SPEAKER IDENTIFICATION IN REVERBERANT ENVIRONMENTS

Aneesh Vartakavi *Iman Mukherjee* *Yan-Ling Chen* *Yonatan Sasoon*

Georgia Tech Center for Music Technology

ABSTRACT

The goal of this project was to explore Computational Auditory Scene Analysis (CASA), specifically, blind source separation in reverberant environments. Additionally, speaker identification, vowel classification and speech generation were also explored.

1. INTRODUCTION

Auditory Scene Analysis, coined by Albert Bregman, is a model that explains human auditory perception. When a listener hears two or more sounds, which overlap in time, they are able to separate the sources of the sounds. The cocktail party problem is commonly cited in literature, where a human is able to understand speech even under noisy conditions. This process of segmentation, grouping and streaming, which humans perform without conscious effort, is a complicated task for a computer. The field of 'machine listening' is formally called Computational Auditory Scene Analysis.

An important problem drawing lot of attention from the signal processing community is Blind Source Separation (BSS). This is the separation of sound sources without a priori knowledge of the nature of the sound itself. While BSS imposes no restriction on the number of different observations of the system, CASA restricts the number of observations to two, similar to human hearing.

One of the special cases of BSS is Independent Component Analysis (ASA). Blind Source Separation can be formulated as finding a linear representation in which the components are statistically independent [2]. ICA solves this problem by assuming signals are non-gaussian in nature, and can give very good results when its underlying assumptions are correct. The reader is referred to [3] for more information on ICA.

It is known that reverberation degrades speech intelligibility and performance of Artificial Speech Recognition (ASR) systems. The demand for robust speech enhancement algorithms has increased tremendously in recent years, many methods such as inverse filtering and prediction residual enhancement have been developed, but few methods have been developed to a practical level [4]. Reverberation also provides a significant obstacle in BSS problems, as multiple echoes of the source signal overlap in both time and frequency.

The Fourier Transform has some inherent disadvantages; good time and frequency resolution simultaneously is not possible. The Wavelet Transform (multiresolution analysis in general) overcomes this disadvantage, and has been applied to problems in vastly different fields. The cross wavelet transform (XWT) is used in the analysis of two separate time series together, the cross wavelet power can reveal regions with high common power. We attempt to follow in the footsteps of [5], using the XWT to localize the initial reflections of the reverberation.

When dealing with the problem of BSS in the context of speech signals, approaching it using speaker identification might prove useful [11]. Several methods have been proven to deal with speaker identification with great success. Extracting formant frequencies of speakers have proven reliable in differentiating male and female speakers (but also with a single speaker in different conditions) [12], along with Linear prediction Coefficient extraction [13].

For our discussion, we apply a few major restrictions on the type of phonemes we allow in order to utilize these methods successfully. Nasals pose an issue since the standard all-pole model of speech couldn't fit, and zeros need to be introduced. Fricatives are difficult to recognize as well, as they would require higher frequency information (6-8 kHz and above). Omitting these will allow us to use formants and Linear Predictions Coefficients (LPC), for speaker identification. Such method will require speaker training for several voiced and un-voiced phonemes and save the location of formants and extract LPC coefficients. These, in turn, will be used to compare the real-time or pseudo real-time speaker identification system.

Artifical Neural Networks (ANN's) are a mathematical model inspired by biological neural networks in the brain and the way it learns and maps information. It is largely, an interconnection of nodes (or neurons) that are weighed in a way to find patterns in data to map the input set of features to an identified output. A range of possible inputs are fed into the model and adaptively iterating over initial weight values and modifying them based on the error signal converges to a trained neural network that can be used to classify an arbitrary set

of similar input. We intended to classify vowels and the human speaker using ANN's.
Speech synthesis is the artificial production of human speech. Such systems attempt to maximize the naturalness and intelligibility. That is, to make the output sound close to human speech and easy to be understood. Concatenative synthesis and formant synthesis are the two major technologies for generating synthetic speech waveforms.

