL) in dB. Since the sound level of the data used for training is set to 70 dBSPL, the same value is used in this equation. To have the same peak-to-peak amplitude of a pure tone sinusoid with amplitude 1, the value of the click stimulus (which is located between 0 and 1) is calibrated into a peak-equivalent SPL, explaining the factor 2 at the start of the equation.

$$Stim_{click} = 2 * \sqrt{2} * p_0 * 10^{\frac{L}{20}} \tag{5.1}$$

The right part of Figure 5.1 depicts the cochlear dispersion of the TL model for this particular click stimulus, this will be the desired output of the trained NH AECNN. These type of plots will be used throughout this chapter: the output plot of the TL model (and in further sections the AECNN models) will have the same temporal dimension but will depict the BM movements for each of the 201 cochlear sections (CS). The 201 CF that are linked to these CS are mentioned in Appendix A.

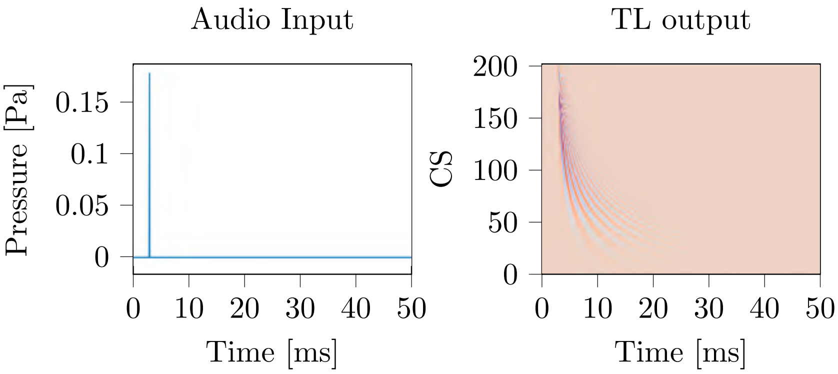


Figure 5.1: Click stimulus input and TL output. (Left) Input pressure [Pa] in the time domain [ms] for a click stimulus of 100 μs. (Right) Output response of the TL model: BM displacements for the selected 201 CS (see Appendix A for the corresponding CF).

Pure Tone Stimuli

Pure tone stimuli of both 1 kHz and 4 kHz will be presented to the models and their stimulus pressure is formed via Equation 5.2. The additional term in comparison with the click stimulus equation is the sinusoid, which is formed based on the frequency of the input tone and depends on the length of the vector t (a time vector of length 2048 (input sample length)).

$$Stim_{tone} = p_0 * \sqrt{2} * 10^{\frac{L}{20}} * \sin(2 * \pi * f_{tone} * t) \tag{5.2}$$

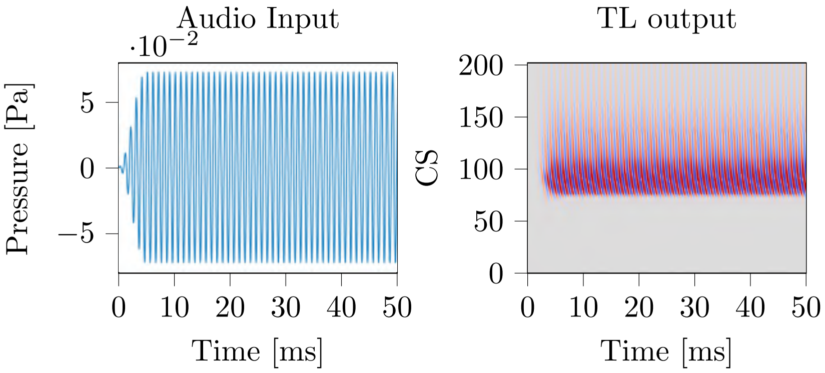


Figure 5.2: 1 kHz tone stimulus input and TL output. (Left) Input pressure [Pa] in the time domain [ms] for a pure tone stimulus of 1 kHz. (Right) Output response of the TL model: BM displacements for the selected 201 CS (see Appendix A for the corresponding CF).

The tones, together with their respective cochlear dispersion are depicted in Figures 5.2 and 5.3. As can be derived from these input pressure forms, the onset (and offset, but not depicted) of the pure tone is multiplied with a Hanning window of 10 ms.

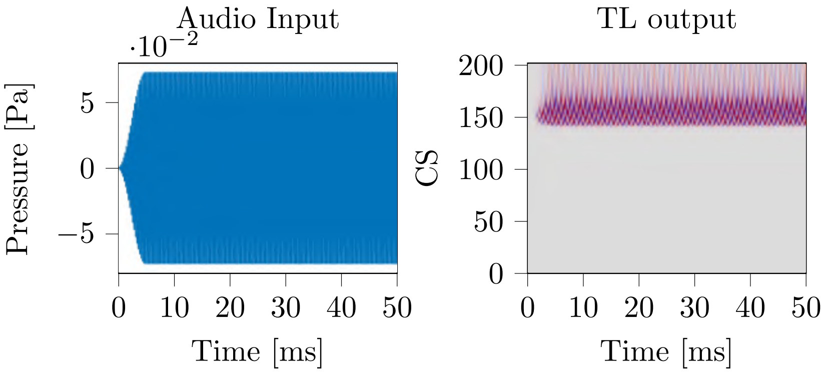


Figure 5.3: 4 kHz tone stimulus input and TL output. (Left) Input pressure [Pa] in the time domain [ms] for a pure tone stimulus of 4 kHz. (Right) Output response of the TL model: BM displacements for the selected 201 CS (see Appendix A for the corresponding CF).

MSE on Input Stimuli

Besides the visual inspection of how the various AECNN outputs are resembling the ones of the TL, the MSE will be calculated of the predicted 411,648 values (201 cochlear sections x 2048 samples). This relative measure will allow to compare the models in a numerical manner, based on their predictive performance on the above stated stimuli.

RMS of Output - Excitation Patterns

This performance measurement will consist of calculating the RMS of each (201 in total) filter channel's outputs in response to the basic input stimuli. These RMS values are subsequently plotted according to their corresponding CF on a frequency axis, giving rise to a so called excitation pattern. Doing this for multiple sound levels (ranging from 10 dBSPL to 90 dBSPL in this project) will allow to visualize the level-dependency in between excitation patterns. This level-dependency should follow a nonlinear behaviour (due to cochlear compression) across the levels. Based on a visual inspection, this performance measure will be taken into account.

5.1.3 MSE Performance on Test Set

Since the trained DNN should also be able to correctly predict cochlear outputs for speech fragments that were not part of the training dataset, the models are not only tested on the basic input stimuli mentioned above. Hence a test set, consisting of 64 unseen speech fragments, was selected. A segment of 2048 samples was chosen from each of the 64 fragments and was fed to both the reference TL model and the trained neural network architecture. Here again, the MSE of all 411,648 samples are calculated and used in the addressing the overall performance.

Figure 5.4 shows the audio input of one of those segments and the corresponding TL output.

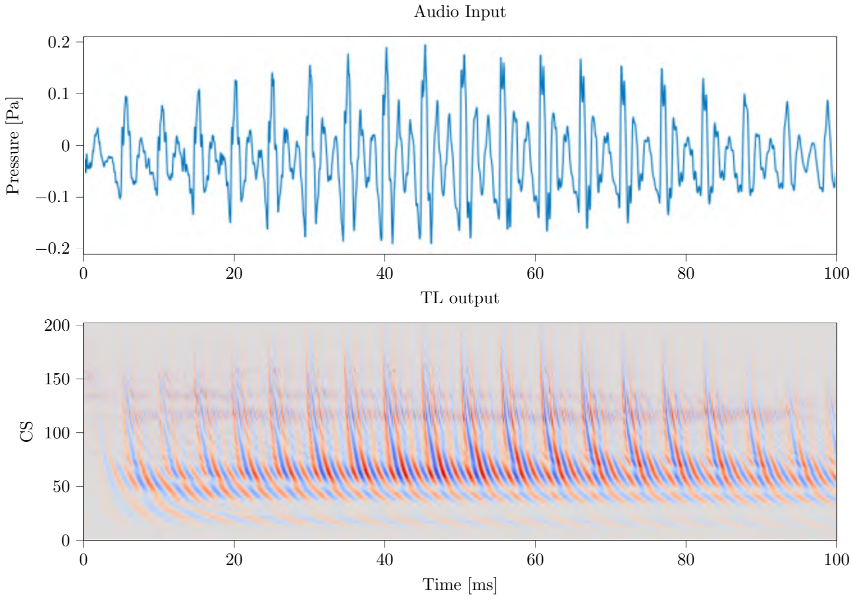


Figure 5.4: Test set example with corresponding TL output. (Top) Input pressure [Pa] in the time domain [ms] for a speech fragment part of the test set. (Bottom) Output response of the TL model: BM displacements for the selected 201 CS (see Appendix A for the corresponding CF).

5.1.4 Q_{ERB}

The final performance measure is the resulting equivalent rectangular bandwidth or the Q_{ERB}. This can be used as a quantification of the sharpness of cochlear tuning (Shera et al., 2010) as a function of level, one of the attributes of the cochlea that was demanded to be included in the trained DNN. At the same time, this Q_{ERB} value as a function of frequency follows a typical curve for humans (Shera et al., 2002). This value is described as:

$$Q_{ERB} = \frac{CF}{ERB} \tag{5.3}$$

Where CF is again the frequency coupled to a certain cochlear section and ERB the, CF-dependent, equivalent rectangular bandwidth: the bandwidth of a rectangular filter with the same peak response that passes the same total power of a power spectrum that is driven by the same stimulus. This power spectrum is calculated from the fast Fourier transform of the stimulus' impulse response at a specific CF. For our evaluation, we use a 100 μs click stimulus as earlier adopted in (Verhulst et al., 2015; Raufer and Verhulst, 2016). Derived $\mathrm{Q_{ERB}}$-values for different CF will be plotted and compared to both the TL model and a human tuning estimate.

Taking all those different performance measurements in consideration for each trained model, the best performing model for every variable hyperparameter will be selected.

5.2 AECNN for NH

5.2.1 Learning Rate

The overview Table 5.1 displaying the models that were trained with a varying learning rate shows that the number of parameters stays the same and the executing time per epoch is more or less equal. The loss term is slightly lower for the two models with the lowest learning rate. Based on the performance on the input stimuli and the test set, there is not really one model that stands out.

Table 5.1: Overview trained NH AECNN models - Variable learning rate. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/lr/FL)	Param.	LT	Time/Epoch	Click x10^{-4}	1kHz x10^{-4}	4kHz x10^{-4}	Testset x10^{-4}
*PReLU/4 lay./**0.001**/63*	5,641,856	0.0404	1h46m12s	23.74	71.08	23.69	57.10
*PReLU/4 lay./**0.0004**/63*	5,641,856	0.0375	1h48m36s	10.63	61.18	63.79	46.40
*PReLU/4 lay./**0.0001**/63*	5,641,856	0.0376	1h46m40s	9.99	166.85	16.49	49.66

The excitation patterns in Figure 5.5 reveal that none of these models were able to capture the nonlinear level-dependency that is clearly present in the TL model patterns. The excitation patterns of the DNN can be interpreted as a shifted variant of the 70 dBSPL pattern, this could be accounted to the fact that, training the architecture only on 70 dBSPL input fragments, will only give a correct pattern for stimuli of that same level.

The choice was made to proceed with a learning rate of 0.0001 since the flattening of the frequency peak was most resembling to the desired output.

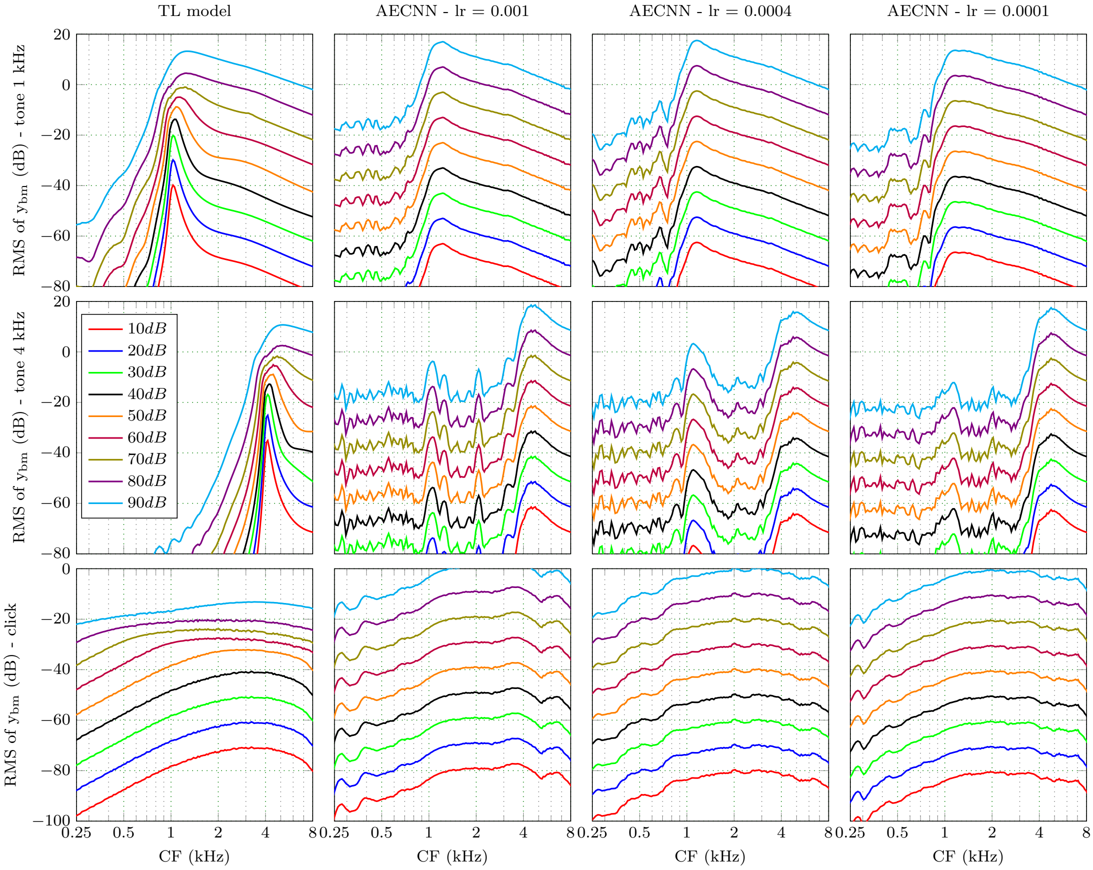


Figure 5.5: Comparison of excitation patterns - Variable learning rate. Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section (corresponding CFs are listed in Appendix A) for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model (left) and AECNN architectures varying in learning rate.

