A DEEP LEARNING APPROACH FOR DATA DRIVEN VOCAL TRACT AREA FUNCTION ESTIMATION

Sasan Asadiabadi, Engin Erzin

Multimedia, Vision and Graphics Laboratory,
College of Engineering, Koç University, Istanbul, Turkey
E-mail: [sabadi15, eerzin]@ku.edu.tr

ABSTRACT

In this paper we present a data driven vocal tract area function (VTAF) estimation using Deep Neural Networks (DNN). We approach the VTAF estimation problem based on sequence to sequence learning neural networks, where regression over a sliding window is used to learn arbitrary non-linear one-to-many mapping from the input feature sequence to the target articulatory sequence. We propose two schemes for efficient estimation of the VTAF; (1) a *direct* estimation of the area function values and (2) an *indirect* estimation via predicting the vocal tract boundaries. We consider acoustic speech and phone sequence as two possible input modalities for the DNN estimators. Experimental evaluations are performed over a large data comprising acoustic and phonetic features with parallel articulatory information from the USC-TIMIT database. Our results show that the proposed direct and indirect schemes perform the VTAF estimation with mean absolute error (MAE) rates lower than 1.65 mm, where the direct estimation scheme is observed to perform better than the indirect scheme.

Index Terms— Speech Articulation, Vocal Tract Area Function, Deep Neural Network, Convolutional Neural Network

1. INTRODUCTION

Vocal tract (VT), starting from the larynx to the lips, is the most important organ in the human speech production system. Given the excitation signal at the larynx, various configurations of the VT result in different spoken sounds. VTAF estimation is of great importance in various applications, such as speech synthesis and automatic speech recognition, specially with spontaneous or pathological speech.

Approaches to vocal tract area function estimation can be broadly classified into two categories: image processing based which generally use MRI frames of the VT to extract the area function and speech processing based that use acoustic speech signals to estimate the VTAF. For a successful image-based VTAF estimation, obtaining a full or semi-automatic vocal tract boundary segmentation is required. Intensive research has been conducted on the automatic vocal tract boundary estimation in the real-time MRI data [1, 2, 3, 4]. Upon estimating the VT boundaries, the VTAF could be calculated in a number of ways. Given the vocal tract boundaries, [4] carried out a recursive bisection algorithm to find the vocal tract midline. The area function was then calculated by finding the distance between the intersection points of perpendicular to the midline and the VT boundaries. In another approach, [2] defined a set of standard grid lines along the tract and calculated the VTAF as the distance between the grid and VT boundaries intersection points, at each grid line.

An image-based VTAF estimation may not seem practical at a first look, but such studies are mainly performed and are necessary for the speech-based approaches. To perform the one-to-many acoustic to articulatory mapping efficiently, in the speech-based approaches, acquiring a large audio-to-articulatory dataset is required, which is attainable via the image-based estimations. A direct estimation of the articulatory by inverse filtering of the acoustic speech was proposed in [5]. Another VTAF estimation approach was carried out using particle swarm optimization [6]. Later multimodal articulatory inversion using acoustic and visual data was proposed [7]. Using DNNs and RNNs for articulatory-to-acoustic conversion or acoustic-to-articulatory inversion problems has gained attraction in the recent years [8, 9]. Such approaches have taken over the traditional GMM or HMM based articulatory inversion methods [10, 11]. Siding window-based sequence to sequence prediction methods using deep neural networks such as algorithms proposed in [12, 13], provide an effective framework for articulatory estimation problem.

In this work, we refer to the cross sectional *distance* (in mm) between the lower and upper vocal tract boundaries, from lips to the larynx, as the area function. To our knowledge the proposed method is the first approach to directly estimate the vocal tract area function from phone sequence, unlike the previous approaches, which commonly utilize acoustic features. We introduce a novel method to efficiently estimate the VTAF from the acoustic and phone data. Looking at the area function estimation task as a sequence to sequence mapping problem, we utilize a deep learning overlapping sliding window regressor, inspired by [13]. We model the VTAF estimation task as a multivariate regression problem. The sliding window regressor approach learns arbitrary non-linear one-to-many mapping from the input feature sequence to the output articulatory sequence. We propose a direct and an indirect approach for the area function estimation task. In the direct estimation scheme, the VTAF values along the tract are directly estimated at the output of the trained networks, whereas in the indirect scheme, a DNN is trained to estimate the VT boundaries and the VTAF is then calculated from the estimated contours. We compare the performance of the direct estimation to indirect scheme, for acoustic and phonetic-based trained networks.

