A Trajectory Mixture Density Network for the Acoustic-Articulatory Inversion Mapping

Korin Richmond

Centre for Speech Technology Research
University of Edinburgh, Edinburgh, United Kingdom
korin@cstr.ed.ac.uk

Abstract

This paper proposes a trajectory model which is based on a mixture density network trained with target features augmented with dynamic features together with an algorithm for estimating maximum likelihood trajectories which respects constraints between the static and derived dynamic features. This model was evaluated on an inversion mapping task. We found the introduction of the trajectory model successfully reduced root mean square error by up to 7.5%, as well as increasing correlation scores.
Index Terms: acoustic-articulatory inversion, conditional trajectory model, mixture density network.

1. Introduction

The acoustic-articulatory inversion mapping involves inverting the forward process of speech production. In other words, for a given acoustic speech signal we aim to estimate the underlying sequence of articulatory configurations which produced it. Doing this well could prove useful for many applications; for example low bit-rate speech coding [1], speech analysis and synthesis [2], automatic speech recognition [3], animating talking heads and so on.

Researchers have been investigating the inversion mapping for several decades. Much work has focused on analysis of acoustic signals based on mathematical models of speech production [4]. Articulatory synthesis models have also been used extensively, either as part of a mimic, analysis-by-synthesis algorithm [5], or to generate acoustic-articulatory databases which may be used as part of a code-book approach [6] or to train other models [7]. More recently, the availability of larger quantities of human articulography data, for example from electromagnetic articulography (EMA), has prompted much work on applying machine learning models to human articulatory data, including artificial neural networks (ANNs) [8], codebook methods [9] and GMMs [10].

It is widely regarded that the difficulty in the acoustic-articulatory mapping lies in its ill-posed nature. There is significant evidence to indicate that multiple articulatory configurations can result in the same or very similar acoustic effect. In light of this instantaneous "non-uniqueness", how is a system intended to perform the inversion mapping to choose been the alternatives?

In previous work [8], we have successfully used the mixture density network (MDN) [11] to address this problem, as it gives a full probability density function (pdf) over the target articulatory domain conditioned on the acoustic input. Other researchers have used dynamic constraints to disambiguate instantaneous non-uniqueness, for example [9, 7, 10]. Of these, the last is particularly interesting. [10] used a GMM to perform the inversion mapping, but formulated it as a statistical trajectory model by augmenting observations with delta and deltadelta features and then using the maximum likelihood parameter estimation (MLPG) algorithm described by [12] to give the maximum likelihood estimation of articulatory trajectories which respects constraints between the static and derived dynamic features. This same technique has also been applied within an HMM-based speech production model for the inversion mapping [13].

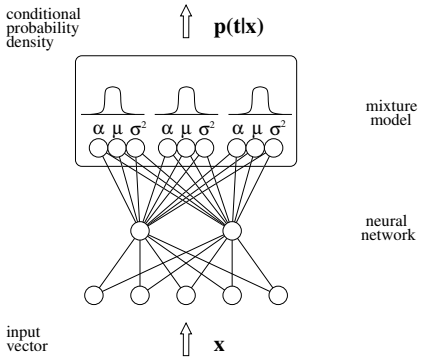

Figure 1: *The mixture density network is the combination of a mixture model and a neural network.*

Due to the similarity in form of these models, it is natural to ask whether MLPG can be usefully applied in the case of the MDN too. The purpose of this paper is to evaluate this augmentation of the MDN with a trajectory model on an inversion mapping task.

2. An MDN-based trajectory model

Since it is not widely known in the speech community, we give here a very brief introduction to the MDN, before describing how it may be extended with the MLPG algorithm to give a trajectory model. For full details, the reader is referred to [11] and [12].

2.1. Mixture density networks

The MDN can be viewed as the amalgamation of a mixture model and an ANN. In theory, any ANN with universal approximation capabilities can be used and the mixture model can contain any of a number of different kernel functions. Here, we will consider only a multilayer perceptron and Gaussian mixture components (priors α, means μ and variances σ^2). In the trained MDN, the ANN part is responsible for mapping from the input vector $\mathbf{x}$ to the control parameters of the mixture model, which in turn gives a full pdf over the target domain, conditioned on the input vector

$p(\mathbf{t}|\mathbf{x})$. The toy-example MDN in Figure 1 takes an input vector $\mathbf{x}$ of dimensionality 5 and gives the conditional probability density of a vector $\mathbf{t}$ of dimensionality 1 in the target domain. This pdf takes the form of a GMM with 3 components, so it is given as:

$$p(\mathbf{t}|\mathbf{x}) = \sum_{j=1}^{M} \alpha_j(\mathbf{x})\phi_j(\mathbf{t}|\mathbf{x}) \quad (1)$$

where M is the number of mixture components (in this example, 3), $\phi_j(\mathbf{t}|\mathbf{x})$ is the conditional probability density given by the jth kernel, and $\alpha_j(\mathbf{x})$ is the mixing coefficient for the jth kernel.