5.2.2 Layer Depth

In theory, changing the layer depth will provide a better performance since the increase of layers, hence tuneable parameters, will give the NN more possibilities to store relevant features that can be found in the input audio signals.

Looking at Table 5.2, a large increase in parameters can indeed be seen for a model that increased its depth to 6 layers. This will induce a small increase in time per epoch (3 minutes) but a reduction in loss term. The MSE values are not significantly different.

Table 5.2: Overview trained NH AECNN models - Variable layer depth. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/lr/FL)	Param.	LT	Time/Epoch	Click x10^{-4}	1kHz x10^{-4}	4kHz x10^{-4}	Testset x10^{-4}
PReLU/4 ***lay.****/0.0001/63*	5,641,856	0.0376	1h46m40s	9.99	166.85	16.49	49.66
*PReLU/****6 lay.****/0.0001/63*	8,836,736	0.0307	1h49m41s	4.41	160.75	15.29	53.95

Figure 5.6 shows two trained models that are fairly similar: the AECNN even after increasing the layer depth, is not able to capture level-dependency. This leaves layer depth as a design choice, set to 6 layers for the next sections based on a lower loss term and better MSE performance.

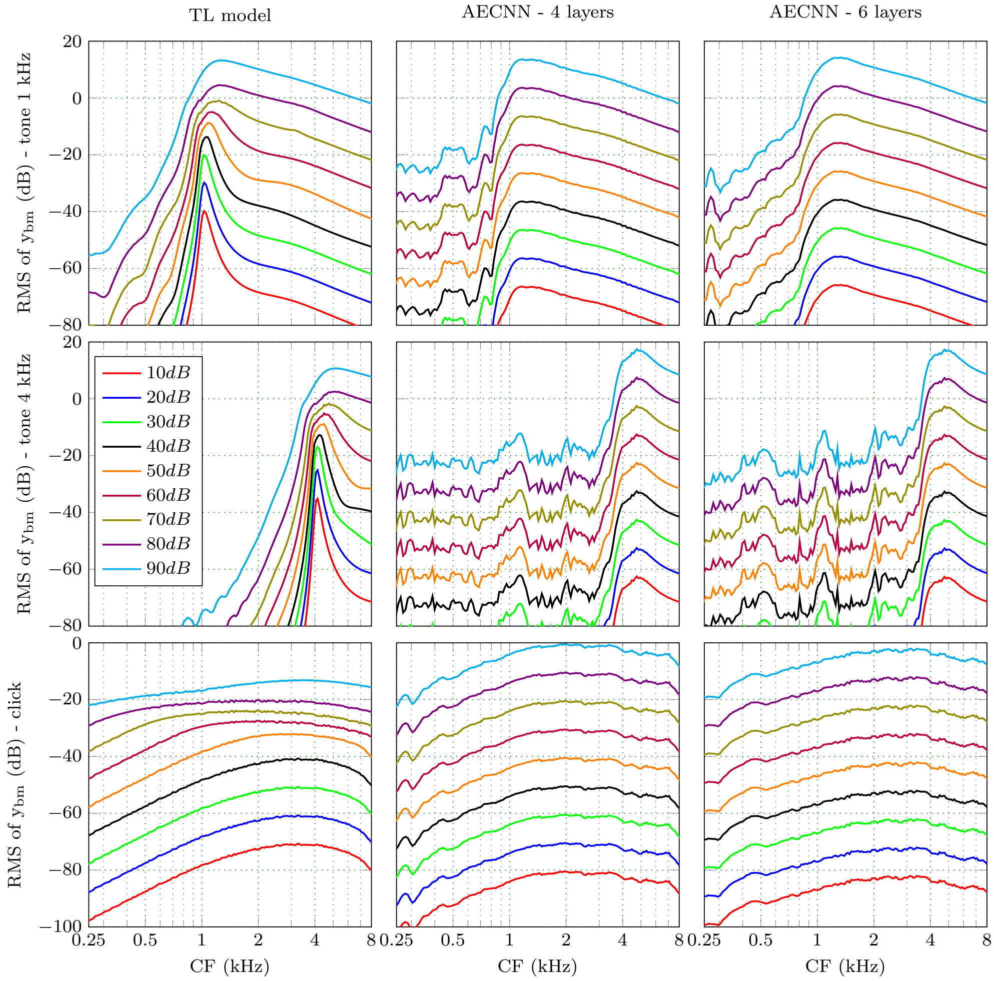


Figure 5.6: Comparison of excitation patterns - Variable layer depth. Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section (corresponding CFs are listed in Appendix A) for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model (left) and AECNN architectures varying in layer depth.

5.2.3 Filter Length

The filter length also accounts for the number of trainable parameters, hence the above effect can also be observed here: a larger number of parameters will decrease the loss term but also increase the execution time per epoch (Table 5.3). Comparing the MSE performance, the model with filter length 127 is the best performing yet. Figure 5.7 depicts these trained models, where the capturing of level-dependent nonlinearities, although an improvement in performance measurement values, is still absent.

Table 5.3: Overview trained NH AECNN models - Variable filter length. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/lr/FL)	Param.	LT	Time/Epoch	Click	1kHz	4kHz	Testset
				x10^{-4}	x10^{-4}	x10^{-4}	x10^{-4}
*PReLU/6 lay./0.0001/**63***	8,836,736	0.0307	1h49m41s	4.41	160.75	15.29	53.95
*PReLU/6 lay./0.0001/**127***	17,380,992	0.0298	1h58m20s	4.45	93.09	13.32	45.43
*PReLU/6 lay./0.0001/**31***	4,564,608	0.0360	1h44m20s	7.07	176.81	30.84	48.29

5.2.4 Nonlinear Activation Functions

The hyperparameter tuning for the PReLU activation function gave an acceptable performance on the 70 dBSPL input stimuli, but wasn't able to capture the nonlinear level-dependency, that is present in the TL reference output. Hence the effect of choosing another (nonlinear) activation function, tanh, was investigated. Table 5.4 indicates that with a moderate increase in time per epoch, the loss term halves and the MSE performance is significantly improved for each type of input signal.

Figure 5.8 shows that the tanh activation function is able to capture the nonlinear level-dependency also present in the TL output, for all three types of input stimuli. This in-

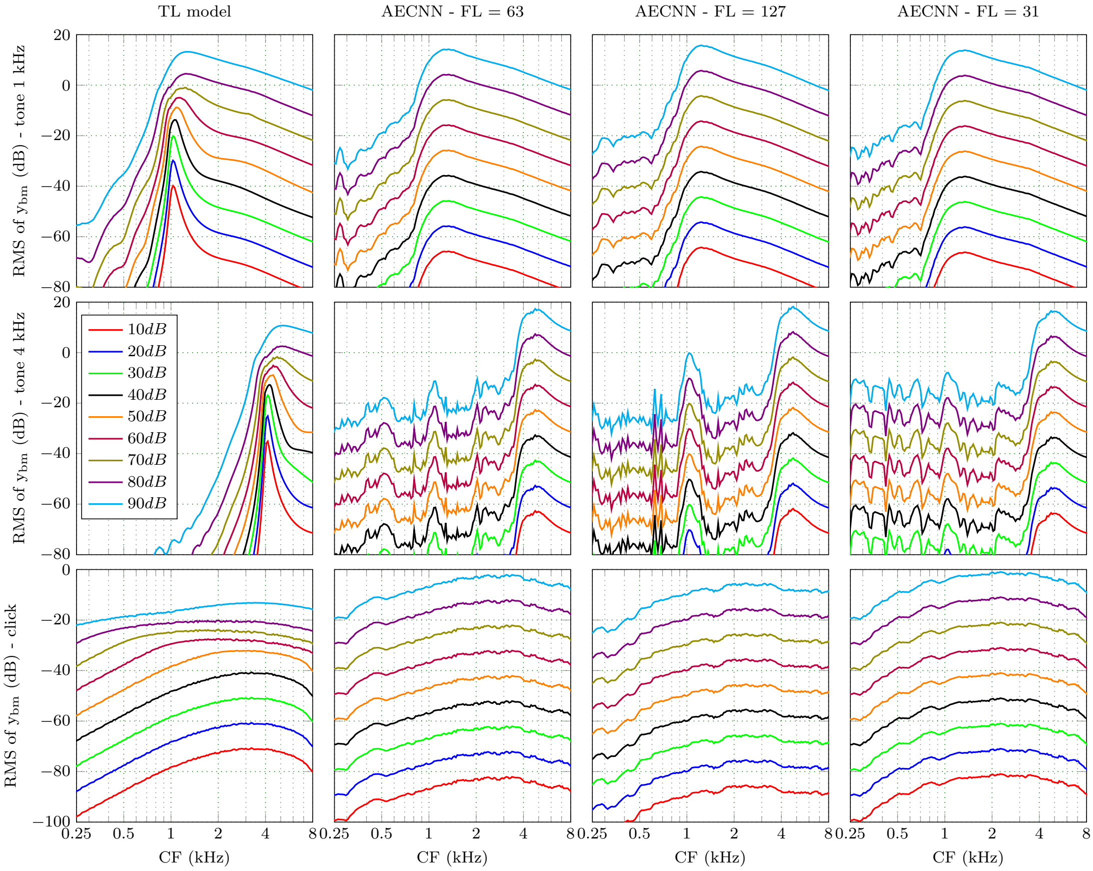


Figure 5.7: Comparison of excitation patterns - Variable filter length. Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section (corresponding CFs are listed in Appendix A) for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model (left) and AECNN architectures varying in filter length.

Table 5.4: Overview trained NH AECNN models - Variable nonlinear activation function. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/lr/FL)	Param.	LT	Time/Epoch	Click	1kHz	4kHz	Testset
				x10^{-4}	x10^{-4}	x10^{-4}	x10^{-4}
***PReLU**/6 lay./0.0001/127*	17,380,992	0.0298	1h58m20s	4.45	93.09	13.32	45.43
***tanh**/6 lay./0.0001/127*	16,955,008	0.0148	1h59m05s	1.21	4.92	6.05	20.06

dicates that an AECNN, with a tanh nonlinear activation function, is able to learn how level-dependent compression is accounted for in the cochlea, and this trained only on input audio data of one particular sound level (70 dBSPL).

5.2.5 Layer Depth - Revisited

Recalling that layer depth was a design choice, the layer depth of the DNN is readdressed for a tanh activation function. Table 5.5 depicts this variation, from which can be concluded that the performance of the model with 6 layers is fairly similar to the one with 8 layers, having only a small increase in loss term, but a decrease of 7,000,000 trainable parameters. Figure 5.9 doesn't display a model which stands out, leaving this hyperparameter a design choice.

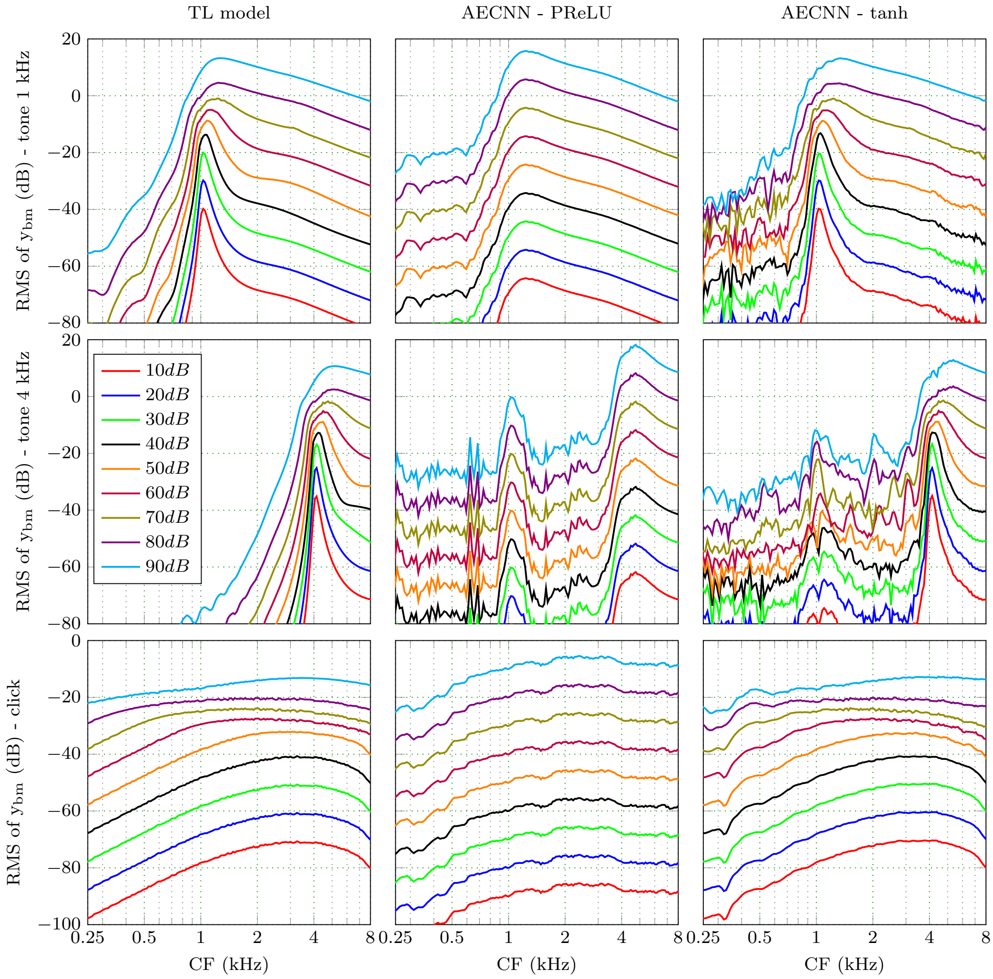


Figure 5.8: Comparison of excitation patterns - Variable nonlinear activation function. Cochlear excitation patterns calculated as the RMS value of the BM displacement ($y_{\rm BM}$) per cochlear section (corresponding CFs are listed in Appendix A) for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model (left) and AECNN architectures varying in nonlinear activation function.