The rest of this paper is organized as follows. In section 2 we describe the training dataset preparation and proposed speech and phone-driven VTAF estimation systems. Experimental evaluations are given in Section 3. Finally, the article is concluded in Section 4.

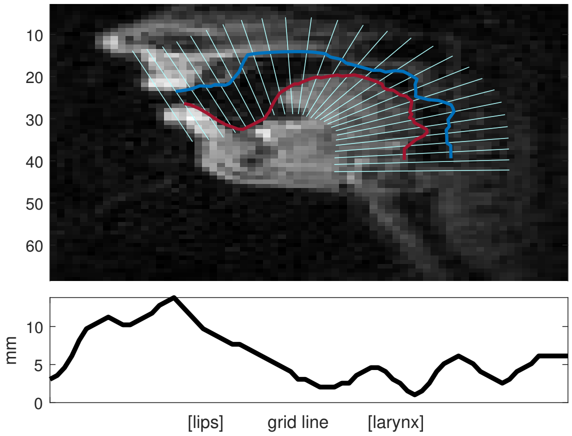


Fig. 1: An example of articulatory data extraction. Above: midsagittal view of a MRI frame. Red: lower VT boundary, Blue: upper VT boundary, Cyan: grid lines from lips to larynx. Below: VTAF along the tract, calculated on each grid line.

2. METHODOLOGY

2.1. Dataset

In this work, the USC-TIMIT database [14] is used to train a DNN for vocal tract area function estimation. The database comprises midsagittal MRI videos of 10 speakers, recorded at a frame rate of 23.13 fps and a spatial resolution of 68 × 68 pixels over 20 × 20 cm (approximately 2.9 mm pixel width). Synchronized audio is recorded at a 20 kHz sampling rate. To create a training dataset, we utilized the the USC software package developed in [2], to extract the vocal tract lower and upper contours and area function. In this method a set of grid lines are placed on the vocal tract and the VT lower and upper contours are estimated on each grid line. The VTAF is then obtained from the estimated VT contours by computing the Euclidean distance between the upper contour points to the closest lower contour point (see [2] for more details). Figure 1 shows a sample extracted VT contour and the corresponding area function using the USC software.

2.2. Feature Representation

A major design decision in training the neural networks is representation of the features. In this work we investigate two different input feature types, acoustic speech and phone sequence, to train DNNs with two different kind of output values, vocal tract boundaries and area function.

2.2.1. DNN input: phone features

The phone features are simply chosen as the phoneme label of the input utterance. The phoneme transcription files available in the USC-TIMIT dataset are utilized to prepare the phone features in this study. An alternative for transcribing the speech is using any off the shelf forced aligner tool. For each video in the training set, we find a one-to-one mapping between the phoneme labels and the video frames i.e each video frame is labeled with a phoneme class according to where the mid-time of the frame lies in the phone transcript timeline. A standard set of 41 phonemes, each encoded as a one hot vector, is used for transcribing the data, including silence and short pause. We represent the extracted phone features as $\left\{f_j^p\right\}_{j=1}^N$, where $f_j^p \in \mathbb{R}^{41\times 1}$ is a column vector, j is the frame index and N is the number of frames in the training set.

The final phone features are obtained by utilizing a sliding window technique on the extracted phone feature set, which captures the temporal information of the phoneme sequence. We call these features *temporal* phone feature sequence and calculate them over a window of size $K_p = 2k_p + 1$ as:

$$F_j^p = [{f_{j-k_p}^p}^\top, ..., {f_j^p}^\top, ..., {f_{j+k_p}^p}^\top]^\top, \tag{1}$$

where the column vectors f_j^p are stacked column-wise resulting in a $41K_p \times 1$ column vector F_j^p. The temporal phone feature sequence, $\left\{F_j^p\right\}_{j=1}^N$, is used to train the phone-based DNN.