Concatenative synthesis is realized by concatenating short samples of recorded sound in a rule-based system. While concatenative synthesis can produce natural sounding speech, audible glitches sometimes arise from the differences between natural variations in speech and the nature of automated techniques for segmenting the waveforms.

We decide to use formant synthesis, which does not use human speech samples at runtime. A formant is a resonance in the voice spectrum. A single formant may thus be modeled using one second-order filter section. Passing a buzz source through only two or three formant filters can simulate the different vowel sounds of speech. Parameters such as fundamental frequency, voicing, and noise levels are varied over time to create a waveform of artificial speech. Thus, the pros of formant synthesis are that the speech is fully intelligible through the telephone bandwidth.

The organization of the paper is as follows: Section *two* reviews several important works that relate to the algorithms used in the system. In section *three*, the system description explains the design of the microphone used for input, the classification algorithm, the reverberation and wavelet analysis for BRIR and ICA and concatenative synthesis for the speech generation. Section *four* will present graphic and numerical results that will complement the demos and wave files that will be presented in class. Following the result is a discussion in section *five* of the algorithms and results, along with some proposals for future work in section *six*.

2. RELATED WORKS

Blind source separation is a very generic problem explored in several different areas. Be it biomedical signals like EEG, MEG, fMRI etc., or hyperspectral imaging, source separation is a way to analyze data from multiple independent sources picked up by single/multiple sensors so as to better understand the system and deal with the data modularly. Separation is usually based on independence (non-gaussianity, non-stationarity, non-whiteness etc.). But, basically the question it tries to answer is for X=A.S given X, can we find A and S? It involves using of prior information on A and S. However, the term 'blind' implies we have no precise information but only certain statistical assumption about A and S is available. Previous efforts to do blind source separation involve various methods in terms of Principal Component Analysis (PCA), Independent Component Analysis (ICA) [7], Multiple decorrelation (solution by generalized Eigen values) [6, 10], Multi User Kurtosis etc. PCA or Karhunen-Loeve transformation involves transforming data to a feature space consisting of the "main" features (principal components) that account for the majority of data. Iteratively, projecting the data into the direction of a subset of eigenvectors as per maximal covariance leaves us with transformed data that stands source separated. ICA, further, assumes a linearly mixed signal having inputs that are statistically independent and non-Gaussian. [10] It looks to provide a rather general and unified solution that summarizes the conditions for blind source separation by formulating a generalized Eigen value decomposition that diagonalizes simultaneously both - the covariance matrix of the observation data and an additional symmetric matrix depending on the assumptions. It points out various assumptions (non-gaussianity, non-stationarity, non-whiteness) that help recover mixed signals from similar sources (EEG, speech). Doing source separation "blindly" also sometimes refers to having no prior training. [11] introduces an algorithm that suggests a set of necessary and sufficient conditions that involves the kurtosis and the covariance of the outputs and is called Multi User Kurtosis (MUK). It assumes the inputs to be mutually independent, i.i.d and non-Gaussian that undergo linear mixing corrupted by Inter-User Interference (IUI).

3. SYSTEM DESCRIPTION

We devised a setup with two humans speaking at the same time into two microphones, which would capture the signal with reverberation from the room they are physically present in. As a simplifying assumption, the speech is restricted to sustained vowel sounds. An XWT is performed on the signal, and the early reflections are localized in time. Then a speech enhancement procedure (inverse filtering or spectral subtraction) is performed. Then ICA is performed on the signals, and the speech from the two speakers is isolated. The MFCC's of the isolated signals is then calculated, and they are fed into an ANN for speaker and vowel classification. After detection, the formants of the vowels are to be sent to a rudimentary speech generator, which is then heard through a pair of speakers.

We divided the project into independent sections, which we discuss in the following subsections.