Table 5.5: Overview trained NH AECNN models - Variable layer depth. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/lr/FL)	Param.	LT	Time/Epoch	Click x10^{-4}	1kHz x10^{-4}	4kHz x10^{-4}	Testset x10^{-4}
*tanh/**6 lay.**/0.0001/127*	16,955,008	0.0148	1h59m05s	1.21	4.92	6.05	20.06
*tanh/**4 lay.**/0.0001/127*	10,712,704	0.0185	1h54m42s	1.63	3.95	8.63	24.04
*tanh/**8 lay.**/0.0001/127*	23,197,312	0.0140	2h01m40s	1.13	4.29	8.64	15.19

In the scope of this thesis, since the AECNN implementation in possible real-time applications is mainly a matter of computational complexity, the choice is more drawn towards the less complex 6 layer model.

5.2.6 Summary NH

An overview of all trained normal hearing AECNN models, with their performance, is given in Appendix C.

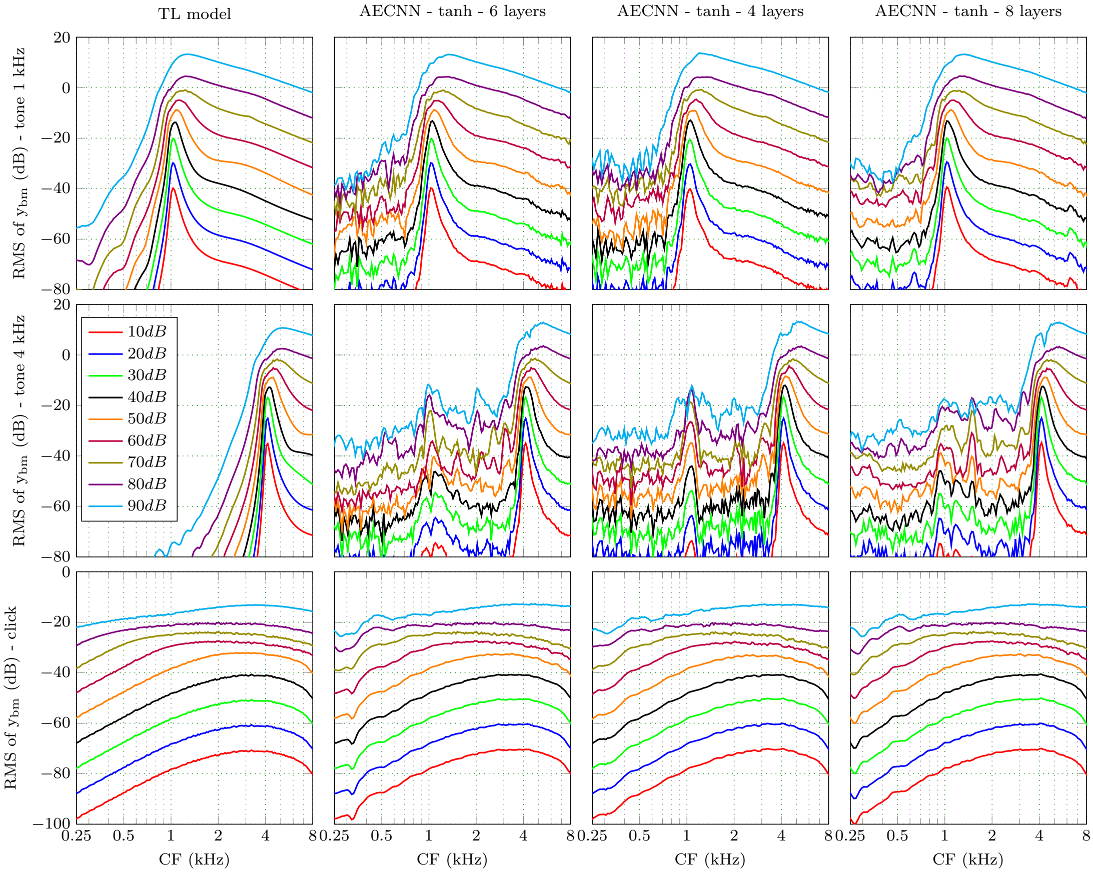


Figure 5.9: Comparison of excitation patterns - Variable layer depth. Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section (corresponding CFs are listed in Appendix A) for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model (left) and AECNN architectures varying in layer depth.

5.2.7 $\mathrm{Q_{ERB}}$

Looking at the $\mathrm{Q_{ERB}}$ plots for the best performing PReLU and tanh-based models, depicted in respectively Figure 5.10 and Figure 5.11, the same interpretation can be made: whereas the PReLU-based architecture is approximating the TL $\mathrm{Q_{ERB}}$ reference-data for a 70 dB intensity click well, it categorizes, due to the lack of level-dependency, the 40 dB click as a 70 dB click as well. The tanh-based architecture, on the other hand, shows a correct distinction between the different intensities and is doing an excellent job in resembling the TL $\mathrm{Q_{ERB}}$ values, which are on their part very well resembling to literature values of human $\mathrm{Q_{ERB}}$ values (Shera et al., 2010).

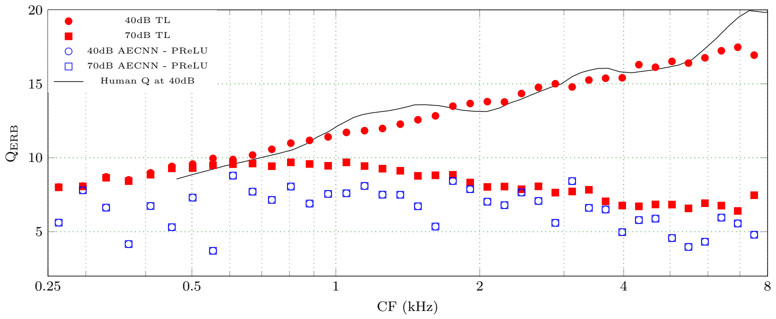


Figure 5.10: $\mathrm{Q_{ERB}}$ values for trained PReLU AECNN. $\mathrm{Q_{ERB}}$ values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the TL model (red), trained PReLU AECNN model (blue) and a literature human $\mathrm{Q_{ERB}}$ estimate (Shera et al., 2010).

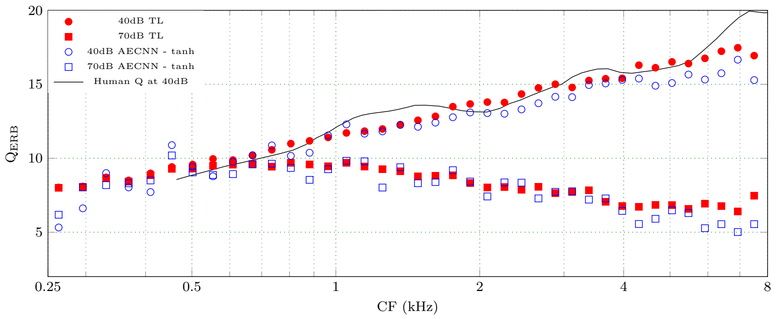


Figure 5.11: $\mathrm{Q_{ERB}}$ values for trained tanh AECNN. $\mathrm{Q_{ERB}}$ values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the TL model (red), trained tanh AECNN model (blue) and a literature human $\mathrm{Q_{ERB}}$ estimate (Shera et al., 2010).

5.2.8 Performance Best Model on Basic Auditory Stimuli

To finalize the NH part of the research question, the cochlear dispersion for the basic auditory input stimuli (click, pure tone 1 kHz and pure tone 4 kHz) and for the example of the test set input, are displayed in Figures 5.12 and 5.13, for the best performing NH AECNN model (tanh - 6 layers - 0.0001 - 127).

5.2.9 Side Note: Context

Both Figure 5.12 (top right: difference in performance for a click stimulus) and Figure 5.13 (bottom: difference in performance on test set example) depict a suboptimal behaviour for (roughly) the first 20 ms of the AECNN output response. A possible explanation can be found in the applied processing pathway of Chapter 4 (Section 4.1.2), where slicing of both the TIMIT dataset and the TL reference model output to segments of 2048 samples was done. However, to receive the reference TL model output, the full length of a training example was presented at the input, thus including the context (the samples that are proceeding and succeeding) of each sample. This context however is lost for the first samples when slicing the input data. It followed that the AECNN is trained on examples (the reference output) that contain information linked to the proceeding context of the cropped audio sample, that the AECNN is not able to see. The result of this omitting of context should only be visible in the region where this context has an influence, and following the visual inspection of the plots, this region can be set at 20 ms. Coupling back to the Q_{ERB} plots depicted on Figures 5.10 and 5.11, it is possible that this slightly poorer resemblance for the lower frequency values can also be explained by this.

This incorporation of context for the data used in the AECNN training phase can be seen as an extension of this project and is not further discussed here.

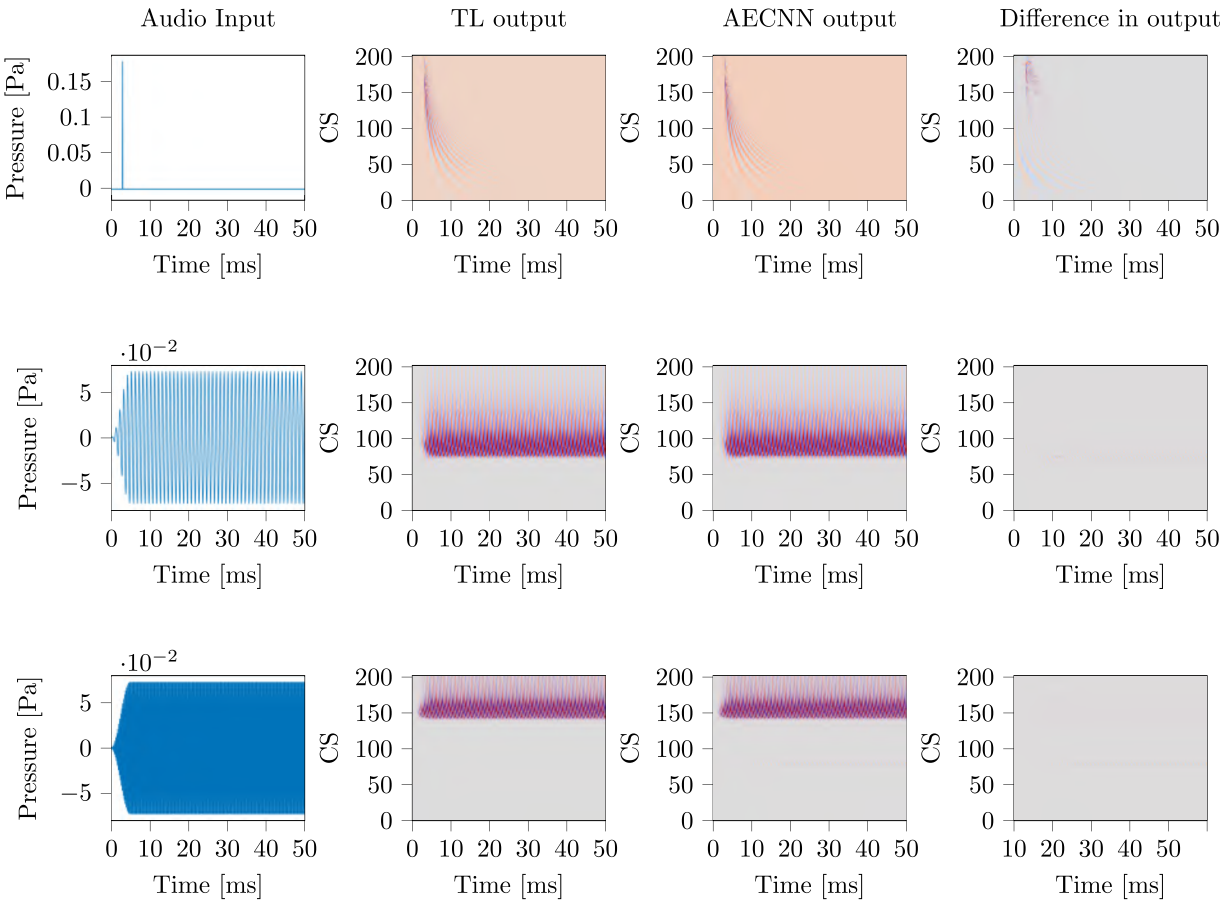


Figure 5.12: Performance best AECNN model on basic auditory input stimuli. (Left column) Input pressure [Pa] in the time domain [ms] for the three different stimuli (click, pure tone 1 kHz and pure tone 4 kHz). (Middle columns) Output cochlear dispersion of the TL and trained NH AECNN model: BM displacements for the selected 201 CS (see Appendix A for the corresponding CF), for their respective input signal. (Right column) Difference between the two previous depicted outputs.

