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

Learning Joint Articulatory-Acoustic Representations with Normalizing Flows

Pramit Saha and Sidney Fels

Department of Electrical and Computer Engineering, University of British Columbia, Canada
pramit@ece.ubc.ca, ssfels@ece.ubc.ca

Abstract

The articulatory geometric configurations of the vocal tract and the acoustic properties of the resultant speech sound are considered to have a strong causal relationship. This paper aims at finding a joint latent representation between the articulatory and acoustic domain for vowel sounds via invertible neural network models, while simultaneously preserving the respective domain-specific features. Our model utilizes a convolutional autoencoder architecture and normalizing flow-based models to allow both forward and inverse mappings in a semi-supervised manner, between the mid-sagittal vocal tract geometry of a two degrees-of-freedom articulatory synthesizer with 1D acoustic wave model and the Mel-spectrogram representation of the synthesized speech sounds. Our approach achieves satisfactory performance in achieving both articulatory-to-acoustic as well as acoustic-to-articulatory mapping, thereby demonstrating our success in achieving a joint encoding of both the domains.

Index Terms: vocal tract geometry, normalizing flow, deep generative models, articulatory-to-acoustic mapping, pink trombone, speech formants.

1. Introduction

The vocal tract (VT) geometry [1, 2] and its dynamic changes play a crucial role in generating distinguishable speech sounds, via modulation of airflow and creation of various resonant cavities inside the tract. Therefore the acoustic signal contains abundant information that can be extracted for better understanding the underlying upper airway geometries involved in speech production mechanism. Determining the relationship between the articulatory gestures and acoustic parameters has been a long-standing issue in related research areas [2–7].

The aforementioned problem is generally classified into two broad types : (a) articulatory-to-acoustic mapping, also known as forward mapping problem and (b) acoustic-to-articulatory inversion or speech inversion, also known as inverse mapping problem . The former deals with the production of audio speech signals from vocal tract, thereby modeling variations in acoustic space with variations in vocal tract shapes, while, the latter one encompasses the recovery of vocal tract configurations responsible for production of given speech signals. The forward mapping, *i.e.*, estimating acoustic response to articulatory behaviour is of utmost importance in the development of articulatory speech synthesizers and other silent speech interfaces as well as in detailed study of speech production and articulatory phonetics. On the other hand, applications of the inverse mapping, *i.e.*, inferring articulatory information from speech acoustics include estimation of vocal tract parameters for efficient speech coding, enhanced speech recognition systems and for developing visual articulatory feedback systems. The related works mostly address either of these two problems independently and as such, there is a lack of unified end-to-end forward and inverse mapping approach that can reversibly map the vocal tract shapes and corresponding speech sounds. This is because it is incredibly challenging to accurately determine a joint distribution of the articulatory and acoustic domains, both having complex generative processes involving a series of motor control and estimation tasks, biomechanical mechanisms and aero-dynamic flow - some being shared across both generations while some being specifically important to one of them.

In order to address this issue, we employ a semi-supervised, invertible, bijective cross-domain mapping between vocal tract geometries and the acoustic outputs, leveraging a pair of deep convolutional autoencoders and normalizing flow based probability density estimation technique. In this paper, we particularly consider the mid-sagittal vocal tract configurations and synthesized vowel sounds, simulated in the online articulatory speech synthesizer application named Pink Trombone [8], as our input-output space. Our approach involves a separated yet shared encoding of the images, capturing diverse vocal tract shape, as well as the mel-spectrograms, possessing the acoustic information pertaining to the resultant speech signals, in an unsupervised manner. The double autoencoders are simultaneously aligned in a supervised fashion by stacking a chain of invertible bijective transformation functions between the bottleneck feature distributions. The core idea is to constrain the latent representations of both the domains to have some domain-specific features pertaining to self-reconstruction as well as a joint feature space that encodes the mutual characteristics for enabling cross domain VT geometry-to-speech and speech-to-VT geometry synthesis. Furthermore, the domain-specific latent codes is kept conditional on the shared cross-domain latent space by enforcing a normalizing flow [9] based conditional prior in the articulatory-acoustic latent representation. In the next section, we will lay the foundation of our approach and present a systematic study on the variational model employed to achieve the target mapping.

