Training a Vocal Tract Synthesizer to Imitate Speech using Distal Supervised Learning

Ian S Howard[1] *& Mark A Huckvale*[2]

Sobell Department, Institute of Neurology[1]
Phonetics & Linguistics[2]
University College London, England
i.howard@ion.ucl.ac.uk, m.huckvale@ucl.ac.uk

Abstract

Imitation is a powerful mechanism by which both animals and people can learn useful behavior, by copying the actions of others. We adopt this approach as a means to control an articulatory speech synthesizer. The goal of our project is to build a system that can learn to mimic speech using its own vocal tract. We approach this task by training an inverse mapping between the synthesizer's control parameters and their auditory consequences. In this paper we compare the direct estimation of this inverse model with the distal supervised learning scheme proposed by Jordan & Rumelhart (1992). Both of these approaches involve a babbling phase, which is used to learn the auditory consequences of the articulatory controls. We show that both schemes perform well on speech generated by the synthesizer itself, when no normalization is needed, but that distal learning provided slightly better performance with speech generated by a real human subject.

1. Introduction

In order for a person to be able to imitate speech (or any other sound) using their vocal tract, it is necessary for them to relate an auditory representation of the sound to the motor control signals needed to move their vocal apparatus appropriately. The imitated acoustic output should aim to be the best possible match to the target speech, even though an exact match will generally not be possible due to differences between the generating and imitation systems. The imitation process thus requires not only information relating to how to control the vocal tract, but also the ability to judge the similarity between sounds, even when there are significant differences between them.

1.1. Inverse Models

At the heart of our approach is an inverse model that maps between an acoustic representation of speech and the motor control signals needed to imitate the input. In the first instance we consider the task of imitating a system that has an identical vocal tract. This avoids issues of speaker normalization, and these will be discussed later. In this case it is theoretically possible to make a very close match to the original speech, since both of the speech generators are identical.

1.2. Articulator Inversion

Relating the acoustic consequences of a vocal tract synthesizer back to its corresponding control signals can be achieved using an inverse model. The field of articulator inversion is by no means new, and many researches have contributed to this field [2]. However, many have been concerned with the use of real articulator measurements or representations in terms of formants frequencies, as well as other theoretical issues [3,4,5]. Our interest lies in building a system that will repeat a given acoustic utterance using a vocal tract synthesizer, and to do so without the use of any intermediate real articulator measurements.

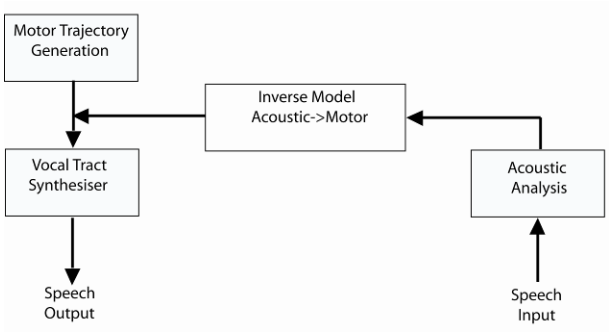

Figure 1: Schematic diagram of speech imitation system.

2. Directly Learning the Inverse Model

2.1. Basic System

Figure 1 shows a block diagram of our speech imitation system. This shows an articulator based speech synthesizer that is driven by motor control signals, resulting in speech output. The control input to the synthesizer can be produced either internally within the system using a motor trajectory generator or from the output from an inverse model, the input to which consists of the auditory analysis of a speech utterance to be imitated. Initially this inverse transformation is unknown and its estimation is the main task of the work described here.

2.2. Using Babble to Perform System Identification

If we run the synthesizer using a random babble generator and then feed its speech output back into the acoustic analysis, we have the basic structure we need to estimate the inverse model (see figure 2). In order to define the input/output mapping of the synthesizer, representative input output samples pairs are clearly required. This necessitates driving the synthesizer in a

fashion that will sample its input in a way that is statistically consistent with its intended use in generating real speech utterances. Since such control signals are not available ***a priori*** (if they were we, would have solved our control problem) we using a scheme based on a Hidden Markov Model Generator (HMM) to generate a first approximation of speech like movements of the articulators. We use this HMM to randomly sample phonetically significant regions of synthesiser space and then interpolate the output between these regions so as to generate a slowly varying time signal vector that corresponds well to the kind of movements the articulators would make when generating real speech. In a previous paper we discuss these issues in more detail [6].