2.2.2. DNN input: acoustic features

Mel-Frequency Spectral Coefficients, also denoted as MFSC are utilized as the acoustic features. MFSC extraction is same as the better known MFCC features, without the discrete cosine transform, which is applied at the last stage of MFCC

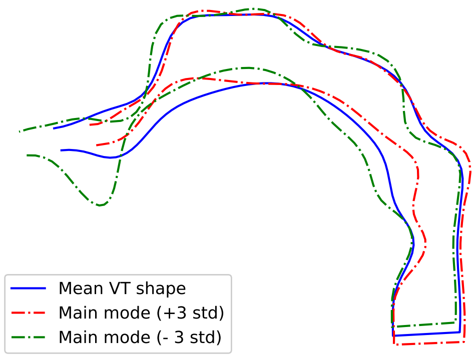


Fig. 2: Variation of the main mode corresponding to largest eigenvalue, around the mean shape. The model parameters are restricted to ±3 of their standard deviation to allow plausible VT shapes.

feature extraction. For each speech frame, 40 MFSC features along with their deltas and delta-deltas, are extracted using HTK toolkit [15] to define the acoustic energy distribution over 40 mel-frequency bands. These features are computed on short term overlapping Hamming window over the speech, with sampling interval chosen according to the frame rate of the corresponding videos. Each video frame length is approximately 44 ms, therefore we choose a Hamming window of size 25 ms with 11 ms frame shift, resulting in a 4-to-1 audio to video frame correspondence. We normalize the acoustic feature set to have zero mean and unit variance in each feature dimension. MFSC, delta and delta-delta features are stacked through the depth dimension like channels of a RGB image. The set of acoustic feature vectors is represented as $\{f_j^a\}_{j=1}^N$, where $f_j^a \in \mathbb{R}^{40\times4\times3}$ and 4 indicates the audio to video frame ratio.

The temporal acoustic feature sequence is calculated over a window of size $K_a = 2k_a + 1$ as:

$$F_j^a = [f_{j-k_a}^a, ..., f_j^a, ..., f_{j+k_a}^a], \tag{2}$$

where the features f_j^a are concatenated row-wise to form the $40 \times 4K_a \times 3$ temporal acoustic instance F_j^a. Each temporal acoustic feature could be interpreted as an image with height 40, width $4K_a$ and depth 3. We use the temporal acoustic feature sequence, $\{F_j^a\}_{j=1}^N$, to train the speech-based DNN.

The target articulatory values of the trained DNNs depend on the direct or indirect estimation scheme, which we explain in the following sections.

2.3. Indirect Estimation Scheme

In the indirect scheme, the VTAF is computed from the estimated vocal tract lower and upper contours. The vocal tract contour is represented by a set of M landmark coordinates $[x_1, y_1, x_2, y_2, ..., x_M, y_M]^\top$. In our experiments $M = 150$

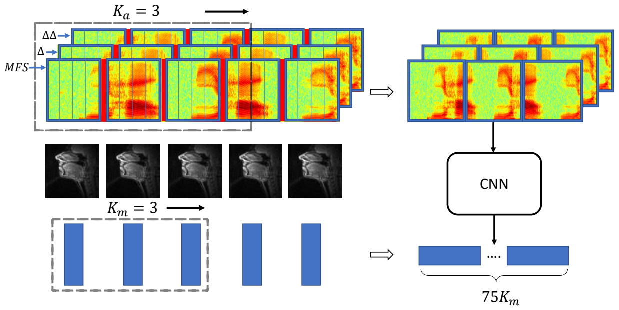


Fig. 3: Sliding window technique for temporal acoustic input and VTAF output feature representation. At each feature frame a window centered to the frame gives the temporal features. Note the 4-to-1 acoustic to video frame ratio.

(twice the number of the grid lines) and 75 points are chosen on the lower and 75 points on upper VT boundary. To estimate the VT boundaries, a statistical shape model is trained for the tract shapes using the Active Shape Model (ASM) [16]. ASM, utilizing Principal Component Analysis (PCA), models a set of shapes with their mean and eigenvectors of their covariance matrix.

