Recurrent Gradient-based Motor Inference for Speech Resynthesis with a Vocal Tract Simulator

Konstantin Sering, Paul Schmidt-Barbo, Sebastian Otte,
Martin V. Butz, Harald Baayen

Eberhard Karls Universität Tübingen, Germany
{konstantin.sering, paul.schmidt-barbo, sebastian.otte,
martin.butz, harald.baayen}@uni-tuebingen.de

Abstract

We describe an inference principle for speech resynthesis using the vocal tract simulator VocalTractLab (VTL). Our method generates smooth and plausible motor trajectories controlling the vocal tract simulator. The method utilizes a differentiable forward model approximation of the VTL, namely, an LSTM that learned the involved temporal motor-acoustics relations. Motor trajectories are inferred through temporal gradient information minimizing the error between the forward prediction and the target acoustics, explicitly incorporating velocity and jerk constraints. We show that our method provides good parameter recovery and generalization, and that it achieves a decent synthesis quality. While the proposed principle is a promising tool for studying the mechanics of human speech generation, it comes with the cost of a significant computational overhead.

Keywords: articulatory synthesis, speech synthesis, motor trajectories

1. Introduction

This study is part of an ongoing project addressing the challenge of learning an inverse mapping between acoustic features and *control parameter trajectories* (cp-trajectories) of a vocal tract simulator. We apply a temporal gradient-based inference method from Otte, Schmitt, et al. (2017) and Martin V. Butz et al. (2019), which is inspired by Friston's active inference principle (Friston, Samothrakis, and Montague 2012), to the dynamics of the vocal tract simulator developed by Birkholz (2013), the *VocalTractLab* (VTL), in version 2.3.

To implement this principle, we use a predictive forward model that receives cp-trajectories as inputs and learns to predict the corresponding *acoustic representations*, in form of log-mel spectrograms. Therefore, the predictive forward model emulates the behavior of the vocal tract simulator, but is much faster to execute and fully differentiable. If the predictions of the forward model are locally of sufficient quality, the gradients of the forward model can be used to plan cp-trajectories for a given acoustic target.

A common problem in inferring cp-trajectories from an acoustic target is that many cp-trajectories result in very similar or identical acoustic manifestations. This is especially prominent for silence where a full closure at any part of the vocal tract results in no speech, i. .e silence. The predictive model maps many different cp-trajectories to one acoustic state, whereas an inverse model has to derive from one acoustic state many different cp-trajectories. By mainly relying on the predictive forward model in the inference the many-to-one problem is circumvented. Starting from an initial guess of cp-trajectories the gradient of the predictive forward model can be used iteratively to improve the initial cp-trajectories. The gradient of matching the predicted acoustics with the target acoustics can be jointly minimized with velocity and jerk constraints, i. e. minimize motor effort. This allows planning smooth cp-trajectories that produce speech closely matching the target acoustics.

Using a predictive forward model is different to most frameworks deriving cp-trajectories for the VTL simulator. Cp-trajectories have been derived mostly by means of a segment-based approach that takes phone sequences as inputs and derives gestural scores out of them. Next, the gestural scores are turned into cp-trajectories. An exception is a LSTM-based direct inverse model presented in Gao, Steiner, and Birkholz (2020). Here, we present and evaluate an LSTM-based predictive forward model, which uses gradient-based inference to generate cp-trajectories in a goal-directed manner.

2. Methods

The recurrent gradient-based motor inference framework for speech resynthesis has two main data structures (gold boxes in **Figure 1**). First, the control parameter (cp) trajectories define the positions of the articulators over time and the glottis parameters of the glottis model for the acoustical synthesis. Second, the acoustic representation that defines the perceptual image of the acoustics. These two data structures are connected in three ways (red boxes and arrows in **Figure 1**).

The first connection and the ground truth for this framework is given by the VocaltTractLab (VTL) simulator, which takes cp-trajectories as inputs and produces a mono audio signal as output. The audio signal is then deterministicly converted into the acoustic representation. The second connection is the inverse model, which takes the acoustic representation as input and predicts a cp-trajectory. The inverse model is used only once at the beginning of each cp-trajectory-inference process. The third and most important connection is the predictive forward model, which shortcuts the first connection and directly connects the cp-trajectories with the acoustic representation. The gradient, which is inferred inversely via the predictive forward model, is used to optimize the cp-trajectory during action inference. All models are implemented in Python 3.7 and PyTorch 1.6.0.