2. Proposed Mapping Strategy

2.1. Problem formulation and overview

In order to investigate the joint distribution of vocal tract shapes and acoustics $p(x_g, x_s)$ which follow the generative processes $p_g(x_g)$ and $p_s(x_s)$ respectively, we define a common latent variable z such that the marginal likelihood $p(x_g, x_s) = \int p(x_g, x_s, z)\ dz$, where the joint probability distribution $p(x_g, x_s, z) = p(x_g, x_s|z)p(z)$. The likelihood $p(x_g, x_s|z)$ indicates the probability distribution over the observed variables in articulatory and acoustic space, given the latent representation z. The standard practise is to compute the likelihood using the posterior distribution $p(z|x_g, x_s)$ via Bayes' rule. However computing the posterior distribution is intractable in general as there exists no closed form solution. Alternatively, a variational distribution $\Psi(z|x_g, x_s)$ is used to approximate the posteriori by optimizing the evidence lower bound (ELBO) [10]. Therefore in our case, the maximization of the likelihood of

http://dx.doi.org/10.21437/Interspeech.2020-2004

$p(x_g, x_s)$ can be achieved by involving a posterior encoding distribution $\Psi_\phi(z|x_g, x_s)$ parameterized by ϕ. As such, our objective boils down to learning the variational posterior distribution $\Psi_\phi(z_{\widehat{gs}}|x_g, x_s)$ related to the shared latent space ($z_{\widehat{gs}}$) between the VT geometry (x_g) and the acoustic representation (x_s). Considering that the shared latent variable is capable of encoding joint articulatory and acoustic information, this implies, for a given pair of articulatory-acoustic data sample (x_g, x_s), the learnt posterior distributions, $\Psi_\phi(z_{\widehat{gs}}|x_g, x_s) \equiv \Psi_\phi(z_{\widehat{gs}}|x_g) \equiv \Psi_\phi(z_{\widehat{gs}}|x_s)$. This can be ensured by enforcing the encoders in both the domains to generate same latent information. However, since the data distribution of articulatory and acoustic domains follow distinct underlying generative models as discussed earlier, it is not admissible to try to enforce exactly same latent variable representation by removing individual domain-specific information from the acoustic or articulatory space. For the same reason, minimizing the mean squared error between the encoder output features for enhancing shared information is not ideal.

To this end, the shared information encoding is modified in two ways. The first is to partition the encodings of both the articulatory and acoustic domains into two parts: one that contains sole articulatory or acoustic information ($z_g \setminus z_{\widehat{gs}}$ or $z_s \setminus z_{\widehat{gs}}$) and the other which has the joint or shared information ($z_{\widehat{gs}}$). Therefore, our model consists of two domain-specific encoders : one for encoding the vocal tract geometry from the input image that learns an articulation-related posterior distribution $\Psi_\gamma(z_g \setminus z_{\widehat{gs}}|x_g, x_s)$ parameterized by γ and the other for encoding the acoustic information from the mel spectrograms, that learns the latent posterior distribution of acoustic domain $\Psi_\alpha(z_s \setminus z_{\widehat{gs}}|x_g, x_s)$ parameterized by α. The second modification is that, instead of constraining the encoders to learn the exact same shared latent encoding dimensions, we respect the constraints specific to articulation or acoustics and alternatively learn an invertible bijective mapping $\Omega_{\omega_{\widehat{gs}}} : \mathbb{R}^{d_{\widehat{gs}}} \rightarrow \mathbb{R}^{d_{\widehat{gs}}}$ between the shared representation of the articulatory and acoustic space. This invertible mapping performs transformation of the $d_{\widehat{gs}}$ dimensional latent vector $z_{\widehat{gs}}$ between the domains g and s.

