INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Self-Supervised Solution to the Control Problem of Articulatory Synthesis

Paul K. Krug[1], *Peter Birkholz*[1], *Branislav Gerazov*[2], *Daniel R. van Niekerk*[3], *Anqi Xu*[4], *Yi Xu*[3]

[1]Institute of Acoustics and Speech Communication, Technische Universität Dresden, Germany
[2]Faculty of Electrical Engineering and Information Technologies, UKIM Skopje, R. N. Macedonia
[3]Department of Speech, Hearing and Phonetic Sciences, University College London, UK
[4]Harbin Institute of Technology, Shenzhen, China

paul_konstantin.krug@tu-dresden.de

Abstract

Given an articulatory-to-acoustic forward model, it is a priori unknown how its motor control must be operated to achieve a desired acoustic result. This control problem is a fundamental issue of articulatory speech synthesis and the cradle of acoustic-to-articulatory inversion, a discipline which attempts to address the issue by the means of various methods. This work presents an end-to-end solution to the articulatory control problem, in which synthetic motor trajectories of Monte-Carlo-generated artificial speech are linked to input modalities (such as natural speech recordings or phoneme sequence input) via speaker-independent latent representations of a vector-quantized variational autoencoder. The proposed method is self-supervised and thus, in principle, synthesizer and speaker model independent.
Index Terms: Acoustic-to-articulatory inversion, VQ-VAE

1. Introduction

Articulatory speech synthesis (ASS) aims to model the speech production mechanism as it occurs in humans. This usually involves the temporal control of individual articulators, which is a challenging task that requires expert knowledge in order to produce intelligible speech [1]. A fundamental problem in ASS is that the acoustic consequences of potential motor actions are unknown prior to synthesis. Modern articulatory synthesizers such as VocalTractLab [2] (VTL) try to circumvent this problem by providing motor state presets that correspond to known phonemes. By rule-based interpolation of these presets, intelligible speech can be generated [1, 3]. However, such control is very limited, because the corresponding motor state presets must be found from articulatory measurements (e.g., magnetic resonance imaging) and must be fine-tuned. Thus, the creation of such presets is very costly and time-consuming. Further, expanding the repertoire of speakers, languages, or phonemes requires new measurements each time. This is not scalable and undermines the potential advantage of articulatory synthesis as a low-cost, low-resource alternative [4] to state-of-the-art systems. Therefore, the development of an automatic method for the generation of motor trajectories is of central importance for the further development of ASS. A natural modality from which articulatory movements can be obtained are acoustic speech signals. The corresponding process is often referred to as acoustic-to-articulatory inversion (AAI). While numerous works have been published on AAI, studies are often either based on articulatory measurements [5–9], expert knowledge [10–12], more concerned with articulatory representations [6] and the quality of motor trajectory prediction [13, 14] and less concerned with the actual control of a human vocal tract model, or, if unsupervised, provide little detail on synthetic data generation and/or insufficient evaluation in terms of intelligibility [12, 15, 16] or provide no acoustic examples [12, 15, 16]. This paper extends the state of research with the following contributions: (i) A deep learning-based framework is presented that can control the articulatory synthesizer VTL (version dev-2.4 using VTL-Python) via AAI and phoneme-to-articulatory conversion. (ii) This work provides a detailed description of the generation of synthetic training data for such a model. (iii) Meaningful evaluations in terms of intelligibility were carried out. Audio examples are provided to demonstrate the capabilities and limitations of the system[1].