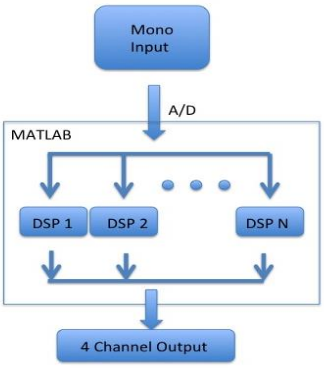

Figure 1. High-level system description

An initial idea for the project was to design several ‘processing units’ (panning, Doppler effect, etc) and apply them sequentially to this input audio, and played back through four channels that are spread in a room. This configuration allows exploration of several processing algorithms and concepts that pertain to 3D audio rendition. Some progress was made in this direction, before this concept was abandoned in favor of our current setup.

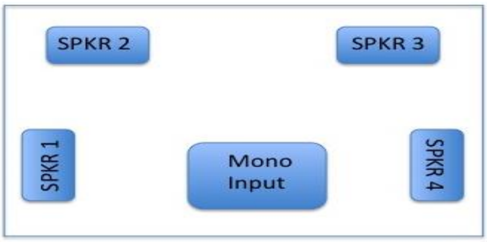

Figure 2. Speaker Locations

The rest of the section is dedicated to the description of the processing unit. Each unit is unrelated to the other and uses the four output channels independently.

3.1.1 Microphone Design

The idea behind designing a microphone (and a microphone pre-amplifier) was to experiment with real-life conditions with respect to equalization, experiment with the junction between the analog microphone and the digital processing, and try to achieve a flat frequency response in different situation in a more controlled fashion. A complete analog circuit was built, including a power supply, around a condenser microphone (electret CMA-6542PF), as can be seen in figure 3 (in the results section). The OP-AMP amplifying the input to the equalization circuitry is the AD797, an ultra-low distortion ultra-low noise operational amplifier. Such characteristics are crucial for amplification of several mV to line level input (the circuit should go up to 40 dB of amplification).

3.1.2 Response Equalization

In order to control the microphone response and make sure its response is adequate in all situations, a two-way equalization system was built using a low-pass filter and a high-pass filter, each with a separate potentiometer to control for the cutoff frequencies (and a separate potentiometer to control the volume). The intention behind the design is to get flat frequency response when both filter outputs are combined. A first order low-pass and high-pass filters were used both for ease of design but more importantly, the shallow responses after (low-pass) or before (high-pass) the cut-off frequency will allow a smoother transition when both signals are superimposed. Figure 2 shows the analog filter configurations, with the potentiometer. The right panel presents the high-pass filter circuit and the left panel shows the low-pass equivalent. It is easy to visually confirm that the resistor-capacitor network is a mirror image in the respective filters.

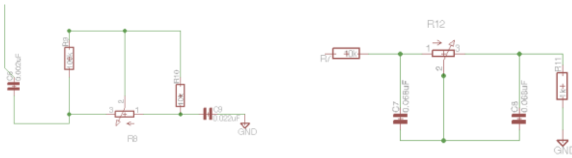

Figure 3 : Analog Filter Circuit

3.1.3 Circuit Analysis

We will show the analysis for the low-pass filter, as the high-pass is very similar. Assuming the frequency-dependant impedances of a capacitor and a resistor are Z()=1jc1;Z()=R respectively, such a circuit, assuming potentiometer at rest in its mid-point value of 25k, will effectively ‘see’ an equivalent resistance of 100k||50k + 50k||10k = 27kwith maximum and minimum resistance of 33k and 8k respectively (for both extreme ends of the potentiometer). Combined with two capacitors, and input coming from the op-amp and output at one end of the second capacitor, the system response is 11+jRC1 + (C1/C2). This will introduce a pole at z = (-1-C1/C2)/RC1, and after plugging in the values, we can steer the pole to be anywhere between ~300 Hz and 3000 Hz. A similar calculation will set the cutoff frequencies for the high-pass filter anywhere between 500 and 4000 Hz.

3.2. Speech Acquisition and Processing

Initial experiments were performed on speech samples that we recorded using a laptop-integrated microphone. We used two vowels (“a” and “e”), recording two samples each for four different speakers (three male, one female).

We used an ICA algorithm from the FastICA toolbox to attempt to separate the speakers.