2.1. Cp-trajectories

Cp-trajectories are the inputs to the VTL articulatory speech synthesizer. They define the positions of the articulators and the parameters of the glottis model over time and are the motor part

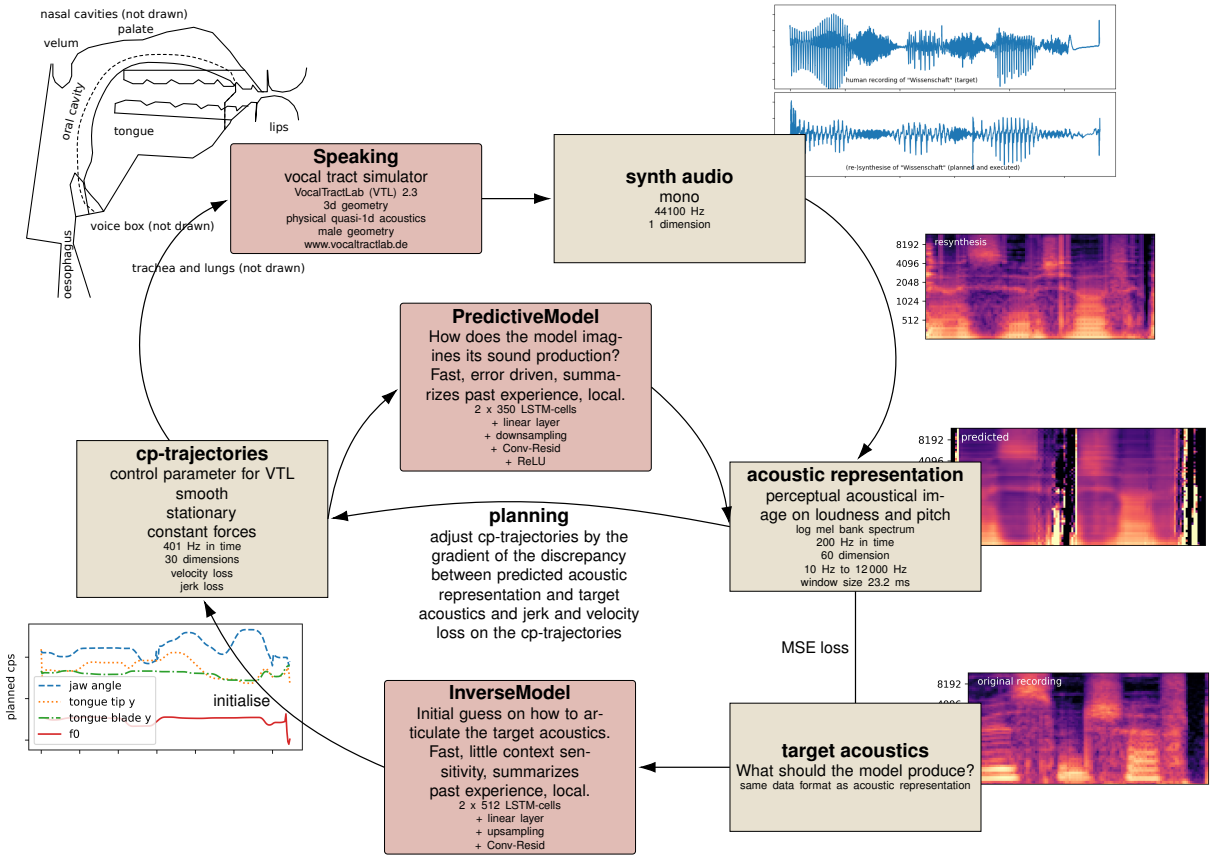

Figure 1: Implementation of the recurrent gradient-based motor inference principle with LSTM based networks. The predictive model imagines the acoustic representation and allows for adjustment prior to execution. The inverse model is only used for initialization.

of the framework. Control parameters (cps) need to be defined every 110 samples of the resulting 44 100 Hz mono audio signal, i. e. every ≈ 2.5 ms . Cp-trajectories should be smooth, as stationary as possible, and should be modified by slowly changing forces, i. e. to adhere to the minimum jerk principle (Viviani and Flash 1995). The 30 cps are individually and linearly scaled and centered so that -1 corresponds to the minimal theoretical value and +1 corresponds to the maximal value as given by the VTL API reference (Birkholz 2018).

2.2. Acoustic Representation

The acoustic representation encodes the human perceptual image of speech by means of pitch and loudness. The acoustic representation is implemented as a log-mel spectrogram with 60 log-mel banks within a frequency range from 10 Hz to 12 000 Hz, a window length of 1 024 samples (23.2 ms), and a time delta of 220 samples (5 ms). The log-mel banks are calculated on a magnitude spectrogram. Calculations are conducted with `librosa` (version 0.8.0) on wave forms with a sampling rate of 44 100 Hz. The log-mel spectrogram was linearly transformed into the 0 to ∞ range, where 0 corresponds to silence and 1 is a loud and clear tone. Data reduction per time slice was from 220 values in time to 60 values in pitch.