2.2. Self-attention based Convolutional Autoencoder

The input-output space of our problem being artificial vocal tract images and mel-spectrograms, both are in the image representation. And our target is to encode the pair of image data into 2 sets of effective latent vectors or bottleneck features which best represent the respective domains and contain maximum relevant information required for their individual reconstruction. Convolutional architecture is a natural choice for the autoencoder network in this case as the convolutional autoencoders preserve the spatial information of the input image data, by incorporation of convolutional filter kernels in the network [11]. Additionally, the self-attention mechanism of [12–14] is also utilized in the encoder-decoder architecture as shown in Fig. 1, leveraging its capability of modeling non-local relationships between widely separated spatial regions - an equivalent of long-range dependency in images. An attention map β is generated by first transforming the feature set from the previous layer into two parallel layers $f(x)$ and $g(x)$ followed by exponentiating the product of these two feature sets and normalizing it as shown below:

$$f(x) = W_f x, \;\; x \in \mathbf{R}^{C\times N}, \; W_f \in \mathbf{R}^{C\times C} \tag{1}$$

$$g(x) = W_g x, \; W_g \in \mathbf{R}^{C\times C} \tag{2}$$

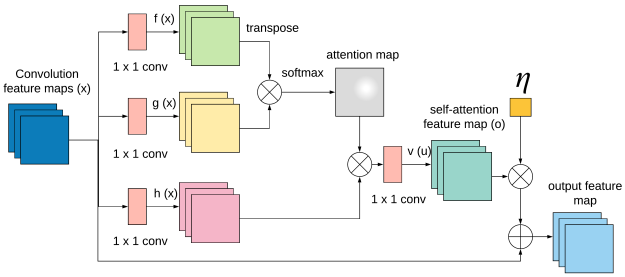

Figure 1: *The self-attention module in convolutional autoencoder architecture*

$$\beta_{j,i} = \frac{f(x_i)^T g(x_j)}{\sum_{i=1}^{N} f(x_i)^T g(x_j)}, \; \beta \in \mathbf{R}^{N\times N} \tag{3}$$

where $\beta_{j,i}$ denotes the impact of i^{th} location while rendering j^{th} location, *i.e.*, the extent to which the network attends to i^{th} location while synthesizing j^{th} location.

Next, the previous layer features are again transformed to another feature set $h(x)$ and multiplied with the computed attention map β to generate the self attention map output o.

$$h(x) = W_h x, \; W_h \in \mathbf{R}^{C\times C} \tag{4}$$

$$v(u_i) = W_v u_i \tag{5}$$

$$o_j = v\left(\sum_{i=1}^{N} \beta_{j,i}\, h(x_i)\right), \;\; o \in \mathbf{R}^{C\times N} \tag{6}$$

W_f, W_g, W_h and W_v are learned weight matrices, implemented as 1×1 convolution operation. The final layer of the self-attention convolution layer is represented as the addition of a weighted self-attention mask (with the learnable scalar weight, η) to the previous layer feature.

$$y_j = \eta\, o_j + x_j, \;\; y \in \mathbf{R}^{C\times N} \tag{7}$$

η is initialized as 0 to let the model explore local spatial information before starting to capture non-local features via self-attention based refinement.