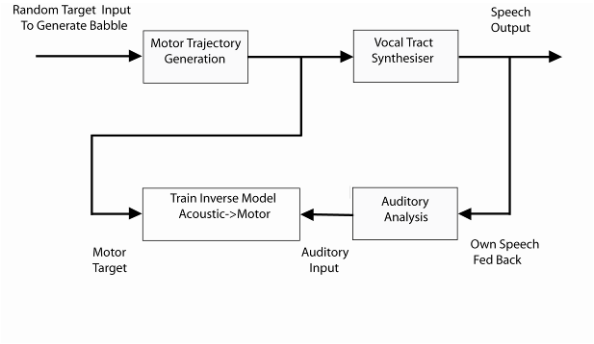

Figure 2: Direct training of the Inverse Model.

2.3. Direct Inverse Estimation

After generating data relating the synthesizer controls to their acoustic consequences by babbling, we can then use this dataset to directly train an inverse model using a classical supervised regression technique [7].

3. Learning the Inverse model using Distal Supervised Learning

3.1. The convexity Problem

As pointer out by in [1], direct estimation of the inverse model can run in to problems. The first of these arises because many vocal tract configurations can lead to similar or identical acoustic consequences. Because pattern regression techniques will tend to average the target over all of these configurations (in this case the vocal tract configurations) it is possible that the resulting estimate may not be a solution to the inverse mapping. This will occur if the target set is not convex, in which case the average over target space in not a member of the target set.

3.2. Goal Directed Optimization

Also as pointed out in [1], another limitation of direct estimation of the inverse model is that it is not goal directed. In our case this means that the output error is formulated in articulator space rather than in acoustic space. Since we are ultimately concerned with acoustic matching (we are interested in how the imitation sounds to us), the latter is clearly more desirable. Thus, the distal learning approach of Jordan and Rumelhart may provide a useful means to evaluate the matching error in a more psycho-acoustically relevant fashion. Indeed if we could represent the acoustic signal a way that optimally related to its psycho-acoustic relevance, even the same simple Euclidian similarity metric would be likely to give better agreement to a human measure of acoustic similarity than otherwise.

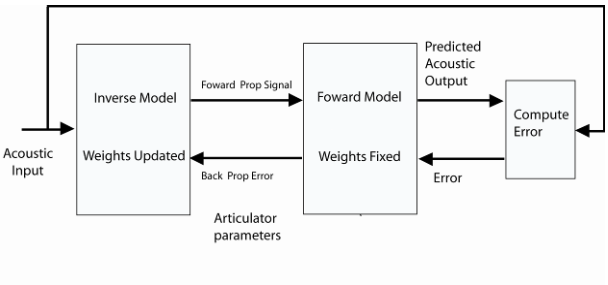

Figure 3: Distal Supervised Training.

Both the convexity problem and lack of goal directed learning can be overcome using the distal supervised learning approach. The basic idea behind this approach involves first training a forward model and an inverse model. The two are then combine into a single network and the parameters of the forward model are fixed. The joint network is then trained to map from the acoustic data, via an intermediate articulatory representation, and then back to the acoustic data. This is illustrated in figure 3. The forward network is used to convert the estimation of articulator space due to the inverse model into acoustic space. In this way an acoustic error for the inverse model can be defined and subsequently propagated backward through the forward model and use to modify the inverse model.

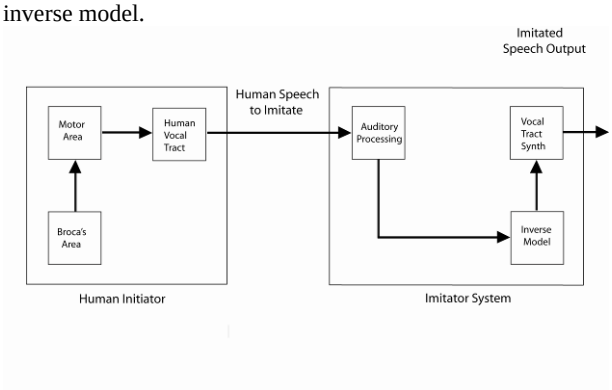

Figure 4: Using the Inverse Model for Imitation.

4. Methods

In out work here we used a 9-parameter articulator synthesiser based on the work of Maeda [8]. To limit computational requirements we adopted a speech sampling rate of 8kHz. The babble generator was run to sample five pure vowels and the consonant targets /b/ and /g/, as well as a state for silence, to generate 300 seconds of articulator trajectory data. The acoustic analysis is based on the JSRU channel vocoder [9] together with a simple autocorrelation estimate for fundamental frequency and voicing, and generates a 21 element data vector every 10ms.