2.3. Speaking

The speaking transformation constitutes the ground truth for the predictive model: It is realized by executing the VTL simulator. VTL uses a 3-dimensional geometrical vocal tract model linked with a quasi-1-dimensional acoustic synthesis model. A mid-sagittal slice, which illustrates the vocal tract geometry, is shown in **Figure 1**. The cp-trajectory essentially specifies the geometrical shape of the vocal tract, the properties of the glottis model, and lung pressure over time. In VTL version 2.3 the resulting mono audio has a sampling rate of 44 100 Hz.

2.4. Predictive model

The recurrent predictive forward model is imagining how execution of the cp-trajectories at hand would sound. It therefore shortcuts VTL simulator and directly maps the cp-trajectories to the acoustic representation. The gradient of the predictive model is later used to correct the initial cp-trajectories via action inference.

The recurrent predictive forward model has an input size of 31 input channels. These comprise the 30 control parameters of the VTL and an extra onset channel that helps initialising the weights on the first time step. The onset dimension is set to one in the first time step and set to zero on all other time steps. The output size is 60 channels, corresponding to the 60 log-mel spectrograms of the acoustic representation. Note that the time resolution is halved from 401 Hz to 200.5 Hz in order to match the time resolution of the acoustic representation.

In a first step, the velocities and accelerations of the cp-trajectory input are calculated (time deltas and delta-deltas) and appended to the state input, resulting in a total of 91 channels.

The 91 input channels are then fed into a two-layer LSTM with 350 LSTM cells followed by a linear output layer of size 60, corresponding already to the 60 output channels. Next, the sequence length is halved by average pooling. In order to account for local corrections we apply a convolution layer with a kernel size of 5 in time and 3 in mel channel in a skip layer fashion. Only one local filter per output channel is learned and all filters are learned independently. Finally, all negative values are clipped to 0 with a ReLu. To mitigate the dying ReLu problem, the biases in the correction layer are initialized strictly positive with 0.01. The total number of trainable parameters in the predictive forward model is 1 625 020.

2.5. Inverse model

The inverse model creates the initial guess of the cp-trajectory needed to produce the target acoustics. It gets a log-mel spectrogram as input and outputs a cp-trajectory. The inverse model is implemented analogous to the predictive model but with 512 LSTM cells, five skip layer convolutions of size 5 in time and width 1 in channels and a final linear layer instead of a ReLu. The total number of trainable parameters in the inverse model is 3 540 898.

2.6. Action inference

Action inference plans a cp-trajectory aiming at producing audio that closely matches the target wave signal with a sampling rate of 44 100 Hz. In a first step the model calculates the log-mel spectrogram for the acoustic target. The inverse model then creates the initial cp-trajectory taking the determined log-mel spectrogram dynamics as input. Next, the initial cp-trajectories are passed through the predictive forward model, generating a consequent log-mel spectrogram prediction, which is then compared to the target acoustics. On the basis of the mean squared error loss (MSE loss) between the two spectrograms and additional velocity- and jerk-losses, which are applied to the input cp-trajectories, the local gradients for the input cp-trajectories are calculated and the initial cp-trajectories are adjusted by 0.05 times the local gradients. This corresponds to a stochastic gradient decent (SGD) update. The resulting adjusted cp-trajectories are again fed into the predictive forward model, are again adjusted, etc. This procedure is repeated 200 times in total. The initial velocity loss is weighted relative to the initial MSE loss with 0.5 and the initial jerk loss has the same size as the initial MSE loss.

SGD is used as ADAM (Kingma and Ba 2015) does not work out here as the inputs should be adjusted and smoothed during planning. Applying ADAM updating diverged after some initial decrease in the loss and yields to erratic cp-trajectories. Note that ADAM normalizes the individual gradient components independently, whereas the individual magnitudes of the gradient components and their mutual relations, which both may be highly relevant for our problem domain, are thrown out (Otte, Hofmaier, and Martin V. Butz 2018).

After the 200 planning steps the control parameters are fed into the actual VTL simulator and the corresponding audio is created. In order to sync the local approximation of the predictive model with the ground truth of the VTL simulator even further, the cp-trajectories together with the produced audio are now used as a new training sample together with 10 training samples from the initial training data. As a result, the predictive model is further learned and attuned to the VTL's speech-generation properties close to the target utterance. Planning and learning are repeated 40 times. The final inferred cp-trajectory is returned as the result of this active inference process.