The key element in the ASM is discarding eigenvectors corresponding to small eigenvalues hence removing the correlation and reducing the dimensionality in the data. Given the eigenvectors $(P_1, P_2, ..., P_{2M})$, sorted according to the descending eigenvalues $(\lambda_1, \lambda_2, ..., \lambda_{2M})$, the first k eigenvectors are chosen such that $(\sum_{j=1}^k \lambda_j / \sum_{j=1}^{2M} \lambda_j) > 0.99$. In our experiments by choosing $k = 37$, the shape model captures 99% of the variation in the training set. Each shape in the training set is transformed into a uncorrelated lower dimensional space as:

$$f = P^T(S - \bar{S}), \tag{3}$$

where $\bar{S}$ is the mean shape, P is the $2M \times k$ truncated eigenvectors matrix and f is the k-dimensional model parameters. Each of the 37 parameters is called a mode of variation in the shape model. Given these parameters, the VT shape is easily reconstructed as $S = \bar{S} + Pf$. Figure 2 illustrates the variation around the mean shape by setting the main mode (first parameter) to ±3 of it's standard deviation in the training set.

In the indirect estimation scheme, the PCA projected parameters are used as the target values at the DNN's output regression layer. The set of PCA values is represented as $\{f_j^i\}_{j=1}^N$, where $f_j^i \in \mathbb{R}^{37\times1}$ and i states the indirect estimation scheme.

2.4. Direct Estimation Scheme

In the direct estimation scheme, a DNN is trained to estimate the 75 area function values on each grid line from lips to the

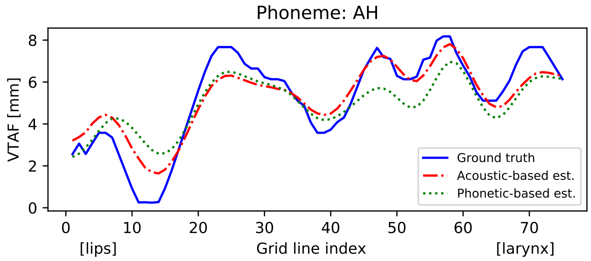


Fig. 4: Direct VTAF estimation scheme for vowel /AH/ using acoustic and phonetic features.

larynx. The area function values are extracted using the USC software package [2]. We represent the set of collected VTAF values as $\{f_j^d\}_{j=1}^N$, where $f_j^d \in \mathbb{R}^{75\times 1}$ and d states the direct estimation scheme.

Upon extracting the direct or indirect articulatory features, similar to the DNN's input sequences, a fixed-length overlapping sliding window is employed to the output articulatory sequence to capture the temporal information. The temporal target sequence is calculated over a window of size $K_m = 2k_m + 1$ as:

$$F_j^m = [f_{j-k_m}^m{}^\top, ..., f_j^m{}^\top, ..., f_{j+k_m}^m{}^\top]^\top \tag{4}$$

where $m \in \{d, i\}$, indicates the direct or indirect scheme and the column vectors f_j^m are stacked column-wise to form the temporal target values F_j^m of size $75K_d \times 1$ and $37K_i \times 1$ for direct and indirect schemes, respectively. We use the temporal articulatory feature sequence, $\{F_j^m\}_{j=1}^N$, as the target values at the DNN's output regression layer.

Figure 3 helps to understand the procedure of sliding window temporal feature extraction for direct acoustic-based scheme.

2.5. Network Architecture

The vocal tract area function estimation is modeled as a *multivariate regression* problem. The non-linear mapping from the temporal acoustic or phonetic feature sequence to the target VTAF or PCA sequence is learned by a regression function. The DNN regressors trained in this study are similar to what we have proposed in a previous work [12], which is explained briefly in the following.

2.5.1. *Phone-based architecture*

For phone-based experiments, we utilize a deep feed-forward neural network. The input layer, fed with the temporal phonetic features, is connected to three fully connected hidden layers with 1024 neurons each and a final output layer. Each hidden layer is followed by a hyperbolic tangent activation function, to induce the non-linearity. We employ standard mini batch stochastic gradient descent algorithm for training. Mini batch size is selected as 128 along with Adam optimizer [17] for learning rate adaptation. To avoid overfitting, dropout [18] with 50% probability is used at each hidden layer. The final output layer is standard multivariate regression layer predicting the target temporal articulatory sequence and trained to minimize the MAE loss.