We employed a multi-layer perceptron (MLP) to implement the forward and inverse models and a Matlab implementation was used [10]. They were trained using back-propagation [11] with conjugate gradient descent. The input to the inverse model consisted of 10 centred adjacent vocoder frames spanning 100ms in time, and the MLP had 40 hidden units and 9 linear outputs. The forward model used only a single input and output frame, 40 hidden units and linear outputs. The time delay between the control parameters and acoustic analysis was estimated by running single input frame forward and inverse models on the data and selecting the delay that minimized the error. Training the inverse, forward and combined models involved 1000 passes over the data set. During re-synthesis of the articulatory data using the inverse models, the output trajectories were smoothed with a 15 point median filter (spanning 150ms) to remove glitches from the output trajectories.

5. Results

5.1. Speech Resynthesis

After training, both the direct and distally trained inverse models were evaluated by re-synthesising input speech. This was achieved by passing an externally generated speech signal (that is, speech from another identical synthesizer and a human subject) through the acoustic analysis, then through the inverse model and finally to the synthesizer, as shown in figure 4. Evaluations were carried by listening tests and also by observation of the corresponding wideband spectrograms. A more detailed examination of the operation of our imitation system performing a re-synthesis its own speech is provided in [12]. This paper examines in detail the behaviour of the articulator control signals generated by an inverse model.

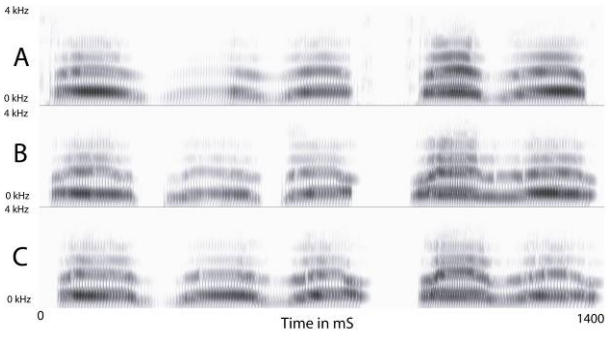

Figure 5: Wideband spectrograms for input utterance /baba/ babble from synthesiser (A), and re-synthesised outputs generated by the direct (B) and distal retrained (C) imitation system.

5.2. On its own Speech

On its own speech, performance was very good, with the re-synthesis speech being almost indistinguishable from the original. This similarity was also verified using spectrographic analysis, as shown in figure 5. It can be seen that the first 2 formants correspond well with the original and even the higher formants 3 and 4 match quite well.

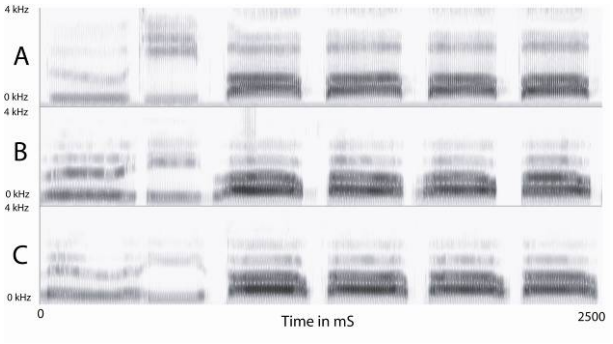

Figure 6: Wideband spectrograms for input utterance /boo gie ba ba ba ba/ from a male speaker (A), and re-synthesised outputs generated by the direct (B) and distal retrained (C) imitation system.

5.3. On Speech from a Male Subject

Re-synthesising speech from a real human subject was also investigated. This is a much more difficult problem because the real vocal tract that generated the real speech will generally have different characteristics from that used by the imitation system. This therefore raises the issues of speaker normalization and the generalization capabilities of the system. Performance in this case was noticeably worse that the original, although simple utterances could still be understood. Figure 6 and 7 shown spectrographic analysis of such utterances. The first format corresponds well in both imitated cases and the second not quite so well. Distal training did slightly improve the sound of the re-synthesis although this improvement is difficult to judge on the basis of the spectrograms. The reader is therefore urged to listen to the speech samples made by the synthesis (see supplementary information section).

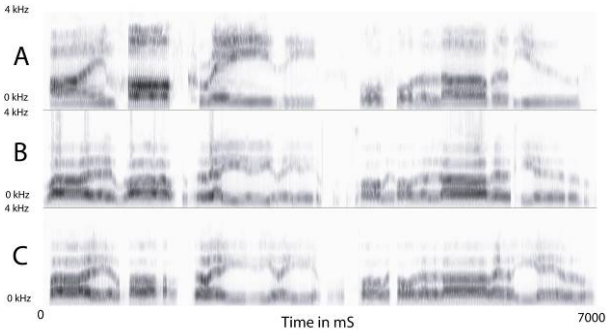

Figure 7: Wideband spectrograms for sung input utterance "I'm half crazy, all for the love of you", taken from the song Daisy, for a male speaker (A), and re-synthesised outputs generated by the direct (B) and distal retrained (C) imitation system.