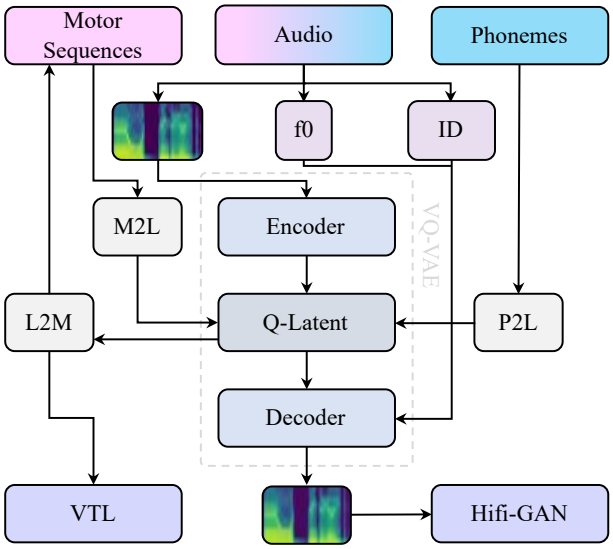

Figure 1: *Schematic of the proposed model TensorTract.*

2. Methods

The proposed system, which is referred to as *TensorTract*, is visualized in Figure 1. It consists of a vector-quantized variational autoencoder [17] (VQ-VAE), which uses TCN [18] - based encoder and decoder networks with multihead-attention [19] (MHA). Each TCN consists of a single stack of five non-causal residual blocks with dilation rates of 1, 2, 4, 8, 16, respectively. The number of filters in the convolutional layers is 80 and the kernel size is 16. Skip connections are used between the input and each residual block. The tanh function is used for the activations. The model uses 80 dimensional log-melspectrograms as input features, which are computed from 22 kHz audio signals with a window length of 1024 samples and a hop length of 220 samples. Consequently, the feature and latent sampling rates are 100 Hz. Thereby, 10 dimensional latent variables and 512 embeddings were used. The decoder was conditioned on the original pitch contours and speaker identities of the input utterances, in order to achieve high reconstruction accuracy while keeping the latent representation maximally

[1]https://tensortract.github.io/

10.21437/Interspeech.2023-2173

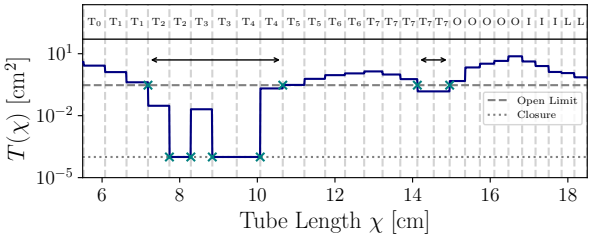

Figure 2: *Section of the area function of an implausible motor state with two constrictions, one of which is a double closure. Tokens corresponding to the individual segments are shown.*

speaker and pitch independent. The output of the decoder can be connected to a neural vocoder such as Hifi-Gan [20], which has been done in this work for evaluation purposes only. The VQ-VAE latent is connected to VTL motor trajectories via a *forward model* (mappig to the latent) and an *inverse model* (mapping from the latent). The motor-to-latent (M2L) mapping was established using an MHA-RNN hybrid model (five BiGRU [21] layers with 32, 64, 256, 256, 256 neurons and $\tanh$ activation, respectively, followed by three dense layers with GeLu [22] activation and a final dense layer with softmax activation). This model essentially replicates the operation of VTL, but with the distinct advantages of being fast and differentiable. Moreover, since this model maps to the latent and not directly to the acoustic level, the speaker voice characteristic of this forward operation can easily be changed. The inverse latent-to-motor (L2M) mapping was established using a similar network but with 256 neurons in each BiGRU layer and a $\tanh$ activation in the final layer instead of softmax. The idea is to first train the VQ-VAE model simultaneously on natural and synthetic data. Then, the M2L and L2M models can be trained on the synthetic data alone (theoretically an L2M model would be sufficient, but in reality an additional forward model can penalize prediction errors of the L2M model and thus better results can be achieved). This creates a link between natural speech data and motor trajectories. In addition, phoneme annotations of the natural speech data can be mapped to the latent (P2L), allowing a mapping of phoneme sequences to articulatory trajectories. For the P2L model, a TCN with MHA similar to the audio encoder network was used.

Natural Speech Data

With the aim of comparing the results of latent inversion with the rule-based (German) phoneme-to-speech functionality [3] of VTL, German speech data was used in the training of the VQ-VAE. For this purpose, the German part of the *Multilingual LibriSpeech* [23] (MLS) corpus was used (approximately 2000 h of speech). In addition, the *BITS unit selection corpus* [24] (BITS), version 1.7 and the *PhonDat90* part of the *Kiel Corpus of Spoken German* [25, 26], *New Edition 2017* (KIEL) were used. These data sets have much less speech material than MLS, but provide high quality phoneme annotations, which are useful for training the P2L model and for evaluation purposes.

Synthetic Speech Data

A central aspect of this work is the generation of meaningful synthetic training data. If random motor states were concatenated into an infinite data set, any combination of phonemes, and thus any word or phrase in any language, could be obtained. In reality, however, the size of a data set must always be limited due to hardware and time constraints. Therefore, it is important to sample motor trajectories that are as meaningful as possible. As a matter of fact however, randomly sampled VTL motor states are usually biologically implausible because the model has few bio-mechanical constraints [27]. Consequently, external articulatory constraints must be applied. The corresponding process is described below and is based on the ideas of Van Niekerk et al. [28] and Krug et al. [29].