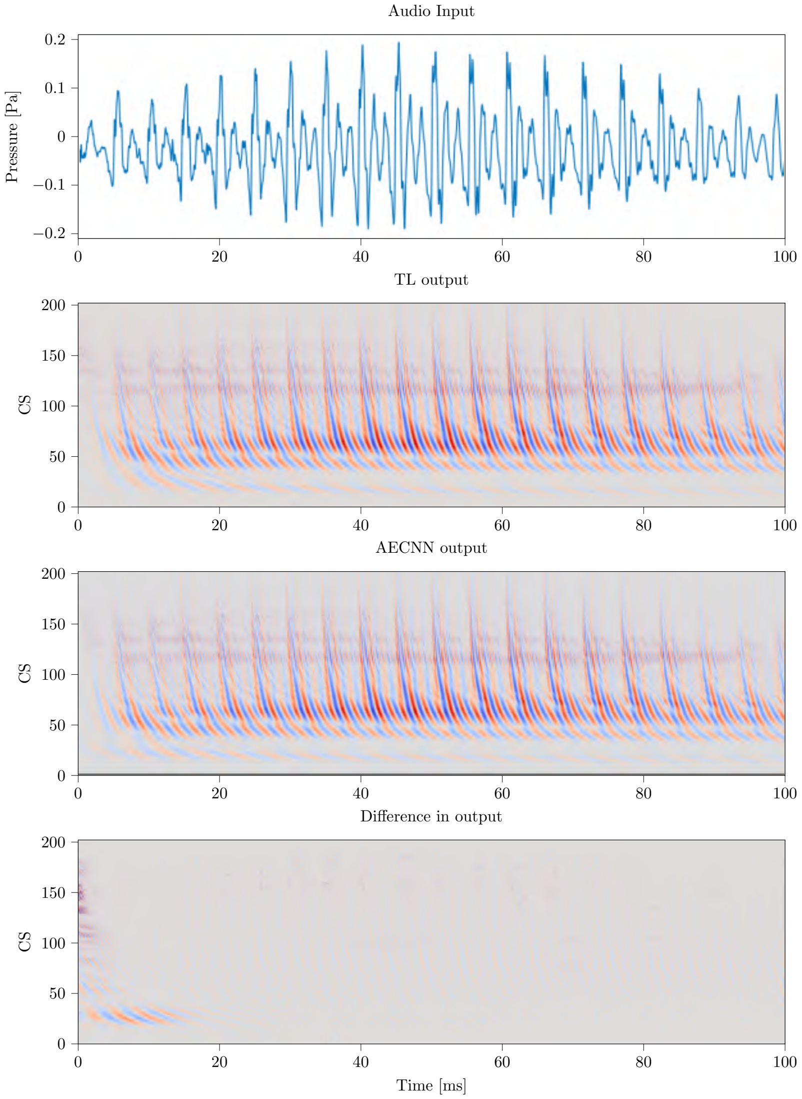


Figure 5.13: Performance best AECNN model on test set input. (Top) Input pressure [Pa] in the time domain [ms] for a speech fragment part of the test set. (Middle) Output response of the TL and trained AECNN model: BM displacements for the selected 201 CS (see Appendix A for the corresponding CF). (Bottom) Difference between the two previous depicted models.

5.3 AECNN for HI

The second goal of this dissertation consists of adapting, via transfer learning, the best performing NH AECNN towards a network that can be used to replace the HI version of the reference TL model. The starting models for this task will be both the 6 and 8 layer variant of the tanh-based NH architecture.

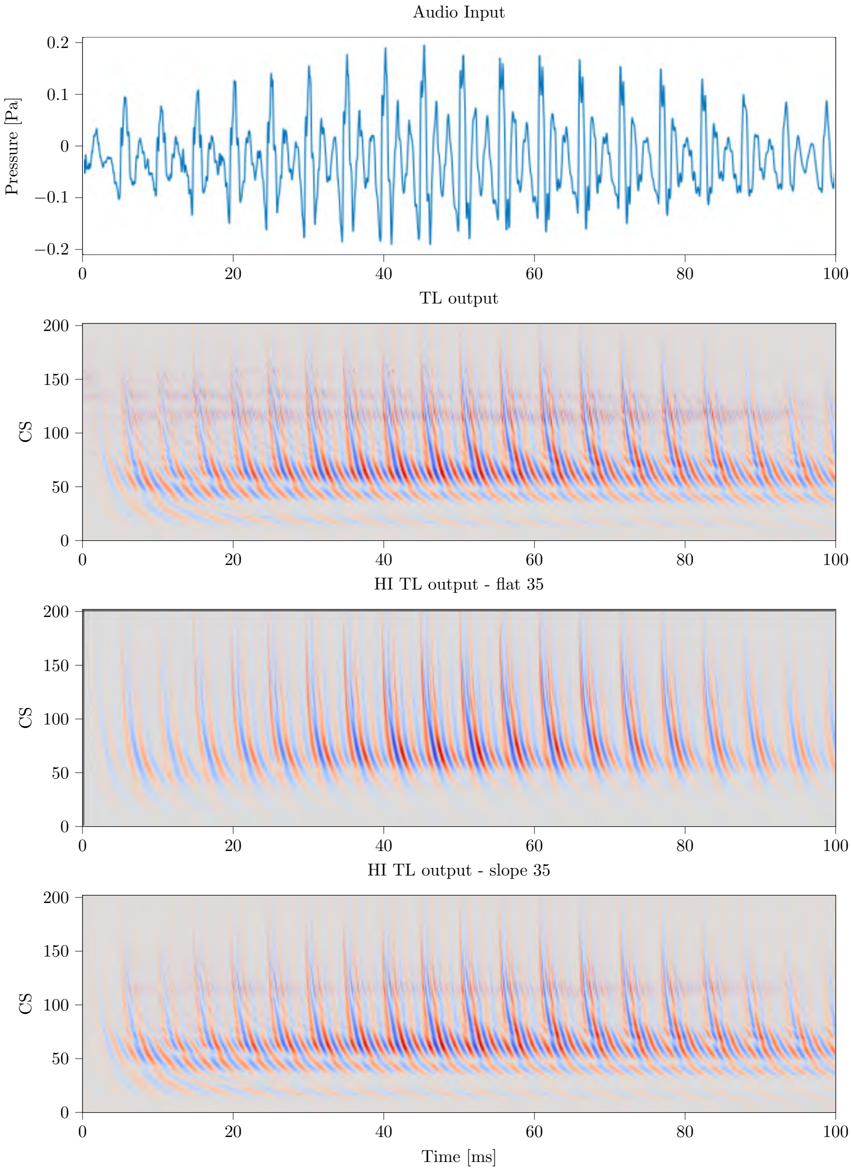


Figure 5.14: Performance NH and HI TL models on test set input. (Top) Input pressure [Pa] in the time domain [ms] for a speech fragment part of the test set. (Bottom) Output responses for the NH TL model, the HI TL model with flat 35 profile and the HI TL model with slope 35 profile: BM displacements for the selected 201 CS (see Appendix A for the corresponding CF).

As mentioned in previous the chapters, the focus will be on the most severe HL profiles that can be implemented in the reference TL model: the slope 25/35 and the flat 25/35 profiles. The resulting cochlear output for a TL model that is made hearing-impaired for the flat 35 and slope 35 profile is shown in Figure 5.14.

5.3.1 Slope HL Profiles

Slope 35

The variable hyperparameter in this transfer learning approach, as mentioned in Chapter 4, is the number of training utterances used. The same type of overview table that was used in the previous sections is set up here to discuss this variational parameter. An attribute that immediately stands out, when comparing Table 5.6 with the previous tables, is a really small time per epoch (order of minutes vs order of hours). This can be accounted to the fact that the starting NH architecture is already trained to include certain specificities of hearing that can be found in both NH and HI profiles, hence those features don't need to be learnt any more by the HI variant of the AECNN thus reducing the time per epoch.

Table 5.6: Overview trained HI AECNN models - slope 35. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/Utt.)	Param.	LT	Time/Epoch	Click	1kHz	4kHz	Testset
				x10^{-4}	x10^{-4}	x10^{-4}	x10^{-4}
tanh/6 lay./50	16,955,008	0.0072	32s	0.35	2.66	0.28	12.28
tanh/6 lay./100	16,955,008	0.0071	1m03s	0.33	2.57	0.20	11.90
tanh/8 lay./50	23,197,312	0.0057	36s	0.40	3.63	0.53	8.12
tanh/8 lay./100	23,197,312	0.0058	1m10s	0.33	2.19	0.53	7.97

Similar to the NH situation, the table indicates that the overall performance of the 6 and 8 layer model is fairly similar, which also can be concluded from Figure 5.15. All the trained models are able to capture the nonlinear level-dependency present in the HI TL excitation patterns. The noisy response for the lower frequencies in the pure tone outputs can be disregarded since these are dB values located 40-50 dB below the peak value.

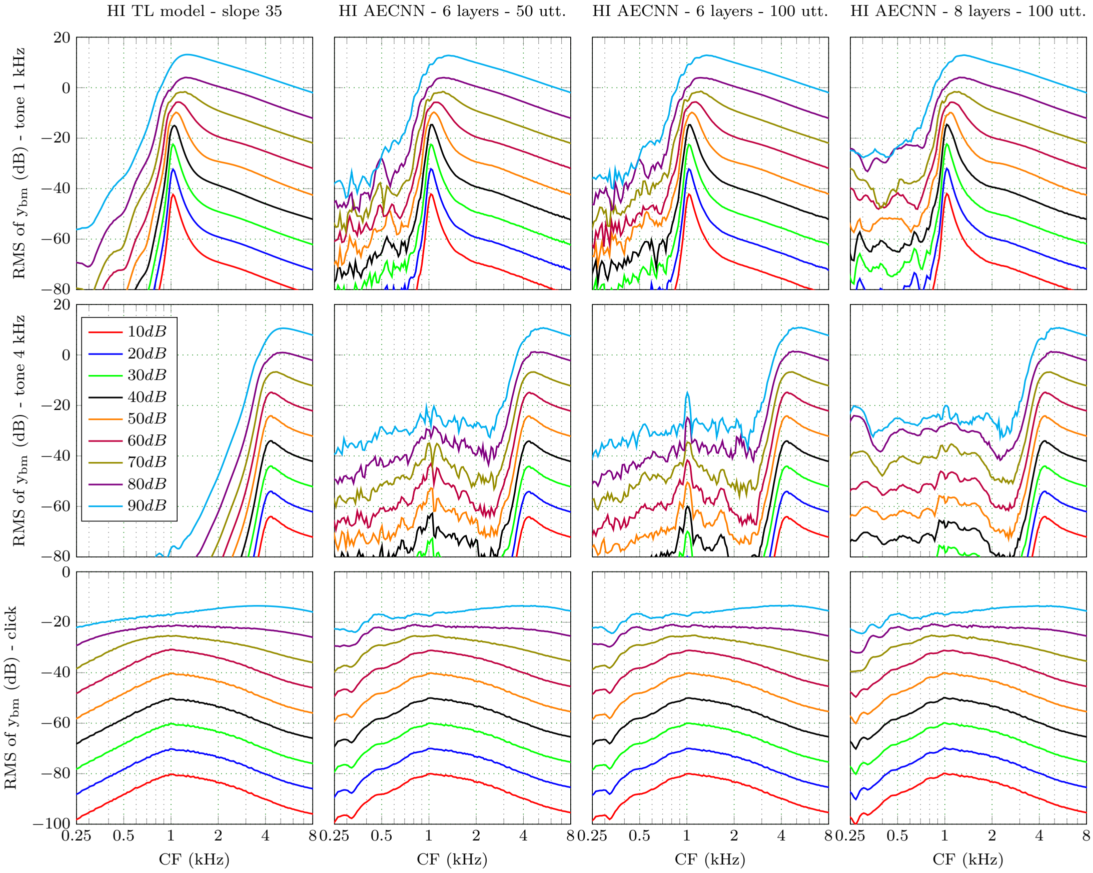


Figure 5.15: Comparison of excitation patterns - slope 35 - Variable training utterances and layer depth. Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section (corresponding CFs are listed in Appendix A) for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model that is made HI with a slope 35 cochlear gain loss profile (left) and trained HI AECNN architectures varying in layer depth and number of used training utterances.

Looking at the corresponding Q_{ERB} plot for the slope 35 profile, Figure 5.16, it immediately shows that the overall tuning curve for a hearing-impaired person is significantly different compared to the NH variant that was introduced above. The depicted Q_{ERB} values for the trained HI model (the 6 layer/50 utt. variant was chosen for its low training time and high performance), show good resemblance with the NH TL data. Since this resemblance can already be obtained with only 50 utterances and a total transfer learning time of around 12 minutes, it proves that transfer learning gives a viable option in the development of HI variants of NH AECNN models.

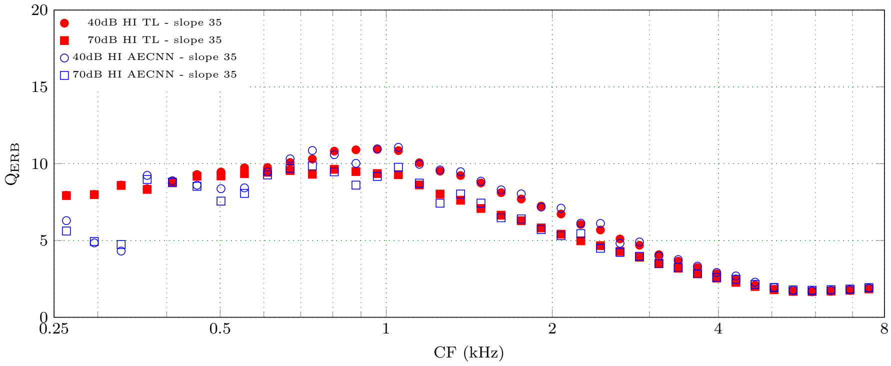


Figure 5.16: Q_{ERB} values for trained HI AECNN - slope 35. Q_{ERB} values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the HI TL model (red) and trained HI AECNN model (blue) for a slope 35 HL profile. Refer to Subsection 5.2.9 for an explanation for the suboptimal low frequency performance.

To round up the performance measurements on the slope 35 profile, the performance of the HI AECNN on the test set example is depicted in Figure 5.20.

