Propagation of Speech Sounds in SPREAD

Jiali Sheng

Advisor: PengFei Huang, Mubbasir Kapadia, Norman Badler

University of Pennsylvania

ABSTRACT

SPREAD is a novel agent-based sound perception model that was just presented in the paper "SPREAD: Sound Propagation and Perception for Autonomous Agents in Dynamic Environments". Although SPREAD tests the 100 different environmental sounds, it has never been tested on speech sounds. Speech sounds are important in a perception system to support character animation and crowd simulation because the majority of human interaction happens when we are speaking to each other. The range of possible signals for speech sounds are much smaller, as a result, Human listeners are much more sensitive to the tiny nuisances. This project seeks to produce a pipeline for SPREAD so it can support speech sounds.

Project Blog: http://jollysound.blogspot.com/

1. INTRODUCTION

Imagine a virtual reality that you can interact with through speech. When you speak through the microphone, your speech will propagate through the virtual environment and reach the other characters. If you spoke another character's name, their reactions are based on their relative position from you.

GOAL: The goal of this project is to produce a functional pipeline with SPREAD that will simulate real time (or close to real time) propagation of speech sounds.

BACKGROUND: SPREAD is a novel agent-based sound perception model, but so far, it has only been tested on environment sounds. Speech sounds are much more similar to each other than environment sounds, which means they are much more difficult to differentiate between.

MOTIVATION: In the real world, one of the most forms of communication is speech sounds. Through talking and communicating, one person can trigger a different set of behaviors in another person. SPREAD is a system that tries to use sound propagation as a means of animating agents in a virtual environment. With the ability to propagate and perceive speech sounds, we are creating the opportunity to simulate a new and bigger set of behaviors in virtual agents.

FEATURE: This project features a working pipeline with SPREAD that propagates speech sound using the phonemes approach. The demo of this project allows the user to speak into a microphone the names of a few characters in the scene. After the sound has been propagated, the agents (if recognized their name has been called) will turn red.

2. Method

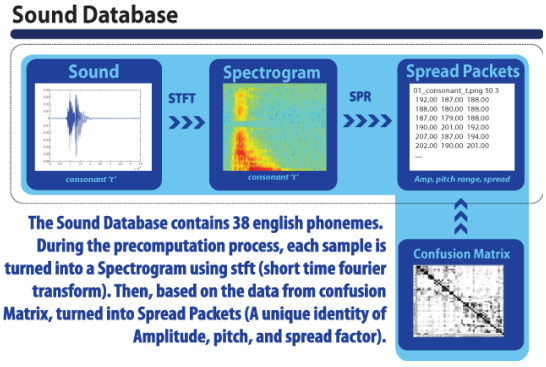
The Approach of this pipeline will be called the "phonemes" approach. Phonemes are the smallest units of speech sound, and American English has about 44 different phonemes. If we can pre-compute SPREAD packets for each individual phonemes, then we should be able to simulate the propagation of whole words by combining the phonemes together.

2.1 Phonemes

There are 38 different phonemes in this project. And 2 sets of recorded data. One set was recorded manually (pronounced by a native-english speaking friend). The other set was taken from the website "A Course in Phonetics" by Peter Ladefoged (Ladefoged).

2.2 Database

The phonemes set is then passed through a piece of MATLAB code that first calculates the spectrogram with uniform sampling (discussion on future works with logrithmic sampling in section 4.1). The Spectrogram then goes into a piece of the original SPREAD pipeline (called SPR), which generates the spread packets.



The "SPR" part of the pipeline also takes in a Confusion Matrix of information regarding sound similarities in a database. This confusion matrix is generated from the paper "Consonant And Vowel Confusion Patterns By American English Listeners" by Andrea Weber. The confusion matrix is taken, normalized, and altered so it would fit the number scale and criteria for SPREAD.

2.3 Input

This project uses Microsoft Speech SDK for sound input processing. Because UNITY is only compitable with .NET 2.0, it was not able to use the Microsoft Speech SDK directly because the speech SDK is built on .NET 2.5. The error occured in the package " Microsoft.Speech " and " Microsoft.Speech.Recognition ".

As a by-pass, an external server was set up (in the same computer) to perform the necessary tasks for voice recognition. It will take the best guess at the word spoken by the user and pass the string to the Unity engine.

The code in unity has a dictionary of strings to phoneme sequence it will look up for propogation.

2.4 Propogation

Propogation for SPREAD is unchanged except for the multi-phoneme support. The original pipeline will propogate on repeat one packet upon program start. The controls for packet selection needs to be changed by hand. New code has been added to support voice controls and propogation of multiple packets.

A SPREAD scene is a 2d grid. Each grid contains sound information, and gets updated at each time step. Propogation works that during every time step, the sound travels up, down, left, and right one cell. The sound package (originally one amplitude and three frequencies) will become lower in ampliture, and distribute the frequencies so it is a "spread" around the original three.

2.5 Recognition

Recognition is through the Dynamic Time Wrapping algorithm. Rankings on matches are determined by a binary hierarchy that contains information about . This hierarchy is first generated from the confusion matrix described in section 2.2.

