EXPERIMENT/IMPLEMENTATION PLAN TO DEMONSTRATE TIME-ORDERING NEURAL NETWORK TECHNOLOGY FOR USE IN HIGH-QUALITY SPEECH RECOGNITION SYSTEM

March 31, 1989

OBJECTIVES/SUMMARY

The objective of this series of experiments is to demonstrate the usefulness of the proprietary time-ordering neural network developed by Dr. Walsh for recognizing and processing actual speech sounds. The experiments are designed to exercise those features of the model judged critical for the implementation of a large vocabulary, continuous speech, speaker independent speech recognition system. The success of these experiments will imply that this neural network technique can be used to develop a successful high-quality speech recognition device.

The endpoint of this series of experiments will be a device which echoes speech. The user will speak into a microphone, and the neural network will process the speech to recognize the position of the muscles in the speaker's vocal tract at various points in time. The vectors describing these muscle positions will then be transmitted to an articulatory synthesizer--a model of the human throat--which will speak the sounds. If implemented on an IBM-PC as proposed because of cost considerations, this demonstration system will be slower than real time.

This system is not like a tape recorder. It is like a child or a speaker of another language listening to your speech, figuring out how to say the sound you are making, and repeating that sound back to you in his or her own voice. The output of the articulatory synthesizer will repeat the same words as the speaker, but the voice will be different since the synthesizer will model a differently shaped vocal tract than the speaker has.

In addition to serving as an experimental application of the new neural network model, this system can be used as a bit-rate reduction system for voice coding and telephony, and so may have direct commercial application. The construction of this system is also a necessary step in the development of a full-scale speech recognition system.

NOTE: This series of experiments has been designed within the constraint of a budget of approximately $50,000. A salary for Dr. Walsh and some equipment costs (A/D and D/A convertors) are included, but expenses such as office space and legal costs are not.

PLAN

1. Implement an ear-modeling spectrum analyzer in C on the IBM-

PC using the time-ordering neural network model. Determine and plot its frequency response.

Milestone: Frequency response to sine wave stimulation shows similar response curves to those of neurons in the ear; i.e. bandwidth increases with increasing frequency.

2. Select and purchase real-time D/A convertor. Implement an inverse spectrum analyzer (spectrum synthesizer) which synthesizes sounds using their spectrum as input. Interface to obtain an audible playback of the synthesized sounds.

   Milestone: When speech is decomposed into its spectrum and then re-combined by the spectrum synthesizer, significant speech information should be retained. E.g. different phonemes should sound audibly different on playback.

3. Check the quality of the Articulatory Synthesizer obtained from Haskins Labs by visiting an installation or Haskins Labs in New Haven, or by obtaining recordings of synthesized sounds. This synthesizer is written in Fortran for a VAX. If quality is acceptable, examine the source code and determine the best method of implementing the synthesizer--by porting to PC or Sun in Fortran, by re-coding algorithms in C on PC, or by obtaining a VAX. Obtain or develop a test suite and desired outputs.

   Milestone: Target machine and language selected. Capabilities and limitations of synthesizer software fully documented.

4. Implement articulatory synthesizer. Obtain target machine, compiler(s), hardware if necessary (cost of new target machine, compilers and hardware not included in $50k figure).

   Milestone: System compiles without error. System runs test suite producing desired output.

5. Implement "random number generator" software to exercise the articulatory synthesizer over all physiologically possible combinations of speech sounds, including utterance length constraints imposed by breathing. Consult linguist if necessary to determine additional linguistic constraints. Interface to articulatory synthesizer. Interface articulatory synthesizer to D/A convertor.

   Milestone: System produces realistic and varied speech sounds spontaneously.

6. Implement time-ordering neural network software in C on the PC. Develop suite of test cases and test.

Milestone: Neural net successfully learns and recognizes examples in test suite.

7. Interface spectrum analyzer, articulatory synthesizer, random number generator, and time-ordering neural network to each other. Train the neural network to recognize the muscle movements corresponding to a given speech sound. Do this by using the input from the spectrum analyzed articulatory synthesizer as the time-varying stimulus to the network, and the muscle movement vectors produced by the random number generator as the conditioned response.

   Milestone: The same muscle movement vector appearing in different contexts triggers the same group of associated neurons after training set is completed.

8. Buy a real-time continuous A/D convertor. Digitize a variety of speech samples from different speakers with different accents (from short wave radio, e.g.) and use to train network. Allow network to self-associate.

   Milestone: The same muscle movement vector appearing in different contexts triggers the same group of associated neurons after training set is completed.

9. Connect output of neural network to the input of the articulatory synthesizer. Use system to reproduce speech input to system.

   Milestone: System accepts input for variety of speakers and echoes what they say by means of the articulatory synthesizer.