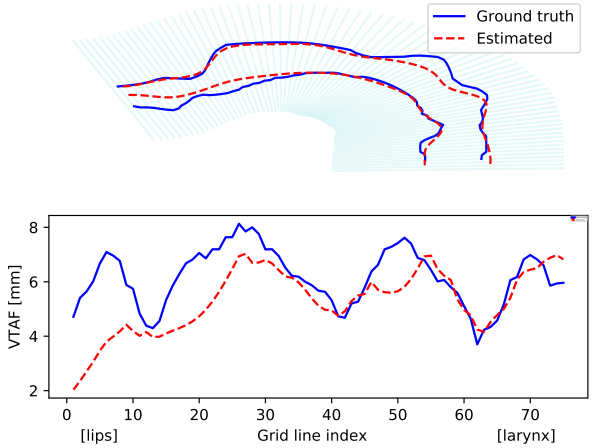


Fig. 5: Acoustic-based indirect VTAF estimation scheme. Area function is computed from the estimated VT boundaries.

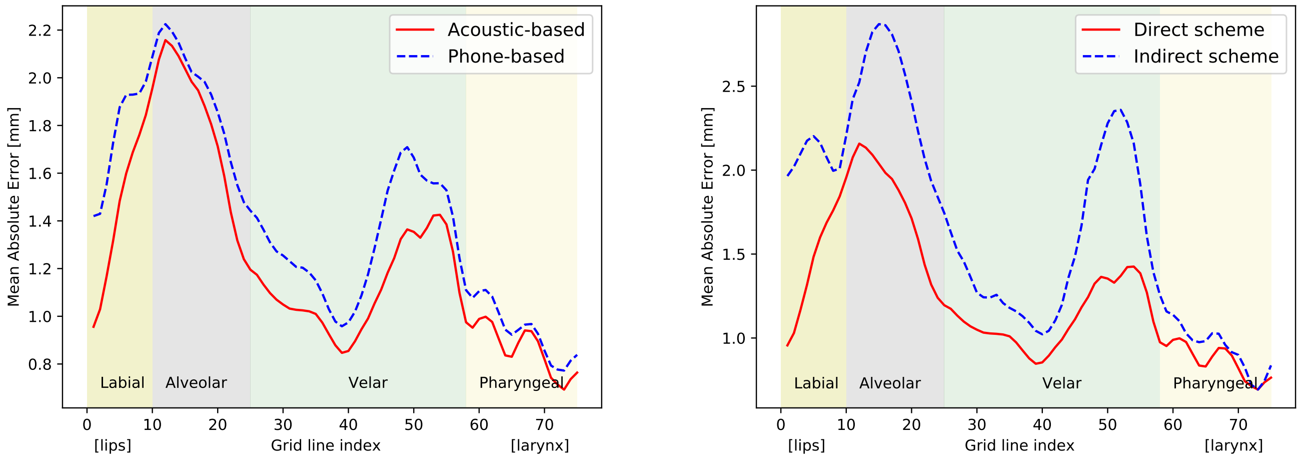
2.5.2. *Speech-based architecture*

Convolutional Neural Networks (CNN), firstly proposed for image classification tasks, are widely used in speech recognition related problems [19]. Having organized the acoustic features as image-like inputs, we employ a CNN architecture for speech-based model training.

The image-like temporal acoustic features are fed to the input layer of the speech-based network. The CNN network contains three convolution layers with first layer having 64 filters of size $7 \times 4K_a$. In the second convolution layer, 128 filters of size 5×1 is selected. The last convolutional layer is set to have 256 filters of size 3×1. All convolutional layers are followed by a pooling layer with window size of 2×1 and stride of 2×1. The network is then connected to three fully connected layers with 1024 hidden neurons each. We use dropout regularization method with probability 50%, in the convolutional and fully connected layers , to prevent overfitting. Hyperbolic tangent activation function is used at each layer with Adam optimizer for hyper learning rate optimizations.

We used Keras[1] with TensorFlow [20] backend to train the networks on a NVIDIA TITAN XP GPU.

[1] https://keras.io/



(a) Direc: acoustic vs phonetic

(b) Direct vs Indirect (acoustic)

Fig. 6: Error plots for direct and indirect schemes using acoustic or phonetic-based DNN, evaluated on different sub-regions of the vocal tract.

3. EXPERIMENTAL EVALUATIONS

In this work we use a single speaker for training and validating our introduced methods i.e. the experiments are performed in a speaker-dependent manner. To evaluate the performance of the proposed methods we calculate the mean absolute error (MAE) between the ground truth and the predicted area functions from different estimation schemes. The MAE is calculated in different sub-regions of the vocal tract on each grid line. The VT sub-regions are (1) grid lines 1∼ 10 for labial, (2) grid lines 11∼25 for alveolar and hard palate region, (3) grid lines 26∼58 for velar and dorsal constriction region and (4) grid lines 59∼75 for pharyngeal wall region.