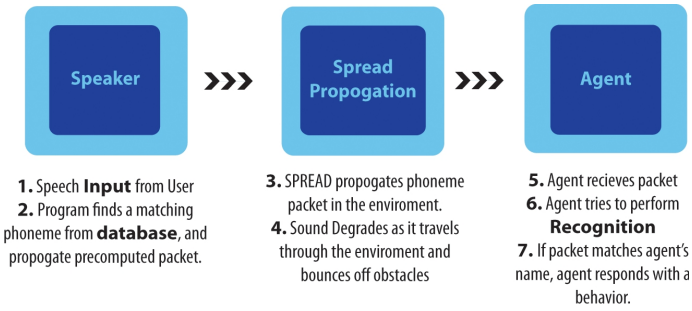
The algorithm that does the generation is an altered form of the "Huffman-encoding" algorithm. The Algorithm takes the data in a confusion matrix (n x n), and creates one node for each element (in total of n nodes). Based on similarity data, each iteration of the algorithm combines two nodes into one parent node. The binary tree is complete after n-1 iterations,.and contains 2n-1 nodes.

After the automatric generation of the binary tree, there is a manual editing step. The generated tree is manually compared to the phoneme Hierarchy from the paper "An Online Algorithm for Hierarchical Phoneme Classification" by Ofer Dekel.

3. RESULTS

In this version of the pipeline. The user is able to speak a word from a set dictionary. The dictionary maps the spoken word to a set of phonemes and will propagate the phoneme

3.1 Pipeline



3.2 Database

The Database built using the sound from the website "A Course in Phonetics" performed a lot better than the Database built with recorded phonemes. This is probably because the recording hardware also recorded handling noise and ambient noise. It could also be due to the fact that perhaps the phonemes were not being pronounced correctly.

3.3 Demo 1

The first version of the pipeline can be seen in this demo: http://youtu.be/hfCMAn8_8x8

In this version, individual phonemes are being recognized by the SPREAD system. It should be noted that some phonemes perform a lot better than others. In general, vowels perform better than consonants. This result is expected because vowels contain more pattern, and are generally longer in duration. Each different vowel features the same three strong frequency bands but at different ratios each. Consonants are generally characterized by a short burst of "noise" (positive amplitudes across a wide range of frequency bands). This makes them hard to distinguish between each other.

3.4 Demo 2

The second version of the pipeline can be seen in this demo: http://youtu.be/p17dvv0xD2k

In this version, a dictionary of words to phonemes are added. This is the foundation for being able to propagate multiple phonemes.

3.5 Target Platform

This program operates in the Windows environment. The entire pipeline is in many different languages: The database is generated via Matlab; The propagation and recognition is in C# and c++; The tree is generated with Java.

4. FUTURE WORK

Although the pipeline is currently in working condition, there are many things to improve. Listed are some of the potential areas of improvement.

4.1 Logrithmic Sampling

The human ear notices intensity changes at a logarithmic scale. Perhaps a way to improve recognition accuracy is to implement a more accurate biological model where sampling phonemes happen at a logarithmic scale.

4.2 Larger Word Database

The pipeline requires a diction of words to phonemes to be present in order to perform propagation correctly. This means that either a parser for an existing dictionary (like dictionary.com) needs to be added into the pipeline, or a bigger dictionary needs to be added.

4.3 Language Perception Models

People are actually worse at discerning between phonemes than we appear. The reason is because we are very good at guessing what the word we hear should be. Numerous research has been done on how to simulate the perception models with statistics. It may make this project a lot more realistic if a Language Perception Model was applied at the "perception" stage of the pipeline.

4.4 Different Languages

This project focuses on American English and it's 44 different phonemes. In actuality, there are about 150 + different phonemes for all the different languages in the world. It may be interesting to apply this pipeline to different languages.

4.5 Ambient Noise

In any environment, there will be ambient noise. This ambient noise can potentially interfere with hearing. To simulate an accurate environment, perhaps we should also add ambient noise into the system.

5. CONCLUSIONS

The result of this project is a working pipeline for speech sound propagation in spread. Although it works, there are still many areas of improvement. Through the process, I learned a lot about programming sound, sound recognition, and how speech recognition works. Through researching the different techniques used in sound analysis, I am starting to have an intuition on how to solve problems evolving sound, and sound recognition.

4. References

Ladefoged, Peter. "IPA Chart." *A Course in Phonetics*. University of California, LA. Web. 22 Feb 2013. <http://www.phonetic-s.ucla.edu/course/chapter1/chapter1.html>.

"Consonant And Vowel Confusion Patterns By American English Listeners" - Weber, Andrea - (2003)

"An Online Algorithm for Hierarchical Phoneme Classification" - Dekel, Ofer - (2004)

Other References

"Music and Computer" - Repetto, Douglas. <http://music.columbia.edu/cmc/musiciandcomputers/>

"The Voice in the Machine: Building Computers That Understand Speech" - Roberto Pieraccini.

"Audio Ancedotes: Tools, Tips, and Techniques for Digital Audio" - Greenebaum, Ke.

"Computational Methods in Acoustics" - Kristiansen, R Ulf.

"Interactive Physically-based Sound Simulation" – Raghuvanshi, Nikunj.

"The Sounds of Language" - Henry Rogers.

"Talking Soundscapes: Automatizing Voice Tranformations For Crowd Simulations" - J Janer

"Keyword Spotting Based on Phoneme Confusion Matrix", Zhang, Pengyuan

"Patterns of English phoneme confusion by native and non-native listeners" - Cutler, Anne