Let the encoder networks corresponding to the vocal tract geometry x_g of dimensions d_g and the acoustic representation x_s of dimensions d_s be denoted as $\mathcal{G}_{\mathcal{E}_g}$ and $\mathcal{S}_{\mathcal{E}_s}$ with parameters $\mathcal{E}_g$ and $\mathcal{E}_s$ respectively, such that $\mathcal{G}_{\mathcal{E}_g} : (x_g)_{d_g} \rightarrow (z_g)_{d_g^l}$ with $\mathcal{E}_g = \{\beta, \gamma\}$ and $\mathcal{S}_{\mathcal{E}_s} : (x_s)_{d_s} \rightarrow (z_s)_{d_s^l}$ with $\mathcal{E}_s = \{\alpha, \gamma\}$, where d_g^l and d_s^l are the latent dimensions of articulatory and acoustic domains. Similarly, let the decoder networks corresponding to the vocal tract geometry and the acoustic representation be denoted as $\mathcal{G}_{\mathcal{D}_g}$ and $\mathcal{S}_{\mathcal{D}_s}$ with parameters $\mathcal{D}_g$ and $\mathcal{D}_s$ respectively, such that $\mathcal{G}_{\mathcal{D}_g} : (z_g)_{d_g^l} \rightarrow (x_g)_{d_g}$ and $\mathcal{S}_{\mathcal{D}_s} : (z_s)_{d_s^l} \rightarrow (x_s)_{d_s}$. Further, let the decoded vocal tract geometry and acoustic representation outputs be denoted as $\tilde{x}_g$ and $\tilde{x}_s$ respectively, then, $\tilde{x}_g = \mathcal{G}_{\mathcal{D}_g}(\mathcal{G}_{\mathcal{E}_g}(x_g))$ and $\tilde{x}_s = \mathcal{S}_{\mathcal{D}_s}(\mathcal{S}_{\mathcal{E}_s}(x_s))$. For vocal tract geometry image, the reconstruction loss between input image (x_g) and reconstructed image $(\tilde{x}_g)$ from the image decoder is computed as $\mathcal{L}_g^{rec}(x_g, \tilde{x}_g) = \left\|x_g - \mathcal{G}_{\mathcal{D}_g}(\mathcal{G}_{\mathcal{E}_g}(x_g))\right\|$ where $\|.\|$ denotes l_2 norm. Similarly, for Mel-spectrogram image, the reconstruction loss between input Mel-spectrogram (x_s) and reconstructed Mel-spectrogram $(\tilde{x}_s)$ from the spectrogram decoder is computed as $\mathcal{L}_s^{rec}(x_s, \tilde{x}_s) = \|x_s - \mathcal{S}_{\mathcal{D}_s}(\mathcal{S}_{\mathcal{E}_s}(x_s))\|$.

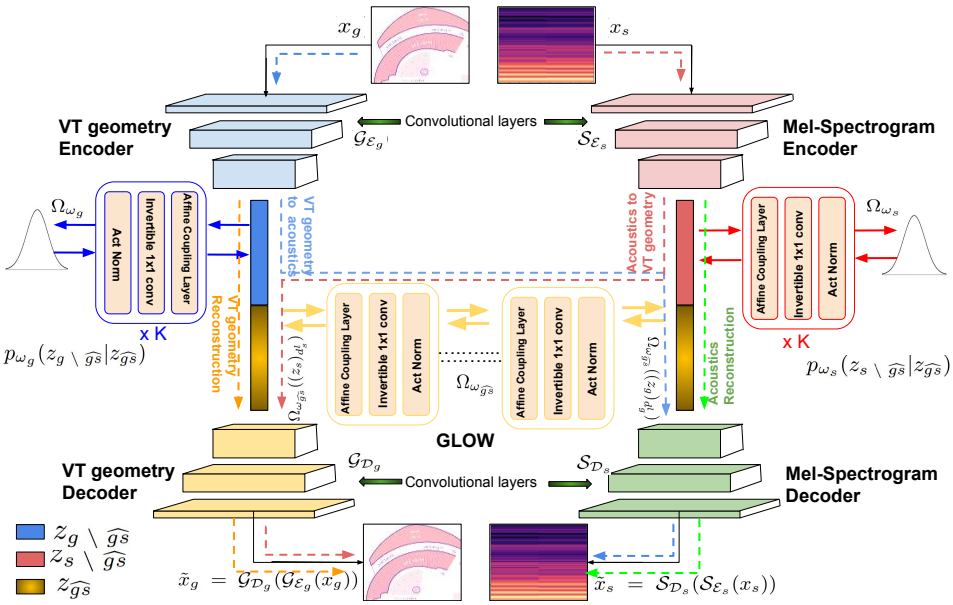

Figure 2: *The proposed articulatory-acoustic forward and inverse mapping*

2.3. Optimization of marginal likelihood

Since our entire latent variable representation is partitioned to three major components as discussed in section 2.1, accordingly, our posterior distribution gets factorized as follows:

$$\Psi_\phi(z|x_g, x_s) = \Psi_\gamma(z_g \setminus \widehat{gs}|x_g, z_{\widehat{gs}})\Psi_\alpha(z_s \setminus \widehat{gs}|x_s, z_{\widehat{gs}}) \Psi_\beta(z_{\widehat{gs}}|x_g, x_s) \quad (8)$$

Similarly, assuming the conditional independence of the latent codes, our prior probability distribution gets factorized as:

$$p(z) = p(z_{\widehat{gs}}, z_g \setminus \widehat{gs}, z_s \setminus \widehat{gs}) = p(z_g \setminus \widehat{gs}|z_{\widehat{gs}})\, p(z_s \setminus \widehat{gs}|z_{\widehat{gs}})\, p(z_{\widehat{gs}}) \quad (9)$$

The computation of the optimal likelihood requires the marginalization of the latent variable, which is potentially challenging to compute. Instead, we optimize the lower bound over the encoding distribution using the standard procedure, as follows:

$$\log p\,(x_g, x_s) = \log\left(\sum \Psi_\phi(z|x_g, x_s)\frac{p\,(x_g, x_s, z)}{\Psi_\phi(z|x_g, x_s)}\right) \quad (10)$$

Now using Jensen's inequality,

$$\begin{aligned}\log p\,(x_g, x_s) &\geqslant \sum \Psi_\phi(z|x_g, x_s)\log\frac{p\,(x_g, x_s, z)}{\Psi_\phi(z|x_g, x_s)} \\ &= \sum \Psi_\phi(z|x_g, x_s)\log\frac{p\,(x_g, x_s|z)p(z)}{\Psi_\phi(z|x_g, x_s)} \\ &\geqslant \sum \Psi_\phi(z|x_g, x_s)[\log p\,(x_g, x_s|z) + \log p(z) \\ &\quad - \log \Psi_\phi(z|x_g, x_s)] \quad (11)\end{aligned}$$

With the help of Equation (8) and (9), the first term or data-likelihood term in Equation (11), can be further simplified as:

$$\begin{aligned}\Psi_\phi(z|x_g, x_s)\log p\,(x_g, x_s|z) = \Psi_\gamma(z_g \setminus \widehat{gs}|x_g, z_{\widehat{gs}})\times \\ \Psi_\beta(z_{\widehat{gs}}|x_g, x_s)\log p(x_g|z_{\widehat{gs}}, z_g \setminus \widehat{gs}) + \Psi_\alpha(z_s \setminus \widehat{gs}|x_s, z_{\widehat{gs}})\times \\ \Psi_\beta(z_{\widehat{gs}}|x_g, x_s)\log p(x_s|z_{\widehat{gs}}, z_s \setminus \widehat{gs}) \quad (12)\end{aligned}$$

Similarly, the second term can be further simplified as:

$$\begin{aligned}\Psi_\phi(z|x_g, x_s)\log p(z) = \Psi_\beta(z_{\widehat{gs}}|x_g, x_s)\log p(z_{\widehat{gs}}) + \\ \Psi_\gamma(z_g \setminus \widehat{gs}|x_g, z_{\widehat{gs}})\log p(z_g \setminus \widehat{gs}|z_{\widehat{gs}}) + \Psi_\alpha(z_s \setminus \widehat{gs}|x_s, z_{\widehat{gs}})\times \\ \log p(z_s \setminus \widehat{gs}|z_{\widehat{gs}}). \quad (13)\end{aligned}$$