In order to constrain the mixing coefficients to lie within the range $0 \leq \alpha_j(\mathbf{x}) \leq 1$ and to sum to unity, the *softmax* function is used to relate the output of the corresponding units in the neural network to the mixing coefficients

$$\alpha_j = \frac{\exp(z_j^\alpha)}{\sum_{l=1}^{M} \exp(z_l^\alpha)} \quad (2)$$

where z_j^α is the output of the neural network corresponding to the mixture coefficient for the jth mixture component. The variance parameters are similarly related to the outputs of the ANN as

$$\sigma_j = \exp(z_j^\sigma) \quad (3)$$

where z_j^σ is the output of the neural network corresponding to the variance for the jth mixture component, which avoids the variance becoming less than or equal to zero. Finally, the means are represented directly by the corresponding outputs of the ANN:

$$\mu_{jk} = z_{jk}^\mu \quad (4)$$

where z_{jk}^μ is the value of the output unit corresponding to the kth dimension of the mean vector for the jth mixture component.

The objective of training the MDN is to minimise the negative log likelihood of the observed target data points

$$E = -\sum_n \ln \left\{ \sum_{j=1}^{M} \alpha_j(\mathbf{x}^n)\phi_j(\mathbf{t}^n|\mathbf{x}^n) \right\} \quad (5)$$

given the mixture model parameters. Since the ANN part of the MDN provides the parameters for the mixture model, this error function must be minimised with respect to the network weights. Therefore, the derivatives of the error at the network output units corresponding separately to the priors, means and variances of the mixture model are calculated (see [11]) and then propagated back through the network to find the derivatives of the error with respect to the network weights. Thus, training the MDN is a problem to which standard non-linear optimisation algorithms can be applied.

2.2. Maximum likelihood parameter generation

The first step to an MDN-based trajectory model is to train an MDN with target feature vectors augmented with dynamic features, standardly derived from linear combinations of a window of static features. For the sake of simplicity, we will consider MDNs with a single Gaussian distribution and a single target static feature c_t at each time step. Next, given the output of this MDN in response to a sequence of input vectors, in order to generate the maximum likelihood trajectory, we aim to maximize $P(\mathbf{O}|\mathbf{Q})$ with respect to $\mathbf{O}$, where $\mathbf{O} = [\mathbf{o}_1^T, \mathbf{o}_2^T, ..., \mathbf{o}_T^T]^T$, $\mathbf{o}_t = [c_t, \Delta c_t, \Delta\Delta c_t]$ and $\mathbf{Q}$ is the sequence of Gaussians output by our MDN. The relationship between the static features and those augmented with derived dynamic features can be arranged in matrix form,

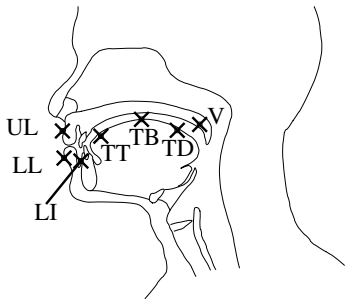

Figure 2: *Placement of EMA receiver coils in the MOCHA database. See Table 1 for the key to abbreviations In this paper, we used the 6 EMA channels for the tongue (x- and y-coords for three positions).*

label	articulator	label	articulator
UL	Upper lip	TT	Tongue tip
LL	Lower lip	TB	Tongue body
LI	Lower incisor	TD	Tongue Dorsum