Prerequisites

Let $T_{\min}$ be the minimum of the tube area function $T_{\mathbf{q}}(\chi)$ related to a given vocal tract state $\mathbf{q}$. Second, let $C(T_{\min})$ be a function of $T_{\min}$ that describes the degree of vocal tract constriction via a set of integers. Thereby, a value of $C = 0$ ($T_{\min} \geq 0.3\,\mathrm{cm}^2$) indicates an *open* vocal tract. Such a state is likely to generate a vowel-like sound when synthesized with a modal voice quality [27, 29]. A value of $C = 2$ ($T_{\min} = 10^{-4}\,\mathrm{cm}^2$) indicates a vocal tract *closure*, i.e. a configuration as needed for plosive consonants. Values of $C = 3$ ($T_{\min} = 0.15\,\mathrm{cm}^2$) and $C = 4$ ($T_{\min} = 0.25\,\mathrm{cm}^2$) indicate vocal tract constrictions that are necessary for fricatives and lateral sounds, respectively. For other values of $T_{\min}$, $C = 1$. Such states may sound vowel-like but with added frication noise. As a rule of thumb, such states often sound rather unnatural and are not particularly desirable. Another important technical observable is the *critical articulator* label, which VTL assigns to each segment of the tube area function, see Figure 2. These labels correspond to the tongue (T), lower incisors (I), lips (L) and other articulators (O), respectively. The large region covered by the tongue-related articulators was divided into eight segments (T_0 to T_7). Articulator tokens within a region of constriction are referred to as *place of articulation* (POA). The length of a constriction is measured in terms of number of tube segments within the region of constriction. Thereby, L_{C} denotes the length of a constriction (region with $T_{\mathbf{q}}(\chi) < 0.3\,\mathrm{cm}^2$), L_{TC} denotes the length of a tight constriction ($T_{\mathbf{q}}(\chi) \in [0.15, 0.25]\,\mathrm{cm}^2$) and L_{CC} denotes the length of a closure constriction ($T_{\mathbf{q}}(\chi) = 10^{-4}\,\mathrm{cm}^2$).

Generation of Vowel States

Vowel-related motor states were determined by unsupervised random exploration. During this process, a number of 10^4 open states were sampled from uniform distributions between the limits of the respective supraglottal VTL parameter ranges as defined in [29]. The velum opening parameter VO was always set to -0.1 (i.e. closed velum), because VO controls the nasality of a sound. However, during the vowel sampling phase nasality is not desired. Subsequently, VTL was used to compute the vocal tract transfer functions of the sampled motor states. The first two tube resonances f_{R1} and f_{R2} were extracted from each transfer function. These resonances are closely related to the formants f_1 and f_2 that would occur if a respective vowel state was synthesized. Hence, the scope of the explorable vowel space is well characterized by the f_{R1}-f_{R2} distribution, e.g. see [27]. Krug et al. have shown that randomly sampled states do not adopt a uniform density distribution in the f_{R1}-f_{R2} space, but that vowels of certain phonetic classes are overrepresented, while the corner vowels /o, u/ are underrepresented [27]. This is problematic because the central vowels can be generated by concatenating vowel states through a coarticulation model and thus, be learned in a subsequent step. Corner vowels, on the other hand, cannot be generated by interpolating central vowels. This means that an extraction of the corner vowels is necessary (and sufficient) for all vowels to be potentially present in the training material. For this purpose, the convex hull of the vowel distribution in f_{R1}-f_{R2} space was computed and the motor states whose tube resonances lied on the convex hull were stored. These data points were then removed from the distribu-

tion and the convex hull was computed again. This procedure was performed a total of five times, resulting in the extraction of 120 vowel states in which all edge and corner vowels were sufficiently represented.