And the third term in equation (11) can be simplified as:

$$\begin{aligned}-\Psi_\phi(z|x_g, x_s)\log\Psi_\phi(z|x_g, x_s) = -\Psi_\beta(z_{\widehat{gs}}|x_g, x_s)\times \\ \log\Psi_\beta(z_{\widehat{gs}}|x_g, x_s) - \Psi_\gamma(z_g \setminus \widehat{gs}|x_g, z_{\widehat{gs}})\log\Psi_\gamma(z_g \setminus \widehat{gs}|x_g, z_{\widehat{gs}}) \\ - \Psi_\alpha(z_s \setminus \widehat{gs}|x_s, z_{\widehat{gs}})\log\Psi_\alpha(z_s \setminus \widehat{gs}|x_s, z_{\widehat{gs}}). \quad (14)\end{aligned}$$

Therefore our task is now to maximize the lower bound [10, 15–17] obtained by plugging the expressions of Equations (12), (13) and (14) in Equation (11).

2.4. Normalizing flow

Normalizing flow [9, 16–20] is a flow-based generative model and is used as a powerful probability density estimator. It is constructed by stacking a sequence of invertible transformation functions which transform a simple distribution into a complex one and eventually learns an explicit data distribution $p(x)$. The probability distribution of the final target variable is obtained by substituting the variables for a new one, flowing through a chain of transformations f_i, following the change of variables theorem. Given an initial distribution z_0, the output x can be obtained by using a series of probability density functions in a step-by-step fashion.

$$x = z_K = f_K(f_{K-1}(f_{K-2}(f_{K-3}(....(f_3(f_2(f_1(z_0))))....)))) \quad (15)$$

Using the change of variables rule, the probability density function of the model can therefore be written as follows:

$$\begin{aligned}\log\ p(x) = \log\pi_K(z_K) = \log\ \pi_{K-1}(z_{K-1}) - \log\left|det\frac{df_K}{dz_{K-1}}\right| \\ = \log\ \pi_0(z_0) - \sum_{i=1}^{K}\log\left|det\frac{df_i}{dz_{i-1}}\right| \quad (16)\end{aligned}$$

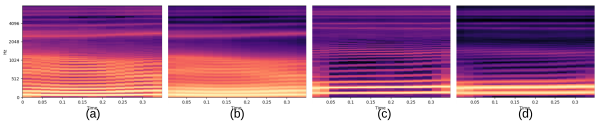

Figure 3: *(a) and (c) respectively shows the mel-spectrogram corresponding to the original vowels /a/ and /u/, (b) and (d) respectively shows their synthesized versions from VT geometry*

The sequence formed by successive distributions π_i is known as normalized flow. Both the conditional priors $\Omega_{\omega_g} = \Psi_\gamma(z_{g \setminus \widehat{gs}}|x_g, z_{\widehat{gs}})$ and $\Omega_{\omega_s} = \Psi_\alpha(z_{s \setminus \widehat{gs}}|x_s, z_{\widehat{gs}})$ as well as the mapping between the shared latent codes $\Omega_{\omega_{\widehat{gs}}}$ are modeled with GLOW [9], a normalizing flow based generative model using invertible 1×1 convolutions. A single step of GLOW involves three substeps - activation normalization (act-norm), invertible 1×1 convolution and an affine coupling layer. The act-norm is an affine transformation using trainable parameters - scale (s) and bias (b) per channel, similar to batch normalization, except that it works for a mini-batch size 1. The transformation for a k^{th} layer can be expressed as $y_{(k)\ i,j} = s \odot z_{(k)\ i,j} + b_{(k)}$. Next, 1×1 convolution with equal input and output dimensions is a generalized way of permuting channel ordering between layers of flow, thereby ensuring that the ordering of channels is shuffled for the flow to act on the entire data sample. Assuming the weight matrix to be W: $[c \times c]$, where c is the number of channels, this step can be written as $v_{(k)\ i,j} = W y_{(k)\ i,j}$. The last substep consists of an affine coupling layer where the convolved outputs $v_{(k)}$ are split into two parts: $v_{(k)\ a}$ and $v_{(k)\ b}$, out of which $(v_{(k)\ a})$ remains the same where as the other part $(v_{(k)\ b})$ undergoes an affine transformation involving scaling $(s(.))$ and translation $(t(.))$. This can be denoted as: $v_{(k)\ a}, v_{(k)\ b} = split(v_{(k)})$, $(log\ s, t) = NN(v_{(k)\ b})$, $u_{(k)\ a} = v_{(k)\ a}$, $u_{(k)\ b} = exp(log\ s) \odot v_{(k)\ b} + t$, $z_{(k+1)} = concat(u_{(k)\ a}, u_{(k)\ b})$.