Table 1: *Key for placement of coils in the MOCHA dataset.*

$$\mathbf{O} = \mathbf{WC} \quad (6)$$

where $\mathbf{C}$ is a sequence of static features and $\mathbf{W}$ is a transformation matrix composed of the coefficients of the delta and deltadelta calculation window and $\mathbf{0}$. Under the condition expressed in Eq. 6, maximising $P(\mathbf{O}|\mathbf{Q})$ is equivalent to maximising $P(\mathbf{WC}|\mathbf{Q})$ with respect to $\mathbf{C}$. By setting

$$\frac{\partial \log P(\mathbf{WC}|\mathbf{Q})}{\partial \mathbf{C}} = 0 \quad (7)$$

a set of linear equations is obtained (see [12])

$$\mathbf{W}^T\mathbf{U}^{-1}\mathbf{WC} = \mathbf{W}^T\mathbf{U}^{-1}\mathbf{M}^T \quad (8)$$

where $\mathbf{M}^T = [\mu_{q_1}, \mu_{q_2}, ..., \mu_{q_T}]$ and $\mathbf{U}^{-1} = diag[\mathbf{U}_{q_1}^{-1}, \mathbf{U}_{q_2}^{-1}, ..., \mathbf{U}_{q_T}^{-1}]$ (μ_{q_T} and $\mathbf{U}_{q_t}^{-1}$ are the 3×1 mean vector and 3×3 (diagonal) covariance matrix respectively). Solving Eq. 8 for $\mathbf{C}$ yields the maximum likelihood trajectory.

3. Inversion experiment

To test whether the MLPG technique can be successfully combined in practice with the MDN to form a trajectory model, we carried out an inversion mapping experiment. We shall first describe the data used and then the experiment itself.

3.1. MOCHA articulatory data

The multichannel articulatory (MOCHA) data set [14] contains four data streams recorded concurrently: the acoustic waveform together with laryngograph, electropalatograph and electromagnetic articulograph (2D EMA) data. Each of the sensors shown in Figure 2 provide x- and y-coordinates in the midsagittal plane sampled at 500Hz. Multiple speakers were recorded reading a set of 460 short, phonetically-balanced British-TIMIT sentences.

The EMA data and speech waveforms for female British English speaker `fsew0` were chosen from MOCHA for the experiments in this paper. This is exactly the same data set as used in

[8], and so enables comparison with those and other similar results reported in the literature (e.g. [10]).

3.1.1. Data processing

The acoustic data in the MOCHA dataset was subjected to filterbank analysis, using a Hamming window of 20ms with a shift of 10ms, resulting in an acoustic vector of 20 melscale filterbank coefficients for each time frame. These were z-score normalised and scaled to lie within the range [0.0,1.0]. Meanwhile, the corresponding EMA trajectories were downsampled to match the 10ms shift rate of the acoustic features, then z-score normalised and scaled to lie within the range [0.1,0.9]. The EMA processing steps also incorporated the normalisation technique described in [15], which aims to reduce the effect of EMA measurement error. Care was taken to discard feature vectors corresponding to the silence at the beginning and end of each file, using the HMM force-alignment labelling provided with the MOCHA dataset.

The partitioning of the data set into training, validation and test sets is also the same as that in [8]. Of the 460 utterances contained in the dataset for speaker `fsew0`, 368 were included in the training set, and the validation and testing sets contained 46 files each. A context window of input acoustic frames was used of length 20 consecutive frames, which increased the order of the input acoustic vector paired with each articulatory vector to 400.

3.2. Method

A straightforward way to implement and evaluate a Trajectory MDN is to train separate MDNs for each of the static and derived dynamic features, the output of which may then be used to perform the MLPG algorithm to yield the maximum likelihood trajectory. Thus, we chose to train 3 MDNs for each of the 6 channels of EMA data for the tongue: one for the standard EMA features, one for the delta features and one for the deltadelta features, making a total of 18 MDNs trained. All networks contained a single hidden layer of 60 units, which had been identified as a suitable number in previous experiments. For these initial experiments, we decided to use a single Gaussian for each of the MDNs as a simplification and to provide a baseline for future experiments.

Training of the networks was canonical; the scaled conjugate gradients non-linear optimisation algorithm was run for a maximum of 2000 epochs, and the separate validation set of 46 utterances was used to identify the point at which an optimum appeared to have been reached.

Generating output trajectories simply involves running the input data for an utterance through the three MDNs for the static, delta and deltadelta features for each articulatory channel, and then running the MLPG algorithm on the resulting sequences of pdfs.