Generation of Consonant States

Consonants are usually coarticulated differently depending on context. To capture plausible coarticulation in the synthetic speech, consonant-related motor states were generated in the context of the previously determined vowel states. For this purpose, the artificial vocal learning framework described by Krug et al. [29] was used. The system has been adapted, because the optimization of the motor states was based only on the tube area function in this case, i.e. no synthesis was required. The optimization was guided by two main principles: (i) A consonant state should be as similar as possible to a previously specified vowel-like state. (ii) The consonant state should have a constriction or closure at a specific location in the vocal tract. The corresponding objective function is the sum of a *similarity loss* $\mathcal{L}_S$ and a *constriction loss* $\mathcal{L}_C$. The former is the cosine similarity between a state to be optimized ($\mathbf{q}$) and a vowel reference state ($\mathbf{r}$). The latter is calculated from the sum of individual components $\mathcal{L}_{Ci}$. Thereby, the term $\mathcal{L}_{C0} = 100$ is applied if $C \neq C_{\mathrm{Ext}}$. This strongly penalizes shapes whose area function has a different constriction level than the externally set reference C_{Ext}. This ensures a quick convergence to the desired level of constriction. The term $\mathcal{L}_{C1} = 100$ is applied if $N_C \neq N_C^{\mathrm{Ext}}$. This penalizes shapes whose number of constrictions N_C is unequal to an externally set reference number N_C^{Ext}. Here, $N_C^{\mathrm{Ext}} = 1$, as shapes with multiple constrictions are often implausible and may lead to unnatural coarticulation.

$$\mathcal{L}_{C2} = \begin{cases} \mathcal{L}_I + \mathcal{L}_E + \hat{\mathcal{L}}_{C2} + \hat{\mathcal{L}}^{\star}_{C2}, & \text{if } N_{\mathrm{C}} \neq 0 \\ 50, & \text{if } N_{\mathrm{C}} = 0 \end{cases}$$

$$\hat{\mathcal{L}}_{C2} = \begin{cases} 25, & \text{if } L_{\mathrm{C}} \neq L_{\mathrm{C}}^{\mathrm{Ext}} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

$$\hat{\mathcal{L}}^{\star}_{C2} = \begin{cases} 25, & \text{if } C_{\mathrm{Ext}} = 2 \text{ and } L_{\mathrm{CC}} \neq L_{\mathrm{CC}}^{\mathrm{Ext}} \\ 25, & \text{if } C_{\mathrm{Ext}} \in [3,4] \text{ and } L_{\mathrm{TC}} \neq L_{\mathrm{TC}}^{\mathrm{Ext}} \\ 0, & \text{otherwise} \end{cases}$$

The term $\mathcal{L}_{C2}$ introduces an *articulator inclusion loss* $\mathcal{L}_I$ and an *exclusion loss* $\mathcal{L}_E$. These losses are computed from $1 - F_1$. To calculate the F_1 score in case of $\mathcal{L}_E$, articulator tokens within a constriction, which are equal to externally set reference tokens are counted as true positives, other cases are counted as false positives. In case of $\mathcal{L}_I$, externally set reference tokens that are equal to articulator tokens within a constriction are counted as true positives, other cases are counted as false positives. That means that $\mathcal{L}_I$ penalizes the absence of the desired tokens, while $\mathcal{L}_E$ penalizes the presence of undesired tokens. On the other hand, $\mathcal{L}_{C2}$ enforces specific constriction length values via the terms $\hat{\mathcal{L}}_{C2}$ and $\hat{\mathcal{L}}^{\star}_{C2}$. This is necessary because the similarity loss minimizes the tube length associated with the critical constriction. This is generally desirable, since too long constrictions are often biologically implausible [28]. However, too short critical constrictions lead to soft motor targets, e.g. the tongue barely touching the palate. This is problematic, as it means that the velocity of the critical articulator at the point of contact with the vocal tract walls is not maximal, but rather minimal. For example, a VCV sequence such as /ada/ may sound like /adZa/ because the articulators move too slowly through the vocal tract region where the constriction may cause frication noise. In this work, a length of $L_C^{\mathrm{Ext}} = L_{\mathrm{TC}}^{\mathrm{Ext}} = L_{\mathrm{CC}}^{\mathrm{Ext}} = 3$ was used. For each POA $\in$ [$\mathrm{T}_{2,3}$, $\mathrm{T}_{3,4}$, $\mathrm{T}_{4,5}$, $\mathrm{T}_{5,6}$, $\mathrm{T}_{6,7}$, T_3, T_4, T_5, T_6, T_7, L] and for each vowel state, the vocal learning framework was used to find 3 consonant motor states. In all cases the constriction constraint was set to $C_{\mathrm{Ext}} = 2$, except for POA = L, where both conditions $C_{\mathrm{Ext}} = 2$ and $C_{\mathrm{Ext}} = 3$ were used. This is because VTL has a specific mechanism that turns tongue-related closures into fricatives on the level of the tube area function if the tongue side elevation parameters adopt specific values. Thus, finding tongue-related closures is sufficient for the later generation of pseudo language. However, to turn a labial closure into a labial fricative, several parameters need to change simultaneously. It was easier to directly search for both the labial closures and fricatives via state optimization. A total number of 4320 consonant-related motor states were generated. Each optimization ran for 5000 steps using the same hyperparameters as in [29].