3.3. Speaker and Vowel Classification

The MIR Toolbox [8] was used to analyze the audio, as it provides easy and fast batch processing of several features of interest. We used the MFCC, averaged across all frames of the audio sample as our feature of choice for our classification, because it provided an acceptable trade-off between accuracy and computational complexity.
We attempted a simple nearest neighbor classification of the MFCC data, and obtained encouraging results. We then implemented an ANN system for classification.

An ANN usually consists of an input layer that forms the nodes for the MFCCs in the context of our project. It has one or more hidden layers that give it the advantage parallel over linear programs. The number of layers depends on how linearly separable the inputs are. Usually, one hidden layer is sufficient for a majority of problems (2 or 3 only used when need to increase performance). The output layer is a set of targets that we map the MFCCs to. In our case, the targets are the speaker id or vowels. The neural network works on the basis of training sets that alter the weights of the interconnections between nodes in the mathematical model. In this project, we used the ANN Toolbox in Matlab that has predefined functions to specify inputs, targets, training and simulation of a neural network to do speaker id or vowel classification based on MFCCs from the speech samples.

3.4. Reverberation and Wavelet Analysis

Parallel and independent to the systems described above, a reverb simulation of a shoebox room was implemented. The simulation extended the Image Method [9] to simulate the acoustics, using the direction of the virtual sources and an HRTF database to simulate a Binaural Room Impulse Response (BRIR).
A cross wavelet transform was performed on the BRIR to localize the reflections. The Cross wavelet and wavelet coherence package for Matlab by Aslak Grinsted, John Moore and Svetlana Jevrejeva was used.

3.5. Speech Generation

Formants are the resonant peaks in the speech spectrum that the ears are most sensitive to when identifying human vowels. The frequencies of these peaks correspond to resonant frequencies of vocal tract, through which glottal pulse is filtered. Based on these characteristics of vowel sound production mechanism, a band-limited impulse train can be used as a glottal source. [11] In order to generate a vowel sounds; we filter the glottal source by multiple resonators with corresponding formant frequencies and bandwidths. First, we use Matlab to implement formant synthesis technique that output a simple /ah/ sound. The code first synthesizes s bandlimited impulse train as a source model. Then it computes the speech vowel with three resonant peak frequencies that correspond to the first three formant frequencies of /ah/ sound. Finally, it performs linear predictive coding (LPC) to encode the output speech.

Next, we try to implement the time-varying digital filter that simulated the vocal tract producing sequential vowels [13]. The simulation is aimed to generate sequential vowels with a natural transition and output a time-varying vocal tract. The modification of the digital filter coefficients is required as the glottal pulse train excited the vocal tract model. The procedure allows the controlled movement of the poles associated with one vowel into the correct position for the second vowel. In order to create the time-varying effect, the vocal tract parameters are adjusted by using a first-order discrete-time system. The system has the initial state corresponding to the initial pole angle and radius and the final state corresponding to the final location. The transition interval from one vowel to the next is selected according to the natural vocal change. The simulation chooses a /u/-/i/ vowel combination as the goal.

Besides the totally formant synthesis, we also use VOICEBOX in our system. VOICEBOX is a speech processing toolbox consists of Matlab routines that are maintained by and mostly written by Mike Brookes, Imperial College, UK. The “sapisynth” function serves as a text-to-speech synthesis of a string or matrix entries. It is depended on the Speech Application Programming Interface (SAPI), an API produced by Microsoft for Speech Recognition and Speech Synthesis. The “sapisynth” function provides various parameters to adjust the speech output. We can specify the speaking rate from -10(slow) to +10(fast), the pitch -10 to +10, and the volume.

4. RESULTS

In this section, we present the results of the systems described in the previous section.

4.1. Microphone and Signal Acquisition

We built and tested a mono microphone as described in section 3.1. We were able to interface this with a DAW (using the computers built-in ADC) and recorded voice samples.