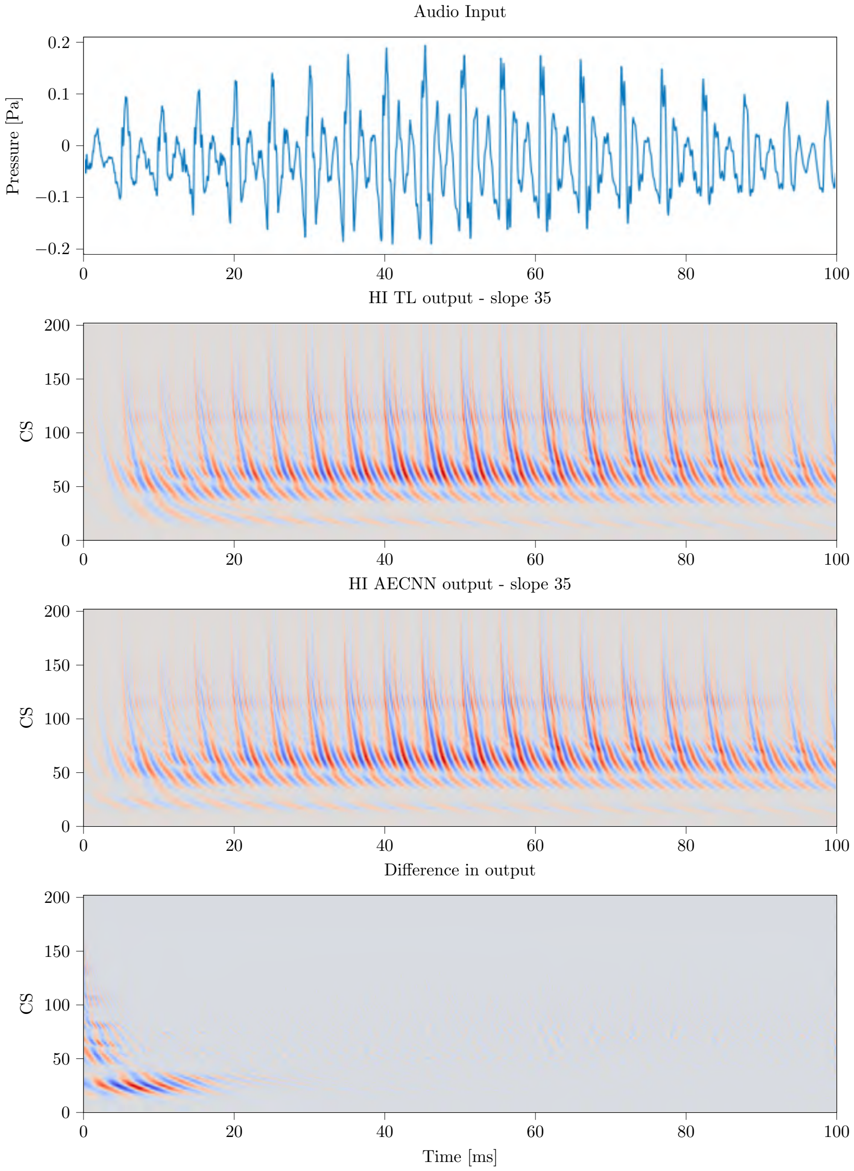


Figure 5.17: Performance HI AECNN model on test set input - slope 35 HL profile. (Top) Input pressure [Pa] in the time domain [ms] for a speech fragment part of the test set. (Middle) Output response of the HI TL model and the trained HI AECNN model: BM displacements for the selected 201 CS (see Appendix A for the corresponding CF), both with a slope 35 cochlear gain loss profile. (Bottom) Difference between the two previous depicted models. Refer to Subsection 5.2.9 for an explanation for the suboptimal performance of the first 20 ms.

Slope 25

The same procedure was done for the (less severe) slope 25 HL profile, of which the overview table is given in Table 5.7. The accompanying excitation patterns and Q_{ERB} plots are given in Appendix D.

Table 5.7: Overview trained HI AECNN models - slope 25. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/Utt.)	Param.	LT	Time/Epoch	Click	1kHz	4kHz	Testset
				x10^{-3}	x10^{-2}	x10^{-1}	x10^{-3}
tanh/6 lay./100	16,955,008	0.0084	1m05s	1.91	6.50	1.76	1.40
tanh/8 lay./100	23,197,312	0.0071	1m10s	1.91	6.47	1.76	0.95

Based on Table 5.7 it can be concluded that an AECNN variant with the HL profile of slope 25 can also be rendered. Extracting this conclusion leads to the following statement: it is possible to go towards a NN representation of any individualized hearing profile with a sloping cochlear gain loss.

5.3.2 Flat HL Profiles

Flat 35

Table 5.8 and Figures 5.18, 5.19 and 5.20 depict the same procedure that was followed for the slope 35 variant, but now for the flat 35 profile. Although a different type of HL profile, the overall HI AECNN performance can also be classified as sufficient.

Table 5.8: Overview trained HI AECNN models - flat 35. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/Utt.)	Param.	LT	Time/Epoch	Click x10^{-4}	1kHz x10^{-4}	4kHz x10^{-4}	Testset x10^{-4}
tanh/6 lay./50	16,955,008	0.0014	31s	0.09	0.06	0.09	3.56
tanh/6 lay./100	16,955,008	0.0013	1m04s	0.10	0.01	0.02	3.34
tanh/8 lay./50	23,197,312	0.0011	35s	0.10	0.05	0.11	1.98
tanh/8 lay./100	23,197,312	0.0010	1m11s	0.08	0.02	0.03	1.74

Flat 25

After the inspection of overview Table 5.9 (and based on the plots depicted in Appendix D), the statement made in the previous section can be adapted to: *It is possible to go towards a NN representation of any individualized hearing profile characterized by cochlear gain loss due to OHC deficits, be it a sloping profile or a flat, constant gain reduction.*

Table 5.9: Overview trained HI AECNN models - flat 25. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/Utt.)	Param.	LT	Time/Epoch	Click x10^{-3}	1kHz x10^{-2}	4kHz x10^{-1}	Testset x10^{-3}
tanh/6 lay./100	16,955,008	0.0022	1m04s	0.83	3.71	1.56	0.47
tanh/8 lay./100	23,197,312	0.0019	1m11s	0.82	3.34	1.53	0.30

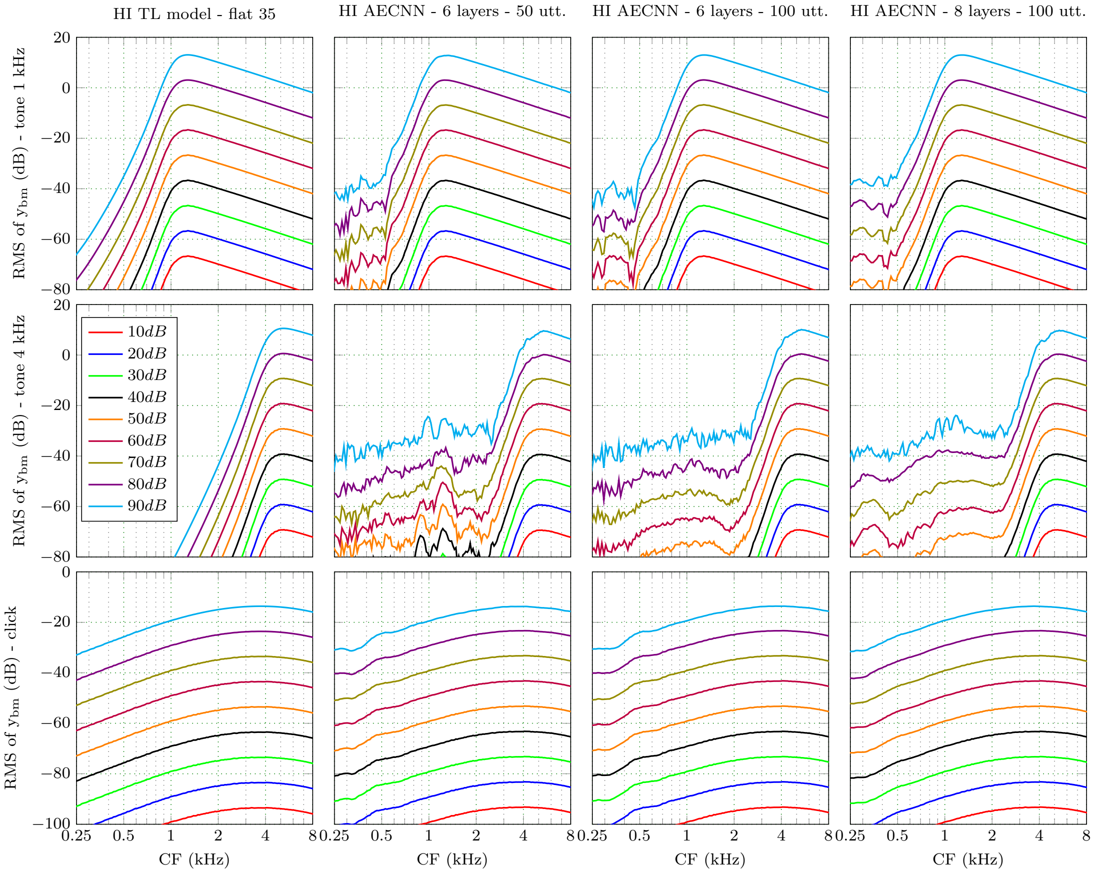


Figure 5.18: Comparison of excitation patterns - flat 35 - Variable training utterances and layer depth. Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section (corresponding CFs are listed in Appendix A) for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model that is made HI with a flat 35 cochlear gain loss profile (left) and trained HI AECNN architectures varying in layer depth and number of used training utterances.

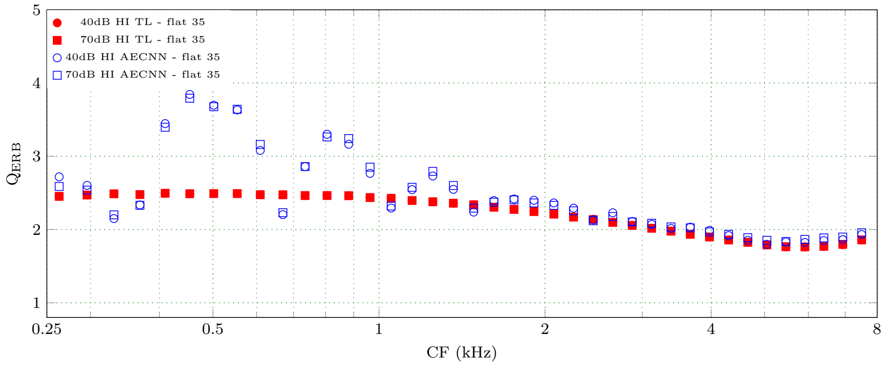


Figure 5.19: $\mathbf{Q_{ERB}}$ values for trained HI AECNN - flat 35. $\mathrm{Q_{ERB}}$ values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the HI TL model (red) and trained HI AECNN model (blue) for a flat 35 HL profile. Refer to Subsection 5.2.9 for an explanation for the suboptimal low frequency performance.

5.3.3 Fixed Layers

This last investigation is influenced by machine vision since it is shown that, once a DNN for image recognition is trained, certain hidden layers are responsible for the detection of specific structures in the images (Liang and Hu, 2015). Suppose this would be true for machine hearing: that certain layers of a trained AECNN are linked to specific hearing characteristics and others to incorporate hearing-impaired profiles in the DNN. This would mean that the total training time can be reduced even more, since only a certain number of parameters should be updated in the process of transfer learning to include hearing-impairment.

To verify this, the performance of the 6 layer/50 utt. model for a flat 35 HL profile, trained in Subsection 5.3.2, is compared to 4 other models. In each of these models only a part of the hidden layers is made trainable. The result is given in Table 5.10, from which it can be concluded that the model, of which only the weights of the last layer were made trainable, is resembling quite good to the reference model. The outcome of this fixation of layers is a reduction of 10,420,096 trainable parameters and a time gain of 12 seconds per epoch, both reducing computational cost.

To confirm the assumption that it is indeed the last convolutional layer of the AECNN architecture that is crucial in addressing hearing-impairment, the $\mathrm{Q_{ERB}}$ values of the model,

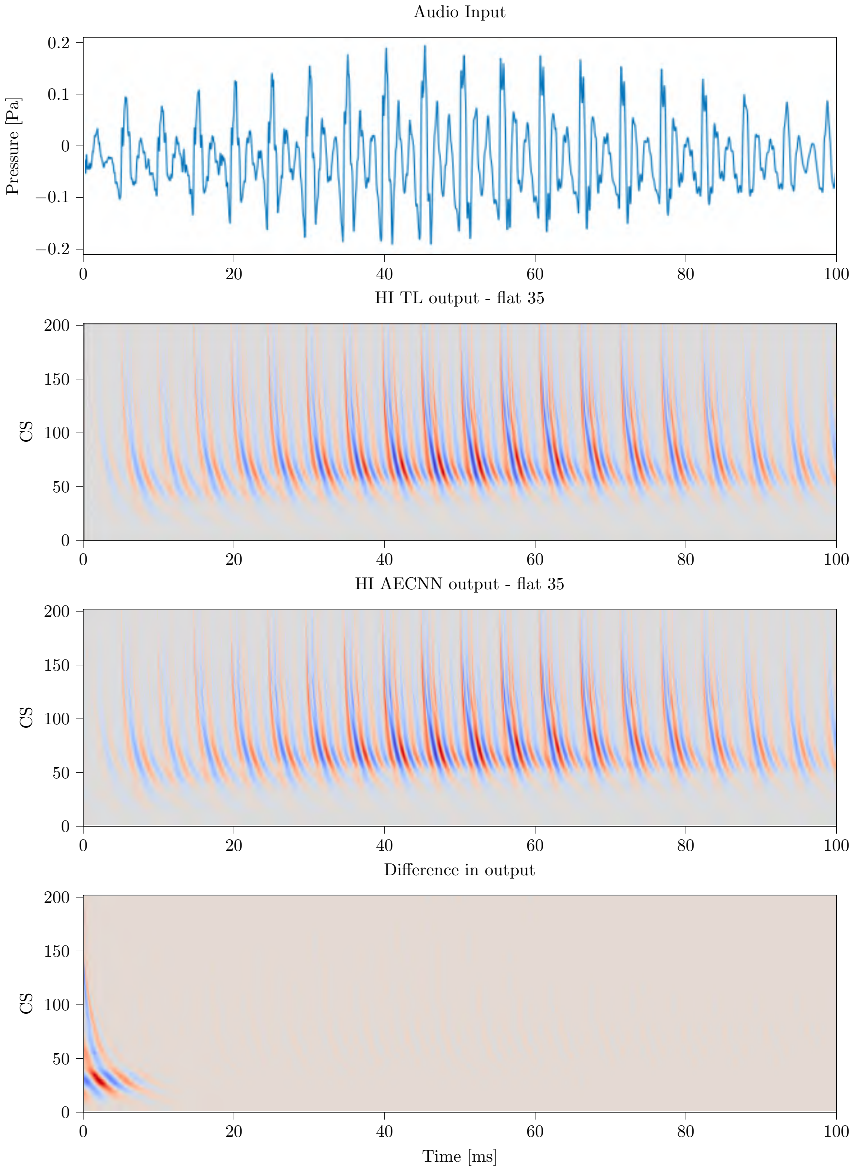


Figure 5.20: Performance HI AECNN model on test set input - flat 35 HL profile. (Top) Input pressure [Pa] in the time domain [ms] for a speech fragment part of the test set. (Middle) Output response of the HI TL model and the trained HI AECNN model: BM displacements for the selected 201 CS (see Appendix A for the corresponding CF), both with a flat 35 cochlear gain loss profile. (Bottom) Difference between the two previous depicted models. Refer to Subsection 5.2.9 for an explanation for the suboptimal performance of the first 20 ms