Monte-Carlo Generation of Pseudo Speech

Finally, the determined motor states were concatenated to form continuous pseudo speech signals. Utterances were generated as follows: For each utterance, a random number of motor state units (N_{MS}) was drawn, ranging from 1 to $N_{\mathrm{MS}}^{\max}$. Based on a 50 % chance, each unit was chosen to be a vowel or consonant (except for $N_{\mathrm{MS}} = 1$, where the unit was automatically declared to be a vowel). In the case of a consonant, it was chosen to be a closure, fricative, lateral, or nasal based on a 25 % chance. With a 50 % chance, closures or fricatives were chosen to be unvoiced. For vowels, one of the vowel motor states was selected; for consonants, one of the consonant states was selected. To create a nasal, the VO parameter of the corresponding shape was set to 0.5. To create fricatives for the front tongue-related configurations with POA $\in$ [$\mathrm{T}_{5,6}$, $\mathrm{T}_{6,7}$, T_5, T_6, T_7], the tongue tip side elevation parameter (TS3) was set to 1.0. For lateral generation, it was set to -1.0. For the posterior tongue-related configurations with POA $\in$ [$\mathrm{T}_{2,3}$, $\mathrm{T}_{3,4}$, $\mathrm{T}_{4,5}$, T_3, T_4], the corresponding central tongue-side parameter TS2 was set to 1.0 to generate fricatives. Laterals were not generated with these posterior configurations. To produce VTL compatible articulatory trajectories, the 19 dimensional (supraglottal) motor targets needed to be paired with 11 dimensional glottal targets. In case of voiced units the modal voice setting from the standard speaker model was used. For unvoiced sounds, the upper and lower displacements of the vocal folds (XT and XB) were changed from their modal values to 0.1 cm, the chink area (CA) was set to $0.1\,\mathrm{cm}^2$ and the relative amplitude (RA) was set to 0. A random duration value between 50 ms and 200 ms was sampled for each unit. However, while voiced units may have the same durations for the supraglottal and glottal parts of the motor targets, the durations for unvoiced units should be asynchronous to avoid implausible onset times or glottal artifacts [29]. For this reason, glottal onset times were shifted by $+50$ ms and -30 ms relative to the supraglottal onset times for targets following voiceless closures and fricatives, respectively. Finally, the target sequences were turned into time-dependent continuous articulatory trajectories by target interpolation via the Target-Approximation-Model [30, 31]. Thereby, a target time constant of 15 ms was used. The trajectories were discretized at a rate of 100 Hz and normalized to $[-1, 1]$. Thereby, $1.5 \cdot 10^5$ utterances were generated with $N_{\mathrm{MS}}^{\max} = 50$ and $5 \cdot 10^4$ utterances were generated with $N_{\mathrm{MS}}^{\max} = 20$, which means the synthetic data set had a total duration of approximately 150 h.

Experiments

First, the full data set (natural + synthetic) was separated into a training and validation part using a stratified split in the sense that 90 % and 10 % of the speech material from each speaker was in the training and validation part, respectively. Then, log-