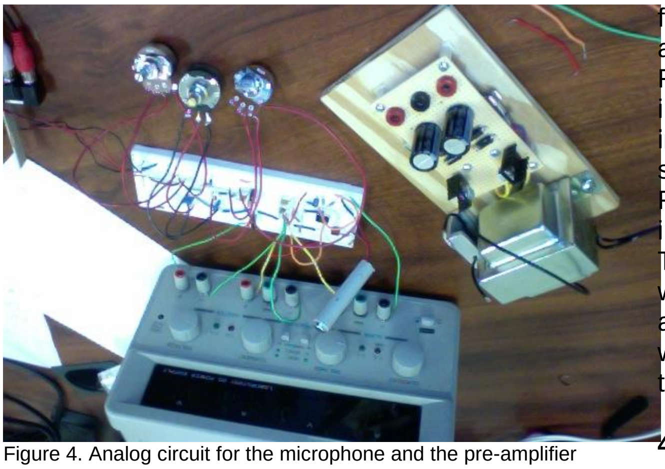

Figure 4. Analog circuit for the microphone and the pre-amplifier

4.2. ICA Analysis

The fastICA algorithm was used a mixture of two speech samples at different gains, simulating an idealized 2-channel mic setup. The plot of the wav files used is presented below.

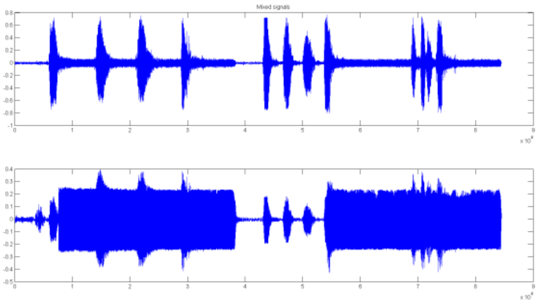

Figure 5: Mixed Signal input

The output of the ICA algorithm is plotted below.

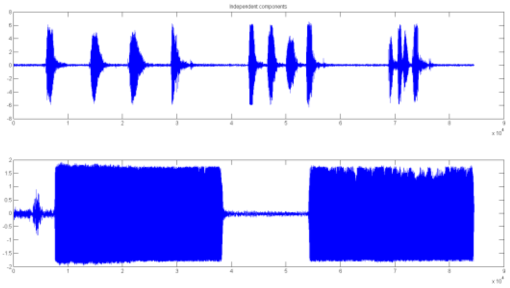

Figure 6: Separated signals

As described in section 3.3, we initially tried classifying vowels and speakers with a nearest neighbor classifier. The results in the following section were tested on un-mixed voice sample The classification was done using the ANN toolbox in MATLAB. The feature vector input were the 13 MFCC coefficients generated from the MIR Toolbox and were classified using a feed forward neural network with the hyperbolic tan sigmoid (tansig) function as the back propagation training function and a pure linear weight training/bias function.

For speaker identification we used samples for 4 individuals for training purpose. And we used 2 individuals' samples to test the accuracy of the system.

For vowel identification we used samples of 4 individuals for 2 vowels, a and e, namely.

The classification accuracy for speaker recognition was about 71.43%, while vowel classification was about 50%. It is important to note that our database was small, and a larger database with more thorough calculations could change these figures.

4.4. Reverberation and Wavelet Analysis

The image method was used to compute the BRIR. An example output for a room of dimensions 10m X 10m X 5m with the source at (7m,8m,1m) and mic at (1m,1m,3m) is presented below.