Table 5.10: Overview trained HI AECNN models - trainable layers - flat 35. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model	Param.	LT	Time/Epoch	Click	1kHz	4kHz	Testset
				x10^{-4}	x10^{-4}	x10^{-4}	x10^{-4}
tanh/6 lay./50	16,955,008	0.0014	31s	0.09	0.06	0.09	3.56
Only first trainable	16,256	0.0185	22s	0.60	13.52	41.78	25.93
Only last trainable	6,534,912	0.0021	19s	0.08	0.17	0.02	3.84
First 3 trainable	4,177,792	0.0049	23s	0.22	0.95	2.17	5.74
Last 3 trainable	12,777,216	0.0018	27s	0.09	0.02	0.04	3.49

that was trained via transfer learning while only the weights of the last layer were adapted, are depicted in Figure 5.21. This shows that there is strong belief that the hearing-impaired character of this CNN is indeed located in the final layer, resulting in an acceptable reduction of the number of trainable parameters and accompanying computational complexity, in the case of a flat 35 HL profile.

Table 5.11 and Figure 5.22 are used in validating if the accountability of the slope 35 hearing loss profile can also be found in the last layer. The results indicates that a strong case can be made that all HI profiles are accounted for in the last layer of this AECNN architecture.

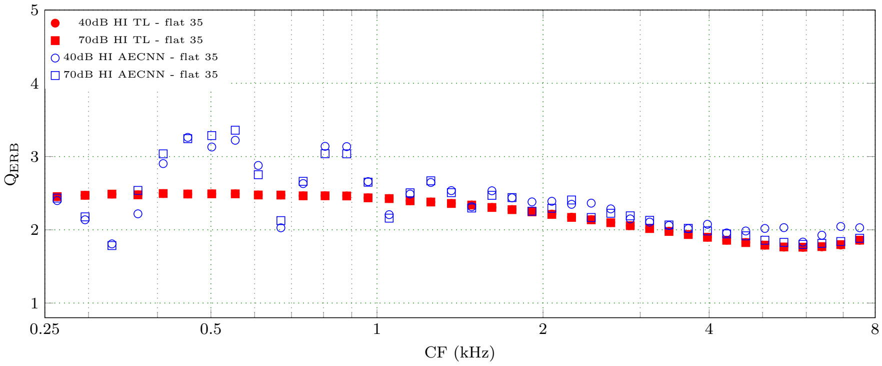


Figure 5.21: Q_{ERB} values for trained HI AECNN - flat 35 - fixed layers. Q_{ERB} values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the HI TL model (red) and trained HI AECNN model (blue) for a flat 35 HL profile. The AECNN model was trained with fixed weights in all but the last layer. Refer to Subsection 5.2.9 for an explanation for the suboptimal low frequency performance.

Table 5.11: Overview trained HI AECNN models - trainable layers - slope 35. Number of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model	Param.	LT	Time/Epoch	Click	1kHz	4kHz	Testset
				x10^{-4}	x10^{-4}	x10^{-4}	x10^{-4}
tanh/6 lay./50	16,955,008	0.0072	32s	0.35	2.66	0.28	12.28
Only last trainable	6,534,912	0.0083	19s	0.39	2.89	1.06	12.95

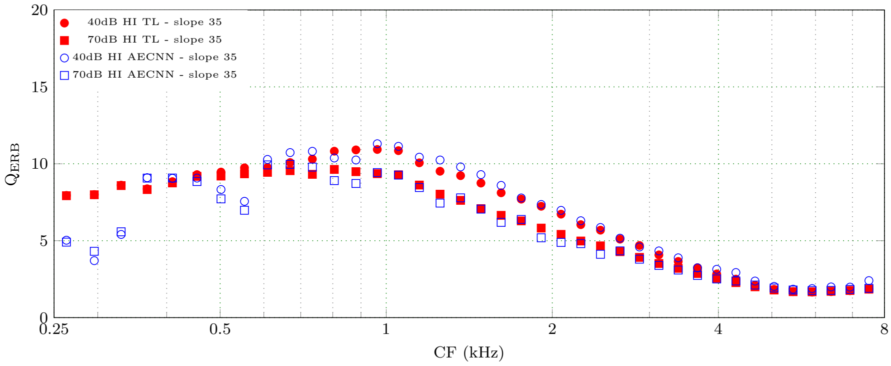


Figure 5.22: Q_{ERB} values for trained HI AECNN - slope 35 - fixed layers. Q_{ERB} values computed for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the HI TL model (red) and trained HI AECNN model (blue) for a slope 35 HL profile. The AECNN model was trained with fixed weights in all but the last layer. Refer to Subsection 5.2.9 for an explanation for the suboptimal low frequency performance.

Chapter 6

Conclusion

In this project, a deep neural network (DNN) was modeled to approximate a state-of-the-art, biophysically realistic model of the human cochlea, based on a cascaded nonlinear transmission-line (TL) model. The TL nonlinearity, which accounts for several hearing aspects (e.g., longitudinal coupling, level-dependent tuning, frequency selectivity), introduces a high computational cost, hence limits the implementation of these correct models in applications compared to the widely used perceptual models (e.g., gammatone, MFCC), which are on their part omitting key features associated with hearing, in order to become faster. Removing this ever-existing compromise between biophysically correctness and computational complexity was the first goal of this master's thesis. A DNN approach, which forms a hybrid of convolutional neural networks and computational auditory modeling, yields a real-time solution of the cochlear processing stage by means of replacing one, slow, nonlinear model, with another, fast, nonlinear DNN architecture.

The reference TL model, on which the DNN architecture was based, possesses also the ability to include hearing-impaired (HI) profiles based on outer hair cell cochlear gain loss in the modelling stages. The second part of this project focussed on the incorporation of this HI behaviour, obtained via adaptation of the DNN found in the first part of this project.

The combination of these two tasks in a single DNN architecture, capable of both accounting for NH as HI profiles in a real-time matter, is something that has never been done before.

The stages in this project consisted of a modeling phase, where a DNN architecture was formed based on the selection of various hyperparameters, followed by a training phase where the weight adaptation was done by exposing the network to input-output combinations of audio (speech) fragments linked to their resembling TL model output. After a certain numbers of training cycles, the best performing model was stored and could be evaluated for its performance on basic auditory stimuli inputs and unseen test data.

Results showed that, with the correct parameter choices, the desired nonlinear features of the cochlea: longitudinal coupling, frequency selective tuning and level-dependent compression, could all be found in the real-time operating DNN. Which was then used as a starting point for a transfer learning procedure, that allowed to, starting from the NH DNN, go towards a HI DNN by training on input-output combinations that resembled specific hearing loss profiles. The process of transfer learning allowed to successfully train a HI version of the DNN with only 50 additional training utterances and with a learning phase that only lasted for 7 minutes.

This project succeeded in its two research goals and proved that this approach can be considered for any transmission-line model that incorporates nonlinearities (e.g., brain networks, electronics applications), but also, by using a DNN approach, has the ability to be applied into low-power implementations (e.g., ASR, hearing-aids, robotics).

Looking at future work in the scope of this thesis, the role of the context of a speech fragment, introduced in Section 5.2.9, is something definitely worth investigating in follow-up projects. Also the extension of the DNN beyond the cochlea can be addressed: to include 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 in auditory models (e.g., synaptopathy). Afterwards, a hearing loss database could be formed, consisting of the collection of individualized hearing profiles, all based on deep neural nets.

Bibliography

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283.

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

Allen, J. (2001). Nonlinear cochlear signal processing. In *Physiology of the Ear, Second Edition*, pages 393–442. Singular Thompson.

Altoe, A., Pulkki, V., and Verhulst, S. (2014). Transmission line cochlear models: improved accuracy and efficiency. *The Journal of the Acoustical Society of America*, 136(4):EL302–EL308.

Arden Dertat (2017a). Applied deep learning - part 3: Autoencoders. `https://towardsdatascience.com/applied-deep-learning-part-3-autoencoders-1c083af4d798/`. Last checked on 2019-05-21.

Arden Dertat (2017b). Applied deep learning - part 4: Convolutional neural networks. `https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2`. Last checked on 2019-05-21.

Baby, D. and Verhulst, S. (2018a). Biophysically-inspired features improve the generalizability of neural network-based speech enhancement systems. In *Interspeech*. ISCA.

Baby, D. and Verhulst, S. (2018b). End-to-end raw speech waveform enhancement using conditional generative adversial networks. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE.

Bourlard, H. and Kamp, Y. (1988). Auto-association by multilayer perceptrons and singular value decomposition. *Biological cybernetics*, 59(4-5):291–294.

Cevora, G. (2019). The relationship between biological and artificial intelligence. *arXiv preprint arXiv:1905.00547*.

Chollet, F. et al. (2018). Keras: The python deep learning library. *Astrophysics Source Code Library*.

Cireşan, D., Meier, U., and Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. *arXiv preprint arXiv:1202.2745*.

Daniel Rothmann (2017). The promise of ai in audio processing. `https://towardsdatascience.com/the-promise-of-ai-in-audio-processing-a7e4996eb2ca`. Last checked on 2019-05-20.

Daniel Rothmann (2018). Human-like machine hearing with ai. `https://towardsdatascience.com/human-like-machine-hearing-with-ai-1-3-a5713af6e2f8`. Last checked on 2019-05-10.

Davis, S. and Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE transactions on acoustics, speech, and signal processing*, 28(4):357–366.

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

Donahue, C., Li, B., and Prabhavalkar, R. (2018). Exploring speech enhancement with generative adversarial networks for robust speech recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5024–5028. IEEE.

Duifhuis, H. (2004). Comment on an approximate transfer function for the dual-resonance nonlinear filter model of auditory frequency selectivity" [j. acoust. soc. am. 114, 2112-21171 (1)]. *Journal of the Acoustical Society of America*, 115(5):1889–1890.

Eguiluz, V. M., Ospeck, M., Choe, Y., Hudspeth, A., and Magnasco, M. O. (2000). Essential nonlinearities in hearing. *Physical review letters*, 84(22):5232.

Elliott, S. J. and Shera, C. A. (2012). The cochlea as a smart structure. *Smart Materials and Structures*, 21(6):064001.

Fletcher, H. (1940). Auditory patterns. *Reviews of modern physics*, 12(1):47.

Frost, J. (1838). *The Class Book of Nature: Comprising Lessons on the Universe, the Three Kingdoms of Nature, and the Form and Structure of the Human Body.* Hartford: Belknap and Hamersley.

Fukushima, K. (1988). Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural networks*, 1(2):119–130.

Ganin, Y. and Lempitsky, V. (2014). Unsupervised domain adaptation by backpropagation. *arXiv preprint arXiv:1409.7495*.

Garofolo, J. S. (1993). Timit acoustic phonetic continuous speech corpus. *Linguistic Data Consortium, 1993*.

Gold, T. (1948). Hearing. ii. the physical basis of the action of the cochlea. *Proceedings of the Royal Society of London. Series B-Biological Sciences*, 135(881):492–498.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.

Green, D. M. (1958). Detection of multiple component signals in noise. *The Journal of the Acoustical Society of America*, 30(10):904–911.

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

Haytham Fayek (2016). Speech processing for machine learning: Filter banks, mel-frequency cepstral coefficients (mfccs) and what's inbetween. `https://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html`. Last checked on 2019-05-26.

He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Heaton, J. (2017). The number of hidden layers. *URL http://www.heatonresearch.com/2017/06/01/hidden-layers.html.*

Hinton, G. E., Osindero, S., and Teh, Y.-W. (2006). A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554.

Hinton, G. E. and Zemel, R. S. (1994). Autoencoders, minimum description length and helmholtz free energy. In *Advances in neural information processing systems*, pages 3–10.

Kell, A. J., Yamins, D. L., Shook, E. N., Norman-Haignere, S. V., and McDermott, J. H. (2018). A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy. *Neuron*, 98(3):630–644.

Kemp, D. T. (1978). Stimulated acoustic emissions from within the human auditory system. *The Journal of the Acoustical Society of America*, 64(5):1386–1391.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980.*

Kishan Maladkar (2018). Types of activation functions in neural networks and rationale behind it. `https://www.analyticsindiamag.com/most-common-activation-functions-in-neural-networks-and-rationale-behind-it/`. Last checked on 2019-05-27.

Kujawa, S. G. and Liberman, M. C. (2009). Adding insult to injury: cochlear nerve degeneration after temporary noise-induced hearing loss. *Journal of Neuroscience*, 29(45):14077–14085.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551.

Liang, M. and Hu, X. (2015). Recurrent convolutional neural network for object recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3367–3375.