melspectrograms were extracted and standardized feature-wise. The VQ-VAE was trained on the audio data set using the same loss functions as in [17] and a learning rate (LR) of 10^{-4}. Then, the M2L forward model was trained on the synthetic data successively for 25 epochs with LR's of $10^{-3}, 10^{-4}, 10^{-5}$ (drop every 10 epochs). Categorical-crossentropy was used as the loss between the model's output and the latent codebook indices (*latent loss)*, which were obtained from the VQ-VAE vector quantization layer. Further, an MSE loss was calculated between the input melspectrograms and the respective acoustic output, obtained from decoding the predicted latent codes using the frozen VQ-VAE decoder (*acoustic loss*). Then, the L2M model was trained on the synthetic data. First, it was trained successively for 5 and 5 epochs using LR's of 10^{-3} and 10^{-4}, respectively. Thereby, the MSE loss between the true and predicted motor trajectories (*motor loss*) was used. Subsequently, the model was trained for 20 epochs (LR of 10^{-4}) using the motor loss, the acoustic forward loss via the frozen M2L model plus VQ-VAE decoder as well as the corresponding latent loss. Finally, the model was tuned on the natural + synthetic data set for 6 epochs, applying the motor loss only in case of the synthetic samples and the latent loss in both cases. The acoustic loss was not used in this case. The P2L model was trained for 50 epochs (LR of 10^{-4}) on the BITS phoneme sequences using both the latent and acoustic losses. For the evaluation, the *Berlin* part of the KIEL set was used. These are 100 short sentences, each of which was uttered by 12 speakers (6 male/female). The natural samples were re-synthesized using following configurations: VQ-VAE logmel reconstruction with Hifi-Gan synthesis (VQV+H), L2M motor prediction with VTL synthesis (L2M+V), L2M motor prediction with audio reconstruction via M2L + VQ-VAE + Hifi-Gan synthesis (L2M+H). For evaluation, the 1200 sentences were also synthesized via the rule-base phoneme-to-speech functionality of VTL using the natural phoneme segments. The resulting motor trajectories were additionally synthesized with M2L + VQ-VAE + Hifi-Gan (M2L+H). The original phoneme segments were also synthesized using P2L with subsequent VTL synthesis. Resulting audio samples were evaluated in terms word- and character-error-rate (WER and CER), which were computed from the original text and transcripts as returned by the Google speech to text Web-API. Speech samples were also evaluated in terms of (extended) short time intelligibility (STOI [32] and ESTOI [33]).

3. Results

The experimental results are shown in Table 1. Apparently, the VQV model allows a relatively good reconstruction of the input signal. The fact that the error rates are somewhat higher than with the Hifi-Gan re-synthesis of the original signals may be due to the fact that a pre-trained Hifi-Gan was used, which was not conditioned on the reconstructed audio features. This gives the synthesized utterances a somewhat metallic sound, which has a negative effect on the naturalness of the speech. The mean Pearson correlation coefficient between motor trajectories predicted by the L2M model from natural utterances and the motor trajectories obtained with rule-based VTL synthesis is $\rho = 0.93 \pm 0.09$. Despite the high correlation, motor trajectories estimated from the natural utterances tend to be non-smooth, see Figure 3. This is because there is no loss during optimization that stops the inverse model from exploiting the forward model in a way to obtain the desired spectrograms. However, when such trajectories are synthesized with VTL, articulatory artifacts and noises occur frequently, since VTL does not simulate fast moving articulators very well. As a consequence, the WER values for L2M+V are rather high as this type of noisy speech may be challenging for ASR systems. Still,

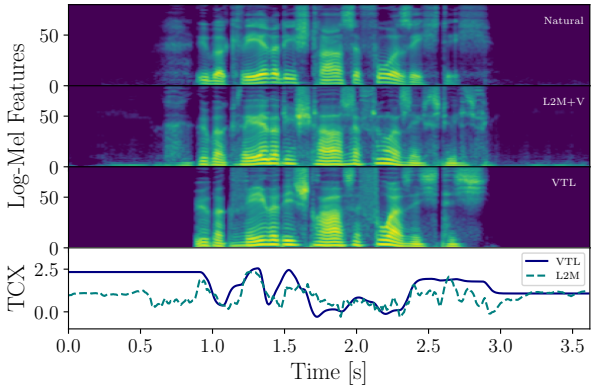

Figure 3: *The sentence "Überquere die Straße vorsichtig" (Cross the road carefully) as uttered by a male speaker from the KIEL data set (top). The upper and lower mid plots show the L2M+V imitation of the natural utterance and the VTL rule-based re-synthesis, respectively. The bottom plot shows the corresponding trajectories of the VTL parameter TCX.*

previous supervised attempts, e.g. by Gao et al. [11] are outperformed in terms of ASR accuracy. It was also observed that the P2L model was basically on a par with the rule-based VTL synthesis in terms of intelligibility.