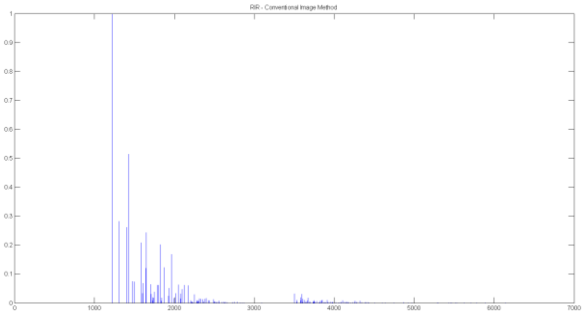

Figure 7: RIR using Image Method

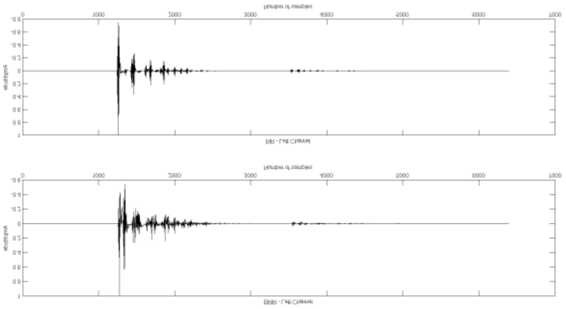

Figure 8: BRIR

This BRIR was analyzed using an XWT, with scales from 1:32 and using Morlet and Paul Wavelets. It can be seen that the Paul wavelet has better time resolution than the Morlet Wavelet.

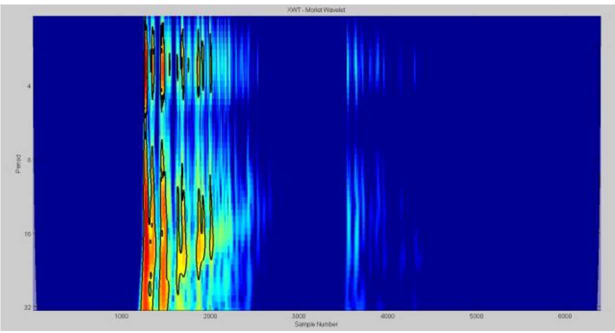

Figure 9: XWT - Morlet Wavelet

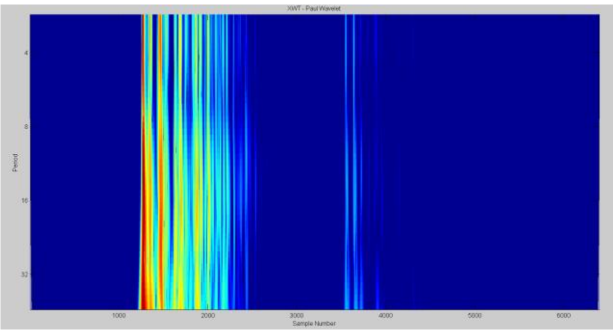

Figure 10: XWT - Paul Wavelet

The sum of the coefficient magnitudes was taken and plotted. It can be seen that the early reflections are extracted from the BRIR (refer to figure 4)

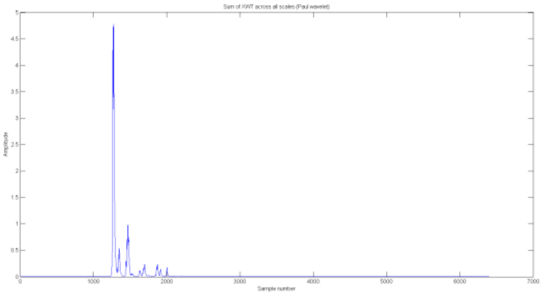

Figure 11: Sum of XWT coeffecients - Paul Wavelet

4.5. Speech generation

Figure 8, 9, and 10 shows the implementation plots of the formant synthesis technique that output a simple /ah/ sound. Figure 8 shows the waveform and spectrum of band-limited impulse train.