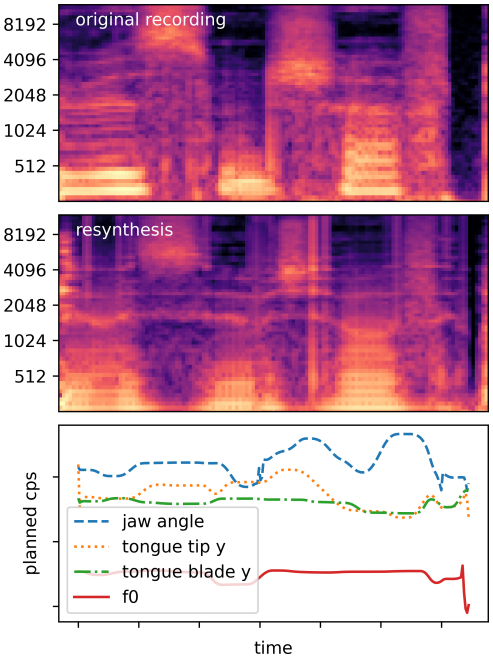

Figure 2: The top panel shows the log-mel spectrogram of the original human recording *Wissenschaft* (science) used as the target. The middle panel shows the resulting log-mel spectrogram after the planned trajectories are executed by the VTL. The bottom panel shows four selected cp-trajectories after planning.

2.7. Initial training

The initial training imprints some experience on the relation between cp-trajectories and acoustic representation into the models. As the model has no notion of phones or gestures these notions are to some extent imprinted by the initial training. This experience essentially leads to the development of an approximate generative model of the VTL simulator.

The predictive and the inverse model are initially trained with cp-trajectories and log-mel spectrograms for German words. Each training sample corresponds to one German word with a sequence length between 246 and 1976 time steps. The log-mel spectrogram is computed on speech synthesised by VTL, based on cp-trajectories derived from phone segment data of the GECO corpus Schweitzer and Lewandowski (2013). Synthesising speech from segment data was introduced in VTL 2.3 and is accomplished approximately as described in Sering et al. (2019). Only the first 5 000 spliced-out word tokens are used, which corresponds to one hour of speech. Of these tokens, 4 500 were assigned to the training set, and 500 tokens to the test set. Both models are trained using ADAM as the learning rule with an initial learning rate of 0.001 and default parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$), applying the MSE loss, a batch size of 8, and 100 epochs of training in total.

3. Results

To test recovery and generalisation of the recurrent gradient-based motor inference framework 30 word tokens from the test set of the initial training data were used. In the recovery test, the target acoustics is produced. Mono audio from the VTL sim-

ulator and the cp-trajectories are known. The gradient-based framework does not achieve full recovery, but improves the MSE loss in the actual synthesis by $53.2\% \pm 15.8\%$. The final MSE loss is 0.0706 ± 0.0266. In the second test target acoustics are the recordings from the GECO corpus, which are matched to the 30 words from the test set. These are recordings from a female speaker. The data from the initial training set comprise the base line with a MSE loss of 0.0772 ± 0.0246. The gradient-based framework achieves to half the MSE loss to 0.0313 ± 0.0103. The loss reduction from the initial cp-trajectories produced by the inverse model is $42.9\% \pm 17.8\%$. Note that the gradient-based framework has no access to the phone segment data, but only works on the target audio, whereas the base line uses aligned phone segments as inputs, but no audio. Computationally, the gradient-based framework is around 1 000 times costlier.

As an example, **Figure 1** shows the wave form of the recorded utterance of the German word *Wissenschaft* ('science') in the top right corner, which is used as target, along with the wave form of a planned resynthesis below. The corresponding planned cp-trajectory is relatively smooth and flat (in some areas even too flat). The log-mel spectrograms show a lot of similarities, but provide less contrast in the resynthesis as in the original recording (**Figure 2**). Even if the resynthesis is intelligible in most cases there is still room for improvement in quality.

4. Discussion and Conclusion

The goal of imitating human speech production on a high level without phones and gestures but by complementing the physics of the speech process with learning experience led to the choices in this framework. The presented framework is a simplistic models that still capture these aspects. While it oversimplifies some of the known aspects of human speech production, it shows first promising results to imitate natural human speech, including its reductions.