Table 1: *Results on speech intelligibility. WER and CER values are separately reported for male (female) input data.* † *and* * *mean the values were computed using the natural samples and Hifi-Gan re-syntheses as references, respectively.*

	WER ↓	CER ↓	STOI ↑	ESTOI ↑
Natural	0.04 (0.03)	0.02 (0.02)		
Nat.+H	0.05 (0.03)	0.03 (0.02)		
VQV+H	0.08 (0.07)	0.04 (0.04)	$0.90^{\star}$	$0.81^{\star}$
L2M+V	0.56 (0.52)	0.37 (0.32)	$0.71^{\dagger}$	$0.51^{\dagger}$
L2M+H	0.24 (0.21)	0.12 (0.11)	$0.84^{\star}$	$0.73^{\star}$
└ V-ID	0.45 (0.37)	0.26 (0.21)	$0.78^{\star}$	$0.64^{\star}$
P2L+V	0.28 (0.25)	0.16 (0.14)	$0.68^{\dagger}$	$0.51^{\dagger}$
VTL	0.25 (0.25)	0.14 (0.12)	$0.63^{\dagger}$	$0.47^{\dagger}$
M2L+H	0.46 (0.35)	0.25 (0.19)	$0.66^{\star}$	$0.50^{\star}$
└ V-ID	0.53 (0.42)	0.31 (0.22)	$0.63^{\star}$	$0.45^{\star}$

4. Conclusion

This work proposes a self-supervised and scalable solution to the control problem of articulatory speech synthesis that enables speaker-independent acoustic-to-articulatory inversion, phoneme-to-articulatory conversion, and articulatory-to-acoustic neural synthesis. Due to the self-supervision, large amounts of data can potentially be obtained for training. It is expected that with larger networks and larger data sets, the forward direction can be better approximated, which would also improve the inverse control. However, future work should focus in particular on eliminating noise induced by fast moving articulators. This may involve a generative adversarial network that could discriminate between smooth and non-smooth movements. However, it may also require the improvement of the aero-acoustic simulation itself. In the supplementary material, examples are shown where the noises have been removed by post processing, which in some cases greatly increases the intelligibility and naturalness of the synthetic speech.

5. References

[1] P. K. Krug, S. Stone, and P. Birkholz, "Intelligibility and naturalness of articulatory synthesis with VocalTractLab compared to established speech synthesis technologies," in *Proc. SSW 11*, 2021, pp. 102–107.

[2] P. Birkholz, "Modeling consonant-vowel coarticulation for articulatory speech synthesis," *PloS ONE*, vol. 8, no. 4, p. e60603, 2013.

[3] P. K. Krug, B. Gerazov, D. R. van Niekerk, A. Xu, Y. Xu, and P. Birkholz, "Modelling microprosodic effects can lead to an audible improvement in articulatory synthesis," *J. Acoust. Soc. Am.*, vol. 150, no. 2, pp. 1209–1217, 2021.

[4] P. K. Krug, P. Birkholz, B. Gerazov, D. R. van Niekerk, A. Xu, and Y. Xu, "Articulatory synthesis for data augmentation in phoneme recognition," in *Proc. Interspeech*, 2022, pp. 1228–1232.

[5] P. Liu, Q. Yu, Z. Wu, S. Kang, H. Meng, and L. Cai, "A deep recurrent approach for acoustic-to-articulatory inversion," in *Proc. ICASSP*, 2015, pp. 4450–4454.

[6] A. Illa and P. K. Ghosh, "Representation learning using convolution neural network for acoustic-to-articulatory inversion," in *Proc. ICASSP*, 2019, pp. 5931–5935.

[7] G. Sivaraman, V. Mitra, H. Nam, M. Tiede, and C. Espy-Wilson, "Unsupervised speaker adaptation for speaker independent acoustic to articulatory speech inversion," *J. Acoust. Soc. Am.*, vol. 146, no. 1, pp. 316–329, 2019.

[8] V. Ribeiro and Y. Laprie, "Autoencoder-Based Tongue Shape Estimation During Continuous Speech," in *Proc. Interspeech*, 2022, pp. 86–90.

[9] M.-A. Georges, J. Diard, L. Girin, J.-L. Schwartz, and T. Hueber, "Repeat after me: self-supervised learning of acoustic-to-articulatory mapping by vocal imitation," in *Proc. ICASSP*, 2022, pp. 8252–8256.

[10] Y. Gao, S. Stone, and P. Birkholz, "Articulatory copy synthesis based on a genetic algorithm." in *Proc. Interspeech*, 2019, pp. 3770–3774.

[11] ——, "Articulatory copy synthesis using long-short term memory networks," *Studientexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung 2020*, pp. 52–59, 2020.