A total of $44K$ samples comprising acoustic and phonetic features with the corresponding articulatory information (VT boundaries, shape model and area function), are collected during the data preparation. The networks are trained on $40K$ and validated on $4K$ samples. The network characteristics and sliding window sizes are optimized using fine tuning methods. We find out that selecting window sizes as $(K_a, K_p, K_d, K_i) = (7, 15, 5, 5)$ give the best results on the train and validation sets.

The estimated values at the output of the trained networks give the articulatory features in a temporal window of size K_m, hence the input feature sequence yields an overlapping output sequence. The articulatory parameters at a frame are calculated as the temporal mean of the overlapping output sequence.

Figure 4 shows the results of the direct VTAF estimation for the vowel /AH/. As observed from the figure, the acoustic-based estimation fits closer to the ground truth area function than the phone-based model; however it is observable that the phone-based estimation is capable of accurately capturing the outline of the VTAF configuration. As addressed in [13], the phone-based estimation results look slightly under articulated compared to the original data and need to be scaled up. We notice that under articulation is slightly less when using the acoustic features as input to the DNN.

In the indirect estimation scheme, the VT temporal shape model parameters are estimated using the trained DNN. The overlapping temporal shape model parameters are blended together using temporal mean to give the frame-wise parameters. The VT contours are then computed using inverse of (3) from the estimated PCA parameters. Upon retrieving the tract boundaries, VTAF is calculated as the distance between each point on the upper boundary to the closest point on the lower boundary, as shown in Figure 5.

Table 1: Performance evaluation of various proposed schemes. Values indicate the average mean absolute error per grid line in mm unit.

Scheme	**Acoustic-based**	**Phone-based**
Direct	1.23	1.41
Indirect	1.65	1.65

Figure 6 illustrates the error analysis plots for direct and indirect schemes with acoustic and phone-based networks. We observe that in a direct estimation scheme, the speech-based system is slightly better than the phone-based system, in all sub-regions of the tract, Figure 6a. From Figure 6b, the direct estimation scheme is observed to result in less error compared to the indirect scheme. The advantage of the direct scheme over the indirect scheme, is particularly more remarkable for a group of rapidly changing articulators including lips, tongue tip and velum, which are more deformable during the articulation. Analyzing the estimation error in different sub-regions of the VT gives insight to discover articulators with higher error results, hence focusing on these articulators to improve the proposed schemes in future studies. These error plots show that the proposed schemes estimate the VTAF for tongue tip and dorsal regions with higher error compared to other regions.

Table 1 presents the average mean absolute error for each grid line, between the estimated and ground truth VTAF, for different proposed schemes. These results suggest that all of the trained models perform the VTAF estimation problem with a small error, as compared to the pixel width (2.9 mm). We observe that an acoustic-based direct estimation scheme performs with less error compared to other schemes. It is observed that in the indirect estimation scheme, none of the acoustic or phone-based networks has a significant advantage over the other.

4. CONCLUSION AND FUTURE WORK

In this paper we proposed different schemes to efficiently estimate the vocal tract area functions using acoustic speech and phone sequence data. A direct and an indirect articulatory estimation model were trained using deep neural networks with acoustic or phonetic input sequences. We observed that all of the introduced estimation schemes are robust across different sub-regions of the VT. The acoustic-based direct estimation scheme was observed to be less erroneous than the indirect scheme, in all the defined sub-regions of the vocal tract.

The accuracy of the proposed methods depends on the correctness of the collected ground truth articulatory data. The articulatory data used in this study for the model training is not the real manually labeled data by an expert, and is solely the estimations of an accurate automatic algorithm. As a future work we will focus on a more error-free image-based articulatory estimator, to create larger and more accurate training datasets for speech-based articulatory estimation. We will work on a speaker independent VTAF estimation system by using acoustic and articulatory information of different speakers for the model training. Preparing a large multi-speaker dataset and approaching the VTAF estimation problem in a multimodal acoustic-phonetic manner, is a prospective approach to a speaker independent articulatory estimation.

5. ACKNOWLEDGMENT

We thank to NVIDIA for donating Titan XP within the GPU Grant Program.