6. Conclusions and Future Work

6.1. Conclusions

In this paper we trained an inverse model to perform imitation of target speech utterances using an articulatory synthesizer. We compared two approaches to the training, both a direct approach and a distal supervised estimation of the inverse model. The system achieved a good imitation of its own speech in both cases. However with real speech from a male speaker, performance was worse in both cases, although there was a slightly improvement using the distal supervised learning scheme. This suggests that the main limitations in the performance of the direct trained re-synthesis system were not due to the convexity problem or the lack of goal directed learning. Rather, it appears likely that the dominant limitation arose because of normalization issues between the human speaker and the system.

6.2. Modeling Articulator Dynamics

The current implementation of the inverse model makes no attempt to model the dynamics of the articulators. It simply performs a static transformation with no regard to their current state. This was partly the reason that median filtering was needed to smooth out glitches in the predicted articulator trajectories. Such a filtering operation is, however, by no means without its deficiencies and it introduces undesirable distortions into the trajectories. It is to be expected that the use of prior knowledge regarding the state of the articulators would improve the estimate of their position. Such issues are elegantly addressed by Kalman filtering techniques, which have been successfully applied to many tasks that requiring object tracking on the basis of noisy measurements (for example [13]).

6.3. Speaker Normalization

Presently we have not made any explicit attempt to account for differences between speech from the synthesizer and human speakers. In the first instance would be interesting to use a more psycho-acoustically motivated representation of the input speech to see if this improved the generalization capabilities of the inverse model. Another approach would be to try to learn a transformation that mapped input acoustic data onto a more canonical representation before reaching the inverse model.

6.4. Phonetic Representation of Speech

A step further would involve using an auditory analysis that explicitly represents the phonetic characteristics of the input speech. Such a scheme would have an advantage over a more general representation because it would be constrained to generate phonetically relevant movements of the articulators.

7. Supplementary Information

A supplement to this paper containing the .wav format audio samples described in the text can be found on the website http://www.ianhoward.de/specom_2005.htm.

8. Acknowledgements

We wish to thank Daniel Wolpert for supporting this work. The implementation of the articulator synthesizer was based on an implementation by Shinji Maeda within the DOS program VTCALCS.

9. References

[1] Jordan, M. I., and Rumelhart, D. E. 1992. ‘‘Forward models—Supervised learning with a distal teacher,’’ *Cogn. Sci. 16, 307–354.*

[2] Schroeter, J., Sondhi, M.M., "Techniques for estimating vocal-tract shapes from the speech signal", *IEEE Trans. Speech and Audio Processing 2 (1994), p133-150.*

[3] Bailly, G. 1997. ‘‘Learning to speak. Sensori-motor control of speech movements,’’ *Speech Commun. 22, 251–267.*

[4] Guenther, F. H. 1994. ‘‘A neural-network model of speech acquisition and motor equivalent speech production,’’ *Biol. Cybern. 72, 43–53.*

[5] Guenther, F. H. 1995. ‘‘Speech sound acquisition, coarticulation, and rate effects in a neural-network model of speech production,’’ *Psychol. Rev. 102, 594–621.*

[6] Howard, I. S. & Huckvale, M. A., ‘‘Learning to Control an Articulator Synthesizer by Imitating Real Speech’’, *ZASPIL, Franco-German Summer School, Lubmin, Germany 2004.*

[7] Kuperstein, M. 1988, “Neural models of adaptive hand-eye coordination for single postures”. *Science, 239, 1308-1311.*

[8] S. Maeda, in "Speech production and speech modelling" (W. J. Hardcastle and A. Marchal, eds.), p.131-149. Kluwer Academic Publishers, Boston, 1990.

[9] Holmes, J.N., “The JSRU Channel Vocoder”, *Proc. IEE, 127, Pt. F, 53-60. (1980).*

[10] Nabney, I. and Bishop, C. 1995. Netlab: Netlab neural network software. *http://www.ncrg.aston.ac.uk/netlab/.*

[11] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986) Learning representations by back-propagating errors. *Nature 323: 533-536.*

[12] Huckvale, M. A., and Howard, I. S., “Teaching a vocal tract simulation to imitate stop consonants”, *Interspeech 2005.*

[13] Wu, W., Shaikhouni, A., Donoghue, J. P., Black, M.J. (2004), Closed-Loop Neural Control of Cursor Motion using a Kalman Filter, *Proc. IEEE Engineering in Medicine and Biology Society*, pp. 4126-4129