The acoustic representation implemented as a log-mel spectrogram approximates the human hearing process up to the inner ear (cochlea) in loudness and pitch but without phase. This is a rough approximation as it is known that humans can perceive phase shifts in the lower frequency range, and humans have dynamic range adaptation in loudness. Furthermore, frequencies are capped at 12 000 Hz, even if most young humans can hear up to 20 000 Hz. MFCCs, a popular choice in speech recognition systems, are deliberately not used as they are hardly motivated by the mechanical parts of the human hearing. Nonetheless, MFCCs have very good data reduction properties and could reduce the acoustic representation from 60 to 13 channels possibly without loosing any relevant information.

For the human speech organ a three dimensional geometrical model of the vocal tract is used, which has transparent input parameters that define the position of the tongue, the jaw, the lip opening and rounding, and the tongue side elevation as well as parameters for the glottis model. Limitations are that the VTL model does not model any biomechanics or muscle contractions. Furthermore, the acoustical model in VTL is not capable of modeling transversal acoustic waves, which might add to the sound quality above 6 000 Hz, i. e. a wave length of roughly 5 cm.

The predictive planning is motivated by the anticipatory nature of human cognition. Humans seem to be sensitive to discrepancies between their predictions and the actual perceived world. We appear to constantly predict the near future and adjust our behavior based on the experienced prediction error. This results in very flexible and adaptive but still locally consistent behavior (Martin V Butz and Kutter 2016).

Next steps involve a more quantitative evaluation and comparison of the presented framework with the segment-based approach shipped with VTL and human behavior by focusing on coarticulation patterns in the acoustical and in the movement domain in reduced speech. Moreover, we intend to add a word classifier to the acoustic representation to emphasize semantic contrast during planning. Relying on the error of the word classifier should help minimizing the mimicry of noise sounds, while improving intelligibility. Moreover, we expect that the classifier helps to produce speech in the tonality of the vocal tract geometry rather than imitating different speakers as closely as possible.

Acknowledgments: This research was supported by an ERC advanced Grant (no. 742545).

5. References

Birkholz, Peter (Apr. 2013). "Modeling Consonant-Vowel Coarticulation for Articulatory Speech Synthesis". In: *PLOS ONE* 8.4, pp. 1–17. DOI: 10.1371/journal.pone.0060603.

— (2018). URL: http://www.vocaltractlab.de/index.php?page=vocaltractlab-about.

Butz, Martin V and Esther F Kutter (2016). *How the mind comes into being: Introducing cognitive science from a functional and computational perspective*. Oxford University Press.

Butz, Martin V., David Bilkey, Dania Humaidan, Alistair Knott, and Sebastian Otte (2019). "Learning, planning, and control in a monolithic neural event inference architecture". In: *Neural Networks* 117, pp. 135–144.

Friston, Karl, Spyridon Samothrakis, and Read Montague (2012). "Active inference and agency: optimal control without cost functions". In: *Biological cybernetics* 106.8-9, pp. 523–541.

Gao, Yingming, Peter Steiner, and Peter Birkholz (2020). "Articulatory Copy Synthesis using Long-Short Term Memory Networks". In: *Studientexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung 2020*. Ed. by Ronald Böck, Ingo Siegert, and Andreas Wendemuth. TUDpress, Dresden, pp. 52–59.

Kingma, Diederik P. and Jimmy L. Ba (2015). "Adam: A Method for Stochastic Optimization". In: *3rd International Conference for Learning Representations* abs/1412.6980.

Otte, Sebastian, Lea Hofmaier, and Martin V. Butz (Oct. 2018). "Integrative Collision Avoidance Within RNN-Driven Many-Joint Robot Arms". In: *Artificial Neural Networks and Machine Learning – ICANN 2018*. Lecture Notes in Computer Science 11141. Springer International Publishing, pp. 748–758.

Otte, Sebastian, Theresa Schmitt, Karl Friston, and Martin V Butz (2017). "Inferring adaptive goal-directed behavior within recurrent neural networks". In: *International Conference on Artificial Neural Networks*. Springer, pp. 227–235.

Schweitzer, Antje and Natalie Lewandowski (2013). "Convergence of articulation rate in spontaneous speech." In: *INTERSPEECH*, pp. 525–529.

Sering, Konstantin, Niels Stehwien, Yingming Gao, Martin V Butz, and Harald Baayen (2019). "Resynthesizing the GECO speech corpus with VocalTractLab". In: *Studientexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung 2019*, pp. 95–102.

Viviani, Paolo and Tamar Flash (1995). "Minimum-jerk, two-thirds power law, and isochrony: converging approaches to movement planning." In: *Journal of Experimental Psychology: Human Perception and Performance* 21.1, p. 32.