In order to demonstrate the effect of using the dynamic features and the MLPG algorithm together to form the Trajectory MDN, we can compare the resulting trajectories with those comprising just the mean of the MDNs trained on the static data. Taking the output corresponding to the mean of the MDN output pdf is equivalent to using an MLP (with linear output activation function) trained with a standard least-squares error function. In this way, therefore, we can directly observe the effect of using the augmented dynamic features without regard to any confounding effect of two systems having been trained differently.

	correlation		RMSE(mm)		% RMSE
channel	static	$+\Delta, \Delta\Delta$	static	$+\Delta, \Delta\Delta$	reduction
tt_x	0.82	0.84	2.30	2.22	3.6
tt_y	0.87	0.89	2.31	2.22	3.9
tb_x	0.82	0.84	2.13	2.04	4.1
tb_y	0.86	0.88	1.93	1.81	6.4
td_x	0.81	0.82	1.98	1.91	3.8
td_y	0.78	0.81	2.07	1.92	7.5

Table 2: *Comparison of results for the inversion mapping estimated by MLP-equivalent (only static features) and the full Trajectory MDN (using dynamic features and the MLPG algorithm).*

	correlation		RMSE(mm)		% RMSE
channel	prev	$+\Delta, \Delta\Delta$	prev	$+\Delta, \Delta\Delta$	reduction
tt_x	0.79	0.84	2.43	2.22	8.7
tt_y	0.84	0.89	2.56	2.22	13.4
tb_x	0.81	0.84	2.19	2.04	6.7
tb_y	0.83	0.88	2.14	1.81	15.4
td_x	0.79	0.82	2.04	1.91	6.6
td_y	0.71	0.81	2.31	1.92	17.0

Table 3: *Comparison of results observed in this experiment with those previously reported in [15]* ("prev").

4. Results

Figure 3 gives an example utterance to compare the two estimated trajectories with the true one. Meanwhile, Table 2 gives the results comparing the performance of the standard MLP equivalent (static features only) with the proposed Trajectory MDN ($+\Delta, \Delta\Delta$). Two measures have been used: correlation and root mean square error (RMSE) expressed in millimetres. The use of dynamic features and the MLPG algorithm within the Trajectory MDN has improved results in terms of a reduction of RMS error and an increase in correlation for all channels tested.

Table 3 compares the performance of the Trajectory MDN presented here with the corresponding results previously reported for the same dataset in [8]. The improvement using the Trajectory MDN proposed here over the MLP results in [8] is substantial, ranging between 6.6% and 17.0% in RMS error reduction.

It is also worth noting the results for the Trajectory MDN are in line with those reported in [10]. The average RMSE for the articulators reported here is 2.02mm, while the average RMSE of the best results reported in [10] for the same articulators is 1.98mm.

5. Discussion

The experiment described has aimed simply to establish that the technique works and to provide a baseline. Subsequent work will increase complexity, including inversion for the full set of articulators, and developing an MDN implementation with diagonal covariance, so the augmented feature vectors may be trained in a single MDN instead of three separate ones. In addition, we intend to evaluate using multiple mixtures in the MDN output pdf, which will require a decoding step for the sequence of mixture components, as described in [12]. Results from [8] indicate that using multiple mixtures does give a more accurate representation of the target articulatory domain than a single Gaussian, therefore we expect this will improve results further. Finally, so far, we have only