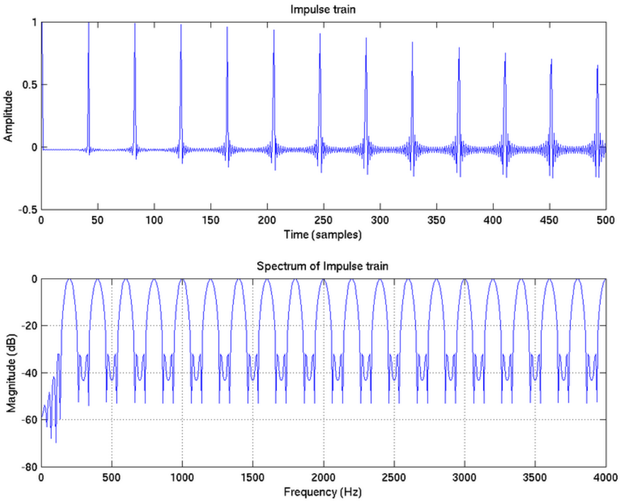

Figure 12: Waveform and spectrum of band-limited impulse train as a source mode

After computing the speech vowel with three resonant peak frequencies that correspond to the first three formant frequencies of /ah/ sound, the frequency response of the filter model is shown in Figure 9. Figure 10 shows the waveform and spectrum of resulting output signal after performing the linear predictive coding.

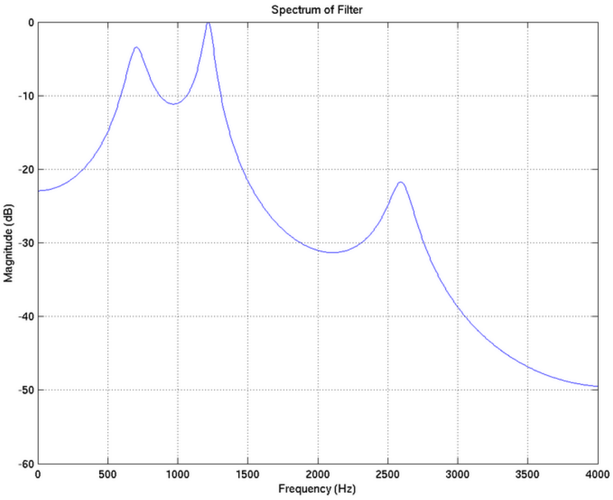

Figure 13: /ah sound, frequency response of the filter model with three resonant peak frequencies

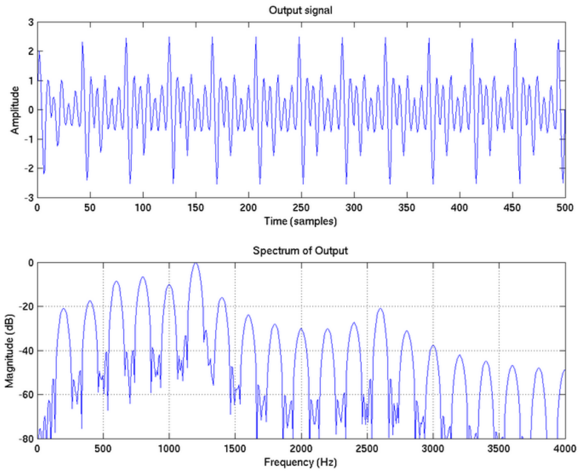

Figure 14: /ah/ sound, output signal and its spectrum

For the time-varying multi-vowel synthesizer, the vowel pair /u/-/i/ simulation result is shown in figure 11. The transition into the /i/ vowel begins at time 12 ms and is over til 60 ms. The first two cycle in figure 11 represents the steady state vowel, /u/, while the last cycle represents the steady state vowel, /i/.

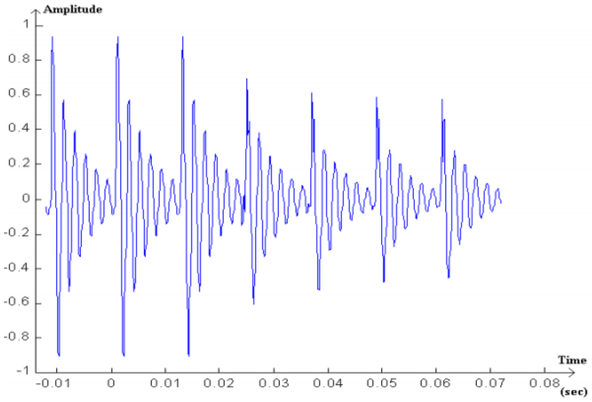

Figure 15: Speech segment during transition from /u/ to /i/

The systems based on formant synthesis technology usually generate artificial and robotic-sounding speech so does ours. However, it is still interesting to see not complicated filter programs producing the intelligible speech sounds.