Lopez-Poveda, E. A., Plack, C. J., Meddis, R., and Blanco, J. L. (2005). Cochlear compression in listeners with moderate sensorineural hearing loss. *Hearing research*, 205(1-2):172–183.

Lu, X., Tsao, Y., Matsuda, S., and Hori, C. (2013). Speech enhancement based on deep denoising autoencoder. In *Interspeech*, pages 436–440.

Lyon, R. F. (2018). *Human and Machine Hearing.* Cambridge University Press book.

Lyon, R. F., Katsiamis, A. G., and Drakakis, E. M. (2010). History and future of auditory filter models. In *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pages 3809–3812. IEEE.

Marr, D. and Poggio, T. (1979). A computational theory of human stereo vision. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 204(1156):301–328.

Michelsanti, D. and Tan, Z.-H. (2017). Conditional generative adversarial networks for speech enhancement and noise-robust speaker verification. *arXiv preprint arXiv:1709.01703*.

Moller, A. R. (1994). Auditory neurophysiology. *Journal of Clinical Neurophysiology*, 11(3):284–308.

Ni, G., Elliott, S. J., Ayat, M., and Teal, P. D. (2014). Modelling cochlear mechanics. *BioMed research international*, 2014.

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

Pascual, S., Bonafonte, A., and Serrà, J. (2017). Segan: Speech enhancement generative adversarial network. *arXiv preprint arXiv:1703.09452*.

Paul, D. B. and Baker, J. M. (1992). The design for the wall street journal-based csr corpus. In *Proceedings of the workshop on Speech and Natural Language*, pages 357–362. Association for Computational Linguistics.

Pickles, J. (2013). *An introduction to the physiology of hearing.* Brill.

Raufer, S. and Verhulst, S. (2016). Otoacoustic emission estimates of human basilar membrane impulse response duration and cochlear filter tuning. *Hearing research*, 342:150–160.

Recio, A. and Rhode, W. S. (2000). Basilar membrane responses to broadband stimuli. *The Journal of the Acoustical Society of America*, 108(5):2281–2298.

Rhode, W. S. (1971). Observations of the vibration of the basilar membrane in squirrel monkeys using the mössbauer technique. *The Journal of the Acoustical Society of America*, 49(4B):1218–1231.

Rhode, W. S. and Robles, L. (1974). Evidence from mössbauer experiments for nonlinear vibration in the cochlea. *The Journal of the Acoustical Society of America*, 55(3):588–596.

Riesenhuber, M. and Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature neuroscience*, 2(11):1019.

Rosen, S., Baker, R. J., and Darling, A. (1998). Auditory filter nonlinearity at 2 khz in normal hearing listeners. *The Journal of the Acoustical Society of America*, 103(5):2539–2550.

Ruggero, M. A., Rich, N. C., Recio, A., Narayan, S. S., and Robles, L. (1997). Basilar-membrane responses to tones at the base of the chinchilla cochlea. *The Journal of the Acoustical Society of America*, 101(4):2151–2163.

Ruggero, M. A., Robles, L., and Rich, N. C. (1992). Two-tone suppression in the basilar membrane of the cochlea: Mechanical basis of auditory-nerve rate suppression. *Journal of neurophysiology*, 68(4):1087–1099.

Sakti, S., Markov, K., Nakamura, S., and Minker, W. (2009). *Incorporating knowledge sources into statistical speech recognition*, volume 42. Springer Science & Business Media.

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

Shera, C. A., Guinan, J. J., and Oxenham, A. J. (2002). Revised estimates of human cochlear tuning from otoacoustic and behavioral measurements. *Proceedings of the National Academy of Sciences*, 99(5):3318–3323.

Shera, C. A., Guinan, J. J., and Oxenham, A. J. (2010). Otoacoustic estimation of cochlear tuning: validation in the chinchilla. *Journal of the Association for Research in Otolaryngology*, 11(3):343–365.

Stevens, S. S. (2017). *Psychophysics: Introduction to its perceptual, neural and social prospects*. Routledge.

Sun, L., Du, J., Dai, L.-R., and Lee, C.-H. (2017). Multiple-target deep learning for lstm-rnn based speech enhancement. In *Hands-free Speech Communications and Microphone Arrays (HSCMA), 2017*, pages 136–140. IEEE.

Szegedy, C., Toshev, A., and Erhan, D. (2013). Deep neural networks for object detection. In *Advances in neural information processing systems*, pages 2553–2561.

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

Verhulst, S., Bharadwaj, H. M., Mehraei, G., Shera, C. A., and Shinn-Cunningham, B. G. (2015). Functional modeling of the human auditory brainstem response to broadband stimulation. *The Journal of the Acoustical Society of America*, 138(3):1637–1659.

Verhulst, S., Dau, T., Harte, J., et al. (2010). *Characterizing and modeling dynamic processes in the cochlea using otoacoustic emissions*.

Verhulst, S., Dau, T., and Shera, C. A. (2012). Nonlinear time-domain cochlear model for transient stimulation and human otoacoustic emission. *The Journal of the Acoustical Society of America*, 132(6):3842–3848.

Von Békésy, G. (1970). Travelling waves as frequency analysers in the cochlea. *Nature*, 225(5239):1207.

Von Békésy, G. and Wever, E. G. (1960). *Experiments in hearing*, volume 8. McGraw-Hill New York.

Von Helmholtz, H. and Ellis, A. J. (1875). *On the Sensations of Tone as a Physiological Basis for the Theory of Music*. London: Longmans, Green and Company.

Wegel, R. and Lane, C. (1924). The auditory masking of one pure tone by another and its probable relation to the dynamics of the inner ear. *Physical review*, 23(2):266.

Xu, Y., Du, J., Dai, L.-R., and Lee, C.-H. (2015). A regression approach to speech enhancement based on deep neural networks. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 23(1):7–19.

Yu, D., Kolbæk, M., Tan, Z.-H., and Jensen, J. (2017). Permutation invariant training of deep models for speaker-independent multi-talker speech separation. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 241–245. IEEE.

Zweig, G. (1991). Finding the impedance of the organ of corti. *The Journal of the Acoustical Society of America*, 89(3):1229–1254.

Appendices

Appendix A

Characteristic Frequencies

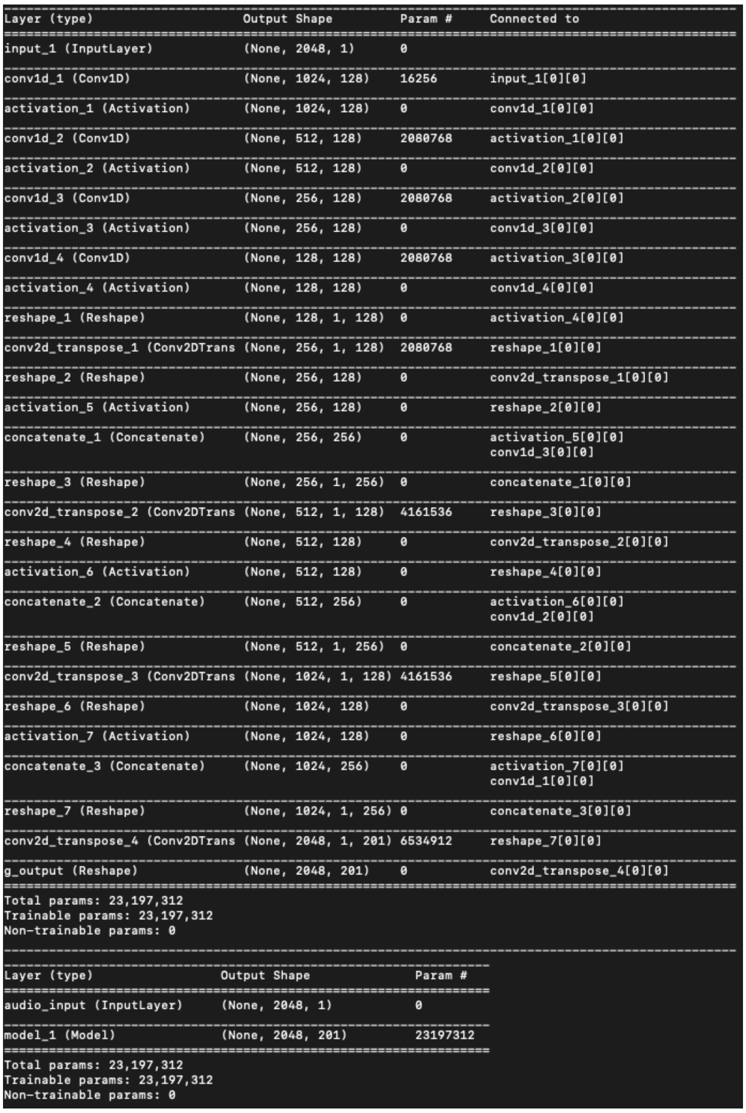
CS	CF (Hz)	CS	CF (Hz)	CS	CF (Hz)	CS	CF (Hz)
1	113	**51**	527	**101**	1,616	**151**	4,479
2	118	**52**	540	**102**	1,650	**152**	4,569
3	123	**53**	553	**103**	1,685	**153**	4,661
4	128	**54**	567	**104**	1,721	**154**	4,755
5	134	**55**	581	**105**	1,757	**155**	4,850
6	139	**56**	595	**106**	1,794	**156**	4,948
7	144	**57**	609	**107**	1,832	**157**	5,047
8	150	**58**	624	**108**	1,870	**158**	5,149
9	156	**59**	639	**109**	1,910	**159**	5,252
10	162	**60**	654	**110**	1,950	**160**	5,357
11	167	**61**	670	**111**	1,990	**161**	5,465
12	173	**62**	685	**112**	2,032	**162**	5,574
13	180	**63**	702	**113**	2,074	**163**	5,686
14	186	**64**	718	**114**	2,118	**164**	5,799
15	192	**65**	735	**115**	2,162	**165**	5,915
16	199	**66**	752	**116**	2,207	**166**	6,034
17	205	**67**	769	**117**	2,253	**167**	6,154
18	212	**68**	787	**118**	2,299	**168**	6,277
19	219	**69**	805	**119**	2,347	**169**	6,403
20	226	**70**	824	**120**	2,396	**170**	6,530
21	233	**71**	842	**121**	2,445	**171**	6,661
22	240	**72**	862	**122**	2,496	**172**	6,794

CS	CF (Hz)	CS	CF (Hz)	CS	CF (Hz)	CS	CF (Hz)
23	248	**73**	881	**123**	2,547	**173**	6,929
24	255	**74**	901	**124**	2,600	**174**	7,067
25	263	**75**	922	**125**	2,653	**175**	7,208
26	271	**76**	942	**126**	2,708	**176**	7,351
27	279	**77**	963	**127**	2,763	**177**	7,498
28	287	**78**	985	**128**	2,820	**178**	7,647
29	296	**79**	1,007	**129**	2,878	**179**	7,799
30	304	**80**	1,029	**130**	2,937	**180**	7,954
31	313	**81**	1,052	**131**	2,997	**181**	8,112
32	322	**82**	1,076	**132**	3,058	**182**	8,273
33	331	**83**	1,099	**133**	3,121	**183**	8,437
34	340	**84**	1,124	**134**	3,184	**184**	8,605
35	349	**85**	1,148	**135**	3,249	**185**	8,776
36	359	**86**	1,173	**136**	3,316	**186**	8,950
37	369	**87**	1,199	**137**	3,383	**187**	9,128
38	379	**88**	1,225	**138**	3,452	**188**	9,309
39	389	**89**	1,252	**139**	3,522	**189**	9,493
40	399	**90**	1,279	**140**	3,594	**190**	9,681
41	410	**91**	1,307	**141**	3,666	**191**	9,873
42	420	**92**	1,335	**142**	3,741	**192**	10,069
43	431	**93**	1,364	**143**	3,817	**193**	10,268
44	443	**94**	1,393	**144**	3,894	**194**	10,471
45	454	**95**	1,423	**145**	3,973	**195**	10,679
46	466	**96**	1,454	**146**	4,053	**196**	10,890
47	477	**97**	1,485	**147**	4,135	**197**	11,105
48	489	**98**	1,517	**148**	4,218	**198**	11,325
49	502	**99**	1,549	**149**	4,304	**199**	11,549
50	514	**100**	1,582	**150**	4,390	**200**	11,777
						201	12,010

Table A.1: Corresponding characteristic frequency (CF) to each cochlear section (CS)