As shown in Fig 2, the mapping between the shared latent components $\Omega_{\omega_{\widehat{gs}}}$ is achieved using a sequence of such transformations. The cost of mapping the latent space of VT geometry image to that of mel-spectrogram $\mathcal{L}_{g2s}(x_g, x_s)$ is defined as mean squared error between the encoded spectrogram representation $(z_s)_{d_s^l}$ and transformed image representation $\Omega_{\omega_{\widehat{gs}}}((z_g)_{d_g^l})$. Similarly, the cost of mapping the latent space of mel-spectrogram to that of VT geometry image $\mathcal{L}_{s2g}(x_s, x_g)$ is defined as mean squared error between the encoded VT geometry representation $(z_g)_{d_g^l}$ and transformed spectrogram representation $\Omega_{\omega_{\widehat{gs}}}((z_s)_{d_s^l})$.

3. Experiments and Results

3.1. Dataset and Training

We varied the pink trombone tongue controller[1] that changes the VT shape and correspondingly captured videos of pink trombone VT with frame rate of 30 fps and audios at a sampling rate of 22,020 Hz. Our model was implemented in PyTorch and we converted the audio into mel-spectrograms using Librosa [21]. We randomly shuffled and partitioned the data (36,081 audios and images extracted from the videos) into train (80%), development (10%) and test sets (10%). The images were downsampled to dimensions $90 \times 98 \times 3$ to reduce the computational time. The network was trained with a batch size of 10 on NVIDIA GeForce GTX 1080 Ti GPU. The loss function was optimized using Adam with a learning rate of .0001 for a total of 200 epochs. In order to mitigate the problem of overfitting, Batch Normalization was used after every convolutional layer and before applying non-linearity.

[1] https://dood.al/pinktrombone/

Figure 4: *The synthesized pink trombone images corresponding to VT configurations for /a/, /ae/, /i/ and /u/ (left to right)*

3.2. Qualitative and quantitative performance analysis

The original and synthesized Mel-Spectrograms of the vowels /a/ and /u/ corresponding to the respective VT shapes have been shown in Fig 3. A qualitative analysis of the figure demonstrates that although the generated mel-spectrograms are blurrier than the original crisp mel-spectrograms, they are indeed recognizable and significantly similar to the ground truth data. In order to quantitatively evaluate the performance of the proposed method in acoustic domain, we further computed the average formant frequencies of the synthesized audio signal and the original audio signal after recovering the synthesized audio with Griffin-Lim based spectrogram inversion method [22]. The mean error of the first three formants of synthesized vowels w.r.t the original vowels are $18.57\%, 24.21\%, 7.69\%$ respectively. The synthesized pink trombone images corresponding to the cardinal vowels $/a/$, $/ae/$, $/i/$ and $/u/$ have been presented in Fig. 4. The generated images are found to be quite similar to the actual VT geometries of pink trombone corresponding to respective vowels. It shows that our model is able to properly recognize the VT shape changes with changes in acoustic input, thereby demonstrating the success of our approach. The mean absolute error between the normalized synthesized pink trombone images and original images is 0.0397, most of which evidently comes from the non-VT part.