[12] K. Sering, P. Schmidt-Barbo, S. Otte, M. V. Butz, and H. Baayen, "Recurrent gradient-based motor inference for speech resynthesis with a vocal tract simulator," in *12th International Seminar on Speech Production*, 2020.

[13] Y. M. Siriwardena, A. A. Attia, G. Sivaraman, and C. Espy-Wilson, "Audio data augmentation for acoustic-to-articulatory speech inversion using bidirectional gated rnns," *arXiv preprint arXiv:2205.13086*, 2022.

[14] Y. M. Siriwardena, C. Espy-Wilson, and S. Shamma, "Learning to compute the articulatory representations of speech with the mirrornet," *arXiv preprint arXiv:2210.16454*, 2022.

[15] Y. Sun and X. Wu, "Embodied self-supervised learning by coordinated sampling and training," *arXiv preprint arXiv:2006.13350*, 2020.

[16] Y. Sun, Q. Huang, and X. Wu, "Unsupervised Acoustic-to-Articulatory Inversion with Variable Vocal Tract Anatomy," in *Proc. Interspeech*, 2022, pp. 4656–4660.

[17] A. Van Den Oord, O. Vinyals *et al.*, "Neural discrete representation learning," *Advances in neural information processing systems*, vol. 30, 2017.

[18] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.

[20] J. Kong, J. Kim, and J. Bae, "Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17 022–17 033, 2020.

[21] K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," *Syntax, Semantics and Structure in Statistical Translation*, p. 103, 2014.

[22] D. Hendrycks and K. Gimpel, "Bridging nonlinearities and stochastic regularizers with gaussian error linear units," *CoRR*, vol. abs/1606.08415, 2016. [Online]. Available: http://arxiv.org/abs/1606.08415

[23] V. Pratap, Q. Xu, A. Sriram, G. Synnaeve, and R. Collobert, "MLS: A Large-Scale Multilingual Dataset for Speech Research," in *Proc. Interspeech*, 2020, pp. 2757–2761.

[24] F. Schiel, C. Draxler, T. Ellbogen, K. Jänsch, and S. Schmidt, "Die BITS Sprachsynthesekorpora – Diphon-und Unit Selection-Synthesekorpora für das Deutsche," in *Proc. KONVENS*, 2006, pp. 121–124.

[25] K. J. Kohler, M. Pätzold, and A. P. Simpson, "From scenario to segment. the controlled elicitation, transcription, segmentation and labelling of spontaneous speech," *Arbeitsberichte des Instituts für Phonetik und digitale Sprachverarbeitung der Christian-Albrechts-Universität Kiel (AIPUK)*, vol. 29, 1995.

[26] ——, "From the acoustic data collection to a labelled speech data bank of spoken standard german," *Arbeitsberichte des Instituts für Phonetik und digitale Sprachverarbeitung der Universität Kiel (AIPUK)*, vol. 32, pp. 1–29, 1997.

[27] P. K. Krug, P. Birkholz, B. Gerazov, D. R. van Niekerk, A. Xu, and Y. Xu, "Efficient exploration of articulatory dimensions," *Studientexte zur Sprachkommunikation: Elektronische Sprachsignalverarbeitung 2022*, pp. 51–58, 2022.

[28] D. R. van Niekerk, A. Xu, B. Gerazov, P. K. Krug, P. Birkholz, L. Halliday, S. Prom-on, and Y. Xu, "Simulating vocal learning of spoken language: Beyond imitation," *Speech Commun.*, vol. 147, pp. 51–62, 2023.

[29] P. K. Krug, P. Birkholz, B. Gerazov, D. R. van Niekerk, A. Xu, and Y. Xu, "Artificial vocal learning guided by phoneme recognition and visual information," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1734–1744, 2023.

[30] Y. Xu and Q. E. Wang, "Pitch targets and their realization: Evidence from Mandarin Chinese," *Speech Commun.*, vol. 33, no. 4, pp. 319–337, 2001.

[31] S. Prom-On, Y. Xu, and B. Thipakorn, "Modeling tone and intonation in Mandarin and English as a process of target approximation," *J. Acoust. Soc. Am.*, vol. 125, no. 1, pp. 405–424, 2009.

[32] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *Proc. ICASSP*, 2010, pp. 4214–4217.

[33] ——, "An algorithm for intelligibility prediction of time–frequency weighted noisy speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 7, pp. 2125–2136, 2011.