5. DISCUSSION

Our work attempted to connect independant efforts from multiple fields; we are not aware of any previous attempt of a similar nature. We went through many iterations of the goals of the project before roughly settling on this one. Time was the most important constraining factor, resulting in validations of concept of the independent portions of the project, not completely integrated into a complete working prototype as we had originally intended.

The analysis of Room Impulse Responses with Cross Wavelet Transforms, proved to be difficult in practice with our rudimentary knowledge of wavelets. The estimation of the room without a priori knowledge is non trivial, and some methods for speech enhancement were found which bypass this problem with simplifying assumptions. Speech enhancement in reverberant environments is considered difficult by many, and we were unable to integrate literature in this field in our work.

Overall, we feel that the project gave us the opportunity to explore and attempt to integrate various fields of interest like analog electronics, spatial audio and CASA.

6. FUTURE WORK

The broad nature of this work opens up many areas for future improvement. A feature vector with many features can be used instead of just MFCC as we worked with. Different classifiers could be implemented and tested, larger training and test datasets could improve classification results. The simulation for the BRIR system assumed many ideal assumptions, improvements could be made to produce more realistic responses. There is much to be explored in the fields of dereverberation, speech enhancement and CASA, new insights could be integrated into this system.

7. REFERENCES

[1] Bregman, A. S. Auditory scene analysis. MIT Press: Cambridge, MA, 1990)

[2] A. Hyvarien et al. Independent Component Analysis, John Wiley and Sons, 2001

[3] A. Hyvarien and Errki Oja, “Independent Component Analysis : Algorithms and Applications”, Neural Networks, 13(4-5):411-430, 2000

[4] Takuya Yoshioka, “Speech Enhancement in Reverberant Environments”, Kyoto University, 2010.

[5] Sampo Vesa, Tapio Lokki, “Detection of Room Reflections From A Binaural Room Impulse Response”, DAFx 2006

[6] Weinstein, E. and Feder, M. and Oppenheim, A.V, "Multi-Channel Signal Separation by Decorrelation", IEEE Trans. Speech Audio Processing, vol. 1, no. 4, pp. 405-413, April 1993

[7] Roberts, S., Everson, R. "Independent Components Analysis : Principles and Practice" Cambridge University Press, June 2001

[8] Oliver Lartillot, Petri Tovivainen, “A MATLAB Toolbox For Musical Feature Extraction From Audio”, DAFx, 2007.

[9] Jont B. Allen, David A Berkeley, “Image Method for efficiently simulating small-room acoustics”, J. Acoust. Soc. Am. Vol. 65, No. 4, April 1979.

[10] Lucas Parra, Paul Sajda, "Blind Source Separation via Generalized Eigenvalue Decomposition", Journal of Machine Learning Research, vol. 4, pp. 1261-1269, 2003

[11] Cook, P.R. “Identification of Control Parameters in an Articulatory Vocal Tract Model, with Applications to the Synthesis of Singing”, Ph.D. thesis, Elec. Engineering Dept., Stanford University (CCRMA) 1990

[12] C. S. Burrus, J. H. McClellan, A. V. Oppenheim, T. W. Parks, R. W. Scafer, and H. W. Schuessler, “Computer-Based Exercises for Signal Processing Using MATLAB”, Prentice-Hall, 1994, p.336

[13] Paul J. Coyne Jr., “Constructing a Time-Varying Multi-Vowel Synthesizer in MATLAB”, Department of Electrical Engineering and Engineering Science, Loyola College in Maryland

[14] M.R. Sambur, “Selection of Acoustic Features for Speaker Identification”, IEEE transactions on Acoustics, Speech and Signal Processing, vol. 23, no. 2, 1973.

[15] M.R. Sambur, “Speaker recognition and verification using linear prediction analysis". Ph.D dissertation, MIT, 1972.

[16] J.J. Wolf, “Efficient acoustic parameters for speaker recognition”, J. Acoustical Society of America, vol. 51, pp. 2044-2056. 1972