6. REFERENCES

[1] S.Asadiabadi and E.Erzin, "Vocal tract airway tissue boundary tracking for rtmri using shape and appearance priors," in *Proc. Interspeech 2017*, 2017, pp. 636–640.

[2] J.Kim, N.Kumar, S.Lee, and Sh.Narayanan, "Enhanced airway-tissue boundary segmentation for real-time magnetic resonance imaging data," in *Tenth international seminar on Speech Production, ISSP10*, 2014.

[3] M.Proctor, D.Bone, N.Katsamanis, and Sh.Narayanan, "Rapid semi-automatic segmentation of real-time magnetic resonance image for parametric vocal tract analysis," in *Interspeech*, 2010, pp. 1576–1579.

[4] E.Bresch, J.Adams, A.Pouzet, S.Lee, D.Byrd, and Sh.Narayanan, "Semi-automatic processing of real-time mr image sequences for speech production studies," in *international seminar on Speech Production, ISSP*, 2006.

[5] H. Wakita, "Direct estimation of the vocal tract shape by inverse filtering of acoustic speech waveforms," *IEEE Transactions on Audio and Electroacoustics*, vol. 21, no. 5, pp. 417 – 427, 1973.

[6] M.A.Ismail, "Vocal tract area function estimation using particle swarm," *Journal of Computers (JCP)*, vol. 3, pp. 32–38, 2008.

[7] A.Katsamanis, G.Papandreou, and P.Maragos, "Face active appearance modeling and speech acoustic information to recover articulation," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, pp. 411–422, 2009.

[8] P.L.Tobing, H.Kameoka, and T.Toda, "Deep acoustic-to-articulatory inversion mapping with latent trajectory modeling," *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 1274–1277, 2017.

[9] Zh.Liu, Zh.Ling, and L.Dai, "Articulatory-to-acoustic conversion using blstm-rnns with augmented input representation," *Speech Communication*, vol. 99, pp. 161–172, 2018.

[10] T. Toda, A.W. Black, and K. Tokuda, "Statistical mapping between articulatory movements and acoustic spectrum using a gaussian mixture model," *Speech Commun.*, vol. 50, no. 3, pp. 215–227, Mar. 2008.

[11] 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, pp. 175–185, 2004.

[12] S.Asadiabadi, R.Sadiq, and E.Erzin, "Multimodal speech driven facial shape animation using deep neural networks," *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2018.

[13] S.L.Taylor, T.Kim, Y.Yue, M.Mahler, J.Krahe, A.Garcia Rodriguez, J.K.Hodgins, and I.A.Matthews, "A deep learning approach for generalized speech animation," *ACM Trans. Graph.*, vol. 36, pp. 93:1–93:11, 2017.

[14] Sh. Narayanan, A. Toutios, V. Ramanarayanan, A. Lammert, J. Kim, S. Lee, K. Nayak, Y. Kim, Y. Zhu, L. Goldstein, D. Byrd, E. Bresch, P. Ghosh, A. Katsamanis, and M. Proctor, "Real-time magnetic resonance imaging and electromagnetic articulography database for speech production research (tc).," *The Journal of the Acoustical Society of America*, vol. 136, no. 3, pp. 1307, 2014.

[15] S.Young, G.Evermann, D.Kershaw, G.Moore, J.Odell, D.Ollason, and V.Valtchev, *The HTK Book, version 3.4*, Cambridge University Engineering Department, Cambridge, UK, 2006.

[16] T. Cootes and C. Taylor, "Active shape model search using local grey-level methods: a quantitative approach," *Proc. British Machine Vision Conference*, pp. 639–648, 1993.

[17] D.P. Kingma and J.L. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[19] O.Abdel-Hamid, A.Mohamed, H.Jiang, L.Deng, G.Penn, and D.Yu, "Convolutional neural networks for speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 1533–1545, 2014.

[20] M.Abadi, P.Barham, J.Chen, Zh.Chen, A.Davis, J.Dean, M.Devin, S.Ghemawat, G.Irving, M.Isard, M.Kudlur, J.Levenberg, R.Monga, Sh.Moore, D.Gordon Murray, B.Steiner, P.A.Tucker, V.Vasudevan, P.Warden, M.Wicke, Y.Yu, and X.Zheng, "Tensorflow: A system for large-scale machine learning," in *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*, Berkeley, CA, USA, 2016, OSDI'16, pp. 265–283, USENIX Association.