4. Conclusions and future works

In this paper, we have developed a one-to-one invertible mapping between the articulatory and acoustic spaces for an online articulatory speech synthesizer application named pink trombone. To the best of our knowledge, this is the first attempt to study an invertible joint articulatory-acoustic representation utilizing the best of deep autoencoder architectures and normalizing flow based techniques. This can be extended to one-to-many or many-to-one scenarios by introducing variational autoencoder architecture which generates a vector of means and standard deviations of the Gaussian distributions as the latent codes and will be addressed in future works. Besides, this work investigates a joint articulatory-acoustic representation for static vowels only, as we are considering VT input image for a particular instant. This can be further extended to continuous vowel spaces by including a sequence of images reflecting the dynamic VT shape changes with time in the articulatory space.

5. Acknowledgements

This work was funded by the Natural Sciences and Engineering Research Council (NSERC) of Canada and Canadian Institutes for Health Research (CIHR).

6. References

[1] B. H. Story, I. R. Titze, and E. A. Hoffman, "Vocal tract area functions from magnetic resonance imaging," *The Journal of the Acoustical Society of America*, vol. 100, no. 1, pp. 537–554, 1996.

[2] P. Saha, P. Srungarapu, and S. Fels, "Towards automatic speech identification from vocal tract shape dynamics in real-time mri," *arXiv preprint arXiv:1807.11089*, 2018.

[3] V. Mitra, G. Sivaraman, C. Bartels, H. Nam, W. Wang, C. Espy-Wilson, D. Vergyri, and H. Franco, "Joint modeling of articulatory and acoustic spaces for continuous speech recognition tasks," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5205–5209.

[4] T. Hueber, E.-L. Benaroya, B. Denby, and G. Chollet, "Statistical mapping between articulatory and acoustic data for an ultrasound-based silent speech interface," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[5] S. Hiroya and M. Honda, "Estimation of articulatory movements from speech acoustics using an hmm-based speech production model," *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 2, pp. 175–185, 2004.

[6] J. Hodgen and P. Valdez, "A stochastic articulatory-to-acoustic mapping as a basis for speech recognition," in *IMTC 2001. Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference. Rediscovering Measurement in the Age of Informatics (Cat. No. 01CH 37188)*, vol. 2. IEEE, 2001, pp. 1105–1110.

[7] P. Saha, Y. Liu, B. Gick, and S. Fels, "Ultra2speech–a deep learning framework for formant frequency estimation and tracking from ultrasound tongue images," *arXiv preprint arXiv:2006.16367*, 2020.

[8] N. Thapen, "Pink Trombone," https://dood.al/pinktrombone/, 2017, version 1.1.

[9] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 215–10 224.

[10] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[11] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," in *International conference on artificial neural networks*. Springer, 2011, pp. 52–59.

[12] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7794–7803.

[13] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," *arXiv preprint arXiv:1805.08318*, 2018.

[14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[15] D. G. Tzikas, A. C. Likas, and N. P. Galatsanos, "The variational approximation for bayesian inference," *IEEE Signal Processing Magazine*, vol. 25, no. 6, pp. 131–146, 2008.

[16] C.-W. Huang, L. Dinh, and A. Courville, "Augmented normalizing flows: Bridging the gap between generative flows and latent variable models," *arXiv preprint arXiv:2002.07101*, 2020.

[17] S. Mahajan, I. Gurevych, and S. Roth, "Latent normalizing flows for many-to-many cross-domain mappings," *arXiv preprint arXiv:2002.06661*, 2020.

[18] D. J. Rezende and S. Mohamed, "Variational inference with normalizing flows," *arXiv preprint arXiv:1505.05770*, 2015.

[19] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp," *arXiv preprint arXiv:1605.08803*, 2016.

[20] L. Dinh, D. Krueger, and Y. Bengio, "Nice: Non-linear independent components estimation," *arXiv preprint arXiv:1410.8516*, 2014.

[21] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, vol. 8, 2015.

[22] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, pp. 236–243, 1984.