Appendix B

AECNN Model Summary



```
--------------------------------------------------------------------------------------------------
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            (None, 2048, 1)      0
--------------------------------------------------------------------------------------------------
conv1d_1 (Conv1D)               (None, 1024, 128)    16256       input_1[0][0]
--------------------------------------------------------------------------------------------------
activation_1 (Activation)       (None, 1024, 128)    0           conv1d_1[0][0]
--------------------------------------------------------------------------------------------------
conv1d_2 (Conv1D)               (None, 512, 128)     2080768     activation_1[0][0]
--------------------------------------------------------------------------------------------------
activation_2 (Activation)       (None, 512, 128)     0           conv1d_2[0][0]
--------------------------------------------------------------------------------------------------
conv1d_3 (Conv1D)               (None, 256, 128)     2080768     activation_2[0][0]
--------------------------------------------------------------------------------------------------
activation_3 (Activation)       (None, 256, 128)     0           conv1d_3[0][0]
--------------------------------------------------------------------------------------------------
conv1d_4 (Conv1D)               (None, 128, 128)     2080768     activation_3[0][0]
--------------------------------------------------------------------------------------------------
activation_4 (Activation)       (None, 128, 128)     0           conv1d_4[0][0]
--------------------------------------------------------------------------------------------------
reshape_1 (Reshape)             (None, 128, 1, 128)  0           activation_4[0][0]
--------------------------------------------------------------------------------------------------
conv2d_transpose_1 (Conv2DTrans (None, 256, 1, 128)  2080768     reshape_1[0][0]
--------------------------------------------------------------------------------------------------
reshape_2 (Reshape)             (None, 256, 128)     0           conv2d_transpose_1[0][0]
--------------------------------------------------------------------------------------------------
activation_5 (Activation)       (None, 256, 128)     0           reshape_2[0][0]
--------------------------------------------------------------------------------------------------
concatenate_1 (Concatenate)     (None, 256, 256)     0           activation_5[0][0]
                                                                 conv1d_3[0][0]
--------------------------------------------------------------------------------------------------
reshape_3 (Reshape)             (None, 256, 1, 256)  0           concatenate_1[0][0]
--------------------------------------------------------------------------------------------------
conv2d_transpose_2 (Conv2DTrans (None, 512, 1, 128)  4161536     reshape_3[0][0]
--------------------------------------------------------------------------------------------------
reshape_4 (Reshape)             (None, 512, 128)     0           conv2d_transpose_2[0][0]
--------------------------------------------------------------------------------------------------
activation_6 (Activation)       (None, 512, 128)     0           reshape_4[0][0]
--------------------------------------------------------------------------------------------------
concatenate_2 (Concatenate)     (None, 512, 256)     0           activation_6[0][0]
                                                                 conv1d_2[0][0]
--------------------------------------------------------------------------------------------------
reshape_5 (Reshape)             (None, 512, 1, 256)  0           concatenate_2[0][0]
--------------------------------------------------------------------------------------------------
conv2d_transpose_3 (Conv2DTrans (None, 1024, 1, 128) 4161536     reshape_5[0][0]
--------------------------------------------------------------------------------------------------
reshape_6 (Reshape)             (None, 1024, 128)    0           conv2d_transpose_3[0][0]
--------------------------------------------------------------------------------------------------
activation_7 (Activation)       (None, 1024, 128)    0           reshape_6[0][0]
--------------------------------------------------------------------------------------------------
concatenate_3 (Concatenate)     (None, 1024, 256)    0           activation_7[0][0]
                                                                 conv1d_1[0][0]
--------------------------------------------------------------------------------------------------
reshape_7 (Reshape)             (None, 1024, 1, 256) 0           concatenate_3[0][0]
--------------------------------------------------------------------------------------------------
conv2d_transpose_4 (Conv2DTrans (None, 2048, 1, 201) 6534912     reshape_7[0][0]
--------------------------------------------------------------------------------------------------
g_output (Reshape)              (None, 2048, 201)    0           conv2d_transpose_4[0][0]
==================================================================================================
Total params: 23,197,312
Trainable params: 23,197,312
Non-trainable params: 0
--------------------------------------------------------------------------------------------------
-----------------------------------------------------------------
Layer (type)                 Output Shape              Param #
=================================================================
audio_input (InputLayer)     (None, 2048, 1)           0
-----------------------------------------------------------------
model_1 (Model)              (None, 2048, 201)         23197312
=================================================================
Total params: 23,197,312
Trainable params: 23,197,312
Non-trainable params: 0
```

Figure B.1: The Keras model summary for an 8 layer AECNN model

Appendix C

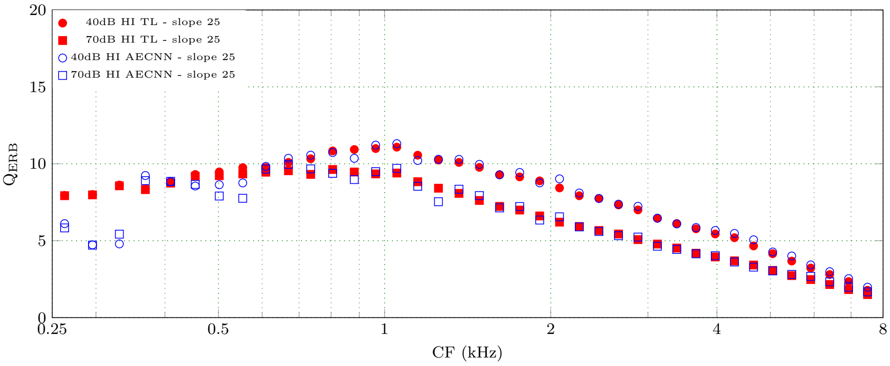
Trained AECNN Models - NH

Table C.1: Overview trained NH AECNN models. Amount of parameters, loss term, time per epoch and MSE of the basic input stimuli and test set predictions for the depicted AECNN models on the left.

Model (NL/Depth/lr/FL)	Param.	LT	Time/Epoch	Click	1kHz	4kHz	Testset
				x10^{-4}	x10^{-4}	x10^{-4}	x10^{-4}
PReLU/4 lay./0.001/63	5,641,856	0.0404	1h46m12s	23.74	71.08	23.69	57.10
PReLU/4 lay./0.0004/63	5,641,856	0.0375	1h48m36s	10.63	61.18	63.79	46.40
PReLU/4 lay./0.0001/63	5,641,856	0.0376	1h46m40s	9.99	166.85	16.49	49.66
PReLU/6 lay./0.0001/63	8,836,736	0.0307	1h49m41s	4.41	160.75	15.29	53.95
PReLU/6 lay./0.0001/31	4,564,608	0.0360	1h44m20s	7.07	176.81	30.84	48.29
PReLU/6 lay./0.0001/127	17,380,992	0.0298	1h58m20s	4.45	93.09	13.32	45.43
tanh/6 lay./0.0001/127	16,955,008	0.0148	1h59m05s	1.21	4.92	6.05	20.06
tanh/4 lay./0.0001/127	10,712,704	0.0185	1h54m42s	1.63	3.95	8.63	24.04
tanh/8 lay./0.0001/127	23,197,312	0.0140	2h01m40s	1.13	4.29	8.64	15.19

Appendix D

Performance HI AECNN on Slope 25 - Flat 25



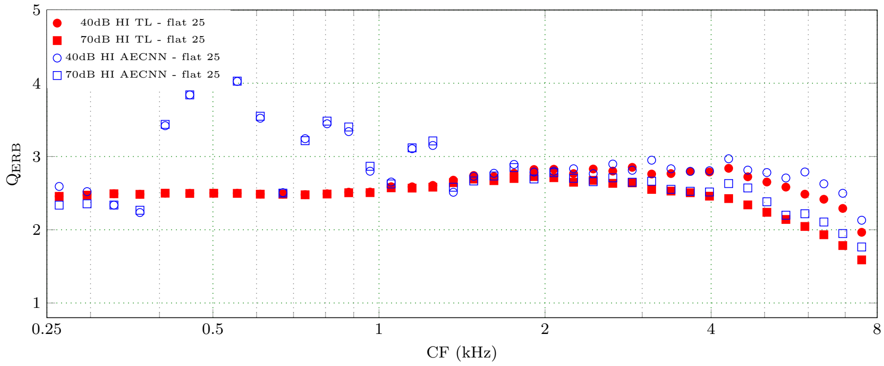


Figure D.1: $\mathbf{Q_{ERB}}$ values for trained HI AECNN - slope 25 / flat 25. Q_{ERB} values for the energy underneath the power spectrum of CF impulse responses to a 100 μs click of different intensities (40 and 70 dB). Simulations are shown for the HI TL model (red) and trained HI AECNN model (blue) for respectively a slope 25 HL profile (Top) and a flat 25 HL profile (Bottom). Refer to Subsection 5.2.9 for an explanation for the suboptimal low frequency performance.

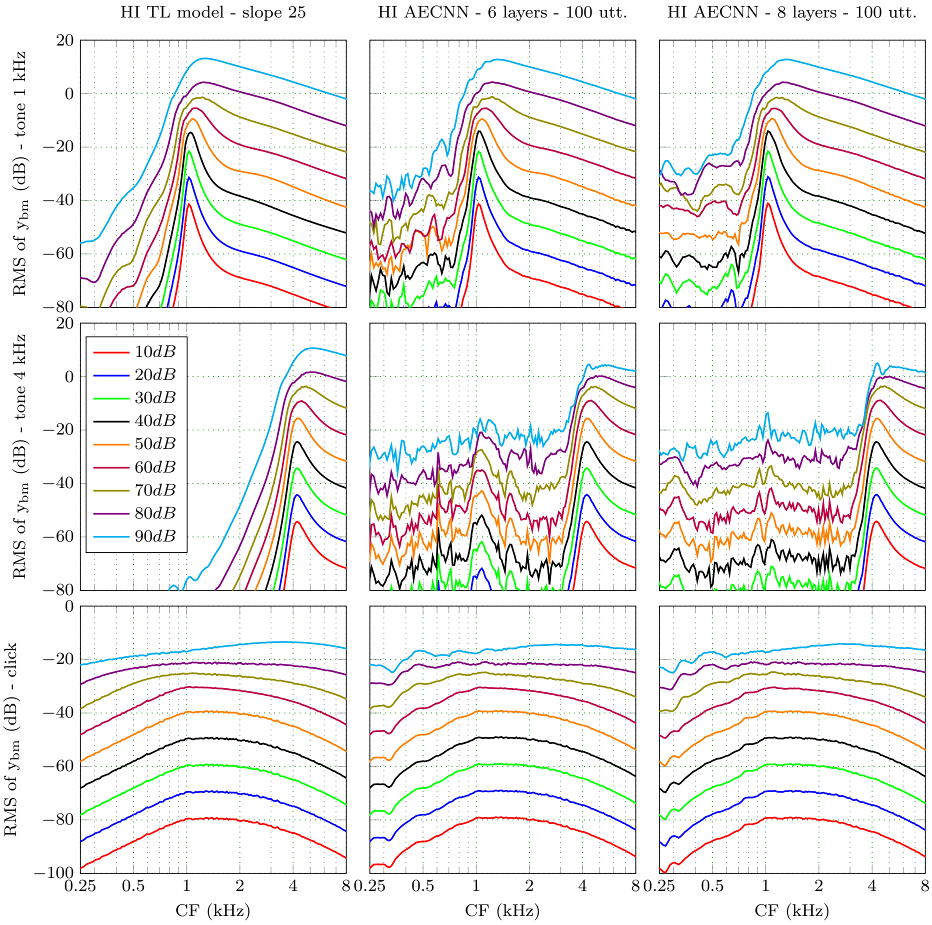


Figure D.2: Comparison of excitation patterns - slope 25 - Variable depth. Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section (corresponding CFs are listed in Appendix A) for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model that is made HI with a slope 25 cochlear gain loss profile (left) and trained HI AECNN architectures varying in depth.

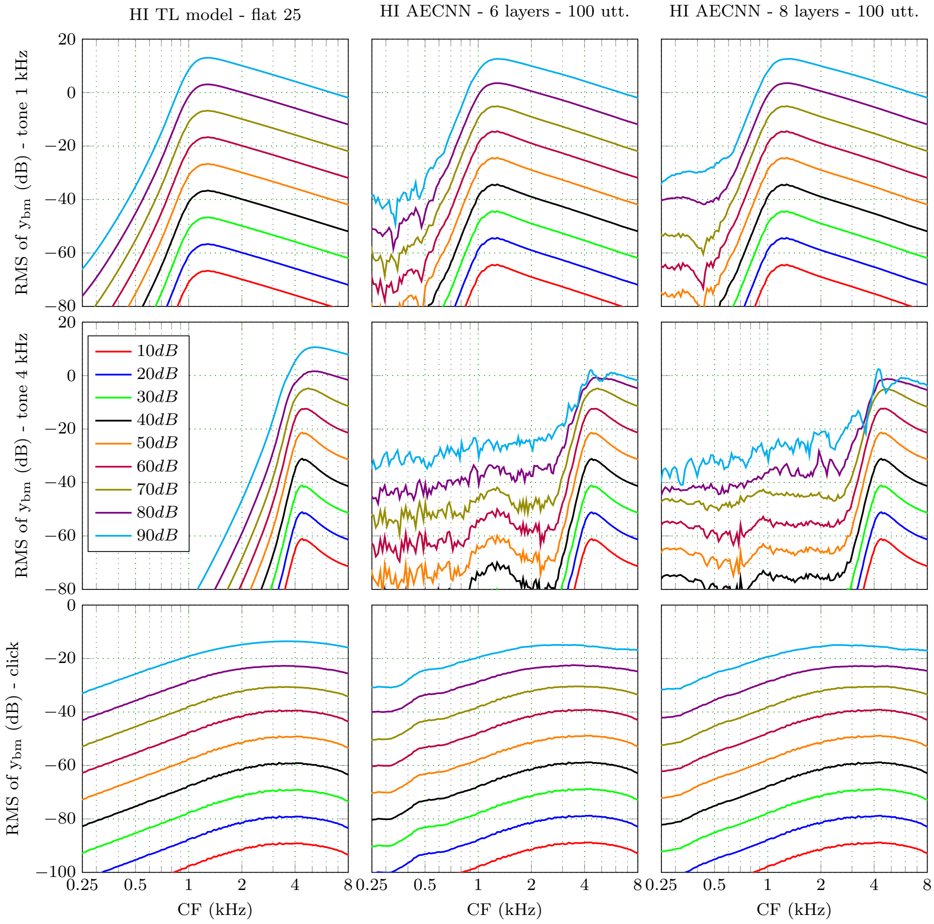


Figure D.3: Comparison of excitation patterns - flat 25 - Variable depth. Cochlear excitation patterns calculated as the RMS value of the BM displacement (y_{BM}) per cochlear section (corresponding CFs are listed in Appendix A) for a stimulation with a 1 kHz pure tone (top row), 4 kHz pure tone (middle row) and click stimulus (bottom row) with intensity levels ranging between 10 and 90 dBSPL. The depicted models are the reference TL model that is made HI with a flat 25 cochlear gain loss profile (left) and trained HI AECNN architectures varying in depth.

FACULTY OF ENGINEERING
AND ARCHITECTURE

Towards Individualized Hearing Profiles Using Deep Neural Nets

Arthur Van Den Broucke
Student number: 01301281

Supervisors: Prof. dr. Sarah Verhulst, Dr. ir. Deepak Baby (Universiteit Gent)

Master's dissertation submitted in order to obtain the academic degree of
Master of Science in Biomedical Engineering

Academic year 2018-2019

GHENT
UNIVERSITY