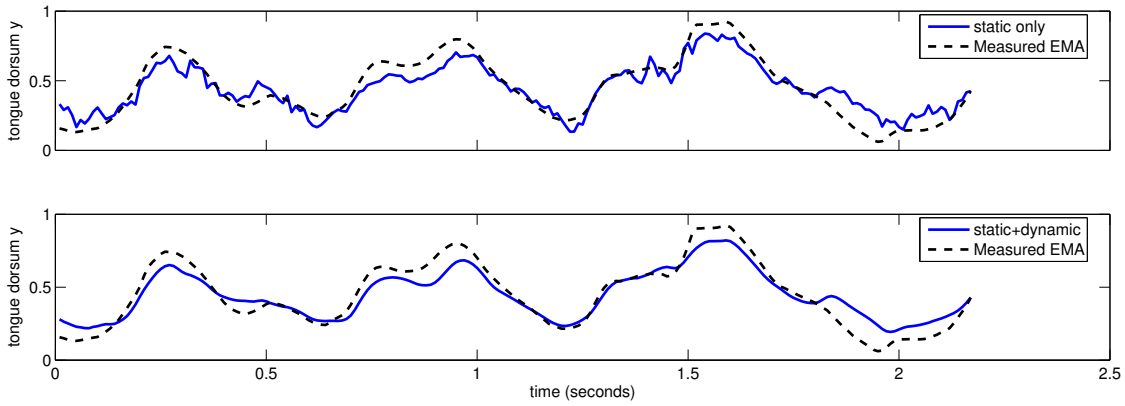

Figure 3: *Comparing the MLP-equivalent (only static features) and the full Trajectory MDN (dynamic features and the MLPG algorithm) for the utterance "The speech symposium might begin on Monday." The trajectory MDN output is smoother and closer in nature to the real trajectory, and more accurate (e.g. around* 0.5 *and* 1.5*s).*

applied the MLPG algorithm for trajectory estimation. In future work, we intend to look at whether respecting the same constraints between static and dynamic features can be applied to MDN training too.

6. Conclusions

We have demonstrated that the MDN may successfully be extended to provide a statistical conditional trajectory model by augmenting the static target features with derived dynamic features and using the maximum likelihood parameter generation algorithm. This method provides a useful way to use the output of the mixture density network where a single trajectory is required rather than a probability density function. Using this method, we have substantially improved upon the performance of our previous neural network inversion mapping. Finally, the success of the method in this case shows promise that the trajectory MDN may prove useful in modelling conditional trajectories in other problems.

7. References

[1] J. Schroeter and M. M. Sondhi, "Speech coding based on physiological models of speech production," in *Advances in Speech Signal Processing*, S. Furui and M. M. Sondhi, Eds., chapter 8, pp. 231–268. Marcel Dekker Inc, New York, 1992.

[2] T. Toda, A. Black, and K. Tokuda, "Mapping from articulatory movements to vocal tract spectrum with gaussian mixture model for articulatory speech synthesis," in *Proc. 5th ISCA Workshop on Speech Synthesis*, 2004.

[3] A. Wrench and K. Richmond, "Continuous speech recognition using articulatory data," in *Proc. ICSLP 2000*, Beijing, China, 2000.

[4] H. Wakita, "Estimation of vocal-tract shapes from acoustical analysis of the speech wave: The state of the art," *IEEE Trans. Acoust. Speech Signal Process.*, vol. ASSP-27, pp. 281–285, 1979.

[5] K. Shirai and T. Kobayashi, "Estimating articulatory motion from speech wave," *Speech Communication*, vol. 5, pp. 159–170, 1986.

[6] B. S. Atal, J. J. Chang, M. V. Mathews, and J. W. Tukey, "Inversion of articulatory-to-acoustic transformation in the vocal tract by a computer sorting technique," *J. Acoust. Soc. Am.*, vol. 63, pp. 1535–1555, 1978.

[7] M. G. Rahim, W. B. Kleijn, J. Schroeter, and C. C. Goodyear, "Acoustic-to-articulatory parameter mapping using an assembly of neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1991, pp. 485–488.

[8] K. Richmond, S. King, and P. Taylor, "Modelling the uncertainty in recovering articulation from acoustics," *Computer Speech and Language*, vol. 17, pp. 153–172, 2003.

[9] J. Hogden, A. Lofqvist, V. Gracco, I. Zlokarnik, P. Rubin, and E. Saltzman, "Accurate recovery of articulator positions from acoustics: New conclusions based on human data," *J. Acoust. Soc. Am.*, vol. 100, no. 3, pp. 1819–1834, September 1996.

[10] T. Toda, A. Black, and K. Tokuda, "Acoustic-to-articulatory inversion mapping with gaussian mixture model," in *Proc. 8th International Conference on Spoken Language Processing*, Jeju, Korea, 2004.

[11] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.

[12] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, and T. Kitamura, "Speech parameter generation algorithms for HMM-based speech synthesis," in *Proc. ICASSP*, Istanbul, Turkey, June 2000, pp. 1315–1318.

[13] Sadao Hiroya and Masaaki 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, mar 2004.

[14] A. Wrench, "The MOCHA-TIMIT articulatory database," http://www.cstr.ed.ac.uk/artic/mocha.html, 1999.

[15] K. Richmond, *Estimating Articulatory Parameters from the Acoustic Speech Signal*, Ph.D. thesis, The Centre for Speech Technology Research, Edinburgh University, 2002.