HEARING POLYPHONIC MUSIC WITH THE CONNECTION MACHINE

Barry Vercoe

Music & Cognition Group
Media Lab
MIT
Cambridge, MA 02139

bv@ems.media.mit.edu
Tel: 617/253-7441

ABSTRACT

Although the human brain accomplishes multisource audio signal separation with apparent ease, there has been little progress in giving machines access to such information. This is due in part to the conceptual limits of traditional signal processing, but also to our not knowing the physical cues by which the brain separates the components of different audio sources. We describe a machine method that resembles how the ear and brain appear to process musical signals. Using a computer that employs massive parallelism, networks of AM and FM detectors enable correlation and grouping of sinusoidal components into separated spectra. The method smoothly combines classical signal processing and neural network processing, offering a form of acoustic intelligence that could enlighten a machine approach to audio processing tasks in general.

1. Introduction

Expressive information in music is contained both in the notes that are played and in the nuances of their acoustic performance. For a system to capture this from acoustic signals alone it must do fast, accurate detection of pitch and envelope. This is especially difficult when the signal is polyphonic.

Multivoice audio tracking is seemingly trivial for the human ear and brain. Because of the low effort required in perception, most human music cultures have

developed complex polyphonic or polyrhythmic traditions. If *machine* access to polyphonic acoustic data were so easy, research could focus on its semantic interpretation. However, this first level of understanding has eluded machines despite the best tools of classical signal processing.

We have encountered the problem previously in several forms. In early work in 1983 (with the late flutist Larry Beauregard of IRCAM) we developed robust methods for tracking, score matching and automatic accompaniment [Vercoe 1984]. The accompaniment proved to be highly responsive to subtle shifts in tempo and phrasing. However, pitch estimates were heavily dependent on optical sensing of the keys, and on the presence of a single acoustic fundamental. When the flute used special effects like multiphonics the tracking became impossible.

A subsequent phase of this project involved tracking violin from acoustic-only information [Vercoe & Puckette, 1985]. The goal was to understand extreme musical behavior, such as occurs in highly expressive or individualistic interpretation. As with the flute, the automatic accompaniment was predictive, making informed guesses to achieve simultaneity with the soloist. In addition, this system could also learn from rehearsals, gathering enough information about the interpretation that it could anticipate habitual idiosyncrasies. The essential violin tracking worked well provided the signal was monophonic. But when it was not (as with double-stops), acoustic-only tracking in realtime was rendered impossible.

2. Polyphonic Approaches

Multisource signal separation has been approached as spectral separation in work on additive co-channel speech. The techniques usually rely on the strict harmonic nature of voiced excitation, gaining separation or improvement either by harmonic magnitude (HM) selection [Parsons] or suppression [Hanson & Wong]; they are ineffective on unvoiced segments. [Naylor & Boll] used maximum likelihood estimation of the louder signal to determine its mode (voiced/unvoiced) and thus its best suppressor. [Childers and Lee] start with a theoretical estimate of each spectrum, then systematically improve it using minimum-cross-entropy between frames. None of these approaches would separate non-harmonic sources, such as whispered speech with a telephone ringing in the background.

Separation by simultaneous pitch estimate has been employed by [Amuedo], in which each peak in the spectrum asserts a hypothesis for each frequency that could be its fundamental, and the assertions are sorted to prescribe different sources. [Moorer] employed multiple levels of processing and grouping, such as

autocorrelation of the composite signal, time segmentation and filtering, and a heuristic method to determine the tones present. More recent approaches have employed the Bounded-Q transform [Schwede] and the Short Time Fourier Transform. The latter has been used effectively in talker interference suppression [Danisewicz & Quatieri], although the system still presumes harmonic partials.

A more general technique may be possible due to recent work by [McAdams]. He showed that if the harmonic partials of three vowels are summed with no vibrato or amplitude motion it is impossible to separate or recognize them; yet if any vowel's partials are given some coherent frequency motion, those partials will *fuse* into a separate recognizable vowel sound. The effect is especially strong if the motion also causes amplitude change, as when partials are being modified by steeply shaped resonances. This suggests that the important agents of signal separation are *amplitude and frequency motion detectors.* These must be neuronal groups (or agents [Minsky]) skilled at pattern matching amongst changes, in turn enabling others to group the partials responsible into likely distinct sources. Some neuronal groups apparently identify parallel random amplitude and pitch perturbations, while perhaps others sense secondary patterns (amplitude behavior due to formants) evident over time.

A computational method to investigate these issues would 1) divide the signal into component sinusoids, and develop *histories* of the amplitudes and frequencies of the most important components; 2) cross-correlate the histories to group those with parallel random micro-perturbations and parallel slower-moving frequency or amplitude motion; 3) auto-correlate the remainder to group those whose attribute changes have equal periods; 4) perform pitch estimation for each group. The stages represented here range from peripheral auditory processing, representable by classical SP models, to higher-level grouping that is perhaps best approached through neural analogies. To accommodate both, we have employed a processor whose architecture is inspired by the parallelism of neural processing.

3. The Connection Machine

The CM is a highly parallel computer of up to 65,536 processors, whose architecture resembles that of the brain more closely than other existing multiprocessors. Each processor has 4 kilobits of local memory, and its instructions operate on variables formed from bit fields in this memory. The simplicity of each node is in keeping with neuronal simplicity, and the inter-processor connection strategy is analogous to a real neural net, having greater local density and some global ability.

The CM is a single-instruction multiple-data (SIMD) machine, in which all processors operate from a common instruction stream. For one add instruction, 65536 actual additions can be performed in parallel. However, each processor is also controlled by a context flag. A processor whose context flag is set will execute all instructions, but when the flag is cleared certain operations will have no effect. For example, the absolute value of a parallel variable (pvar) is found by setting each context flag if its local pvar is negative, then sending a conditional negate; the variable is then non-negative in all processors.

The CM receives its instruction stream and new data from a host processor. Within the CM, data is transferred between processors by two main methods: a news network (of adjacent processors), and a general routing network. The news network views the processors in a square grid and allows each to send data efficiently to its four neighbors to the east, west, north and south. The routing network is a multistaged hypercube message passing system that allows any processor to send a message to any other processor. This takes an amount of time depending on the number of messages sent and the number of message collisions; it is slower than the news network.

4. Audio Processing on the CM

Although the power of CM computation resides in its parallel computation, most audio applications cannot use identical operations on arrays of say 16384 elements, and might instead call for several smaller tasks to be done in succession. These tasks could each be allocated a different group of CM processors and be resident simultaneously, but they would each need a different sequence of host-supplied instructions. Such serialization would underutilize the CM by keeping most processors idle. A solution (proposed by M.S.Puckette) is to organize the calculations so that each stage uses the same basic operations, i.e. design a basic cell in terms of which different algorithms can be described. The primitives of LTI signal processing (multiplication by a constant, signal addition, and unit delays) suggested the Universal Processing Element (UPE) of Carver Mead [Warwrzynek & Mead], and this was modified for use on the CM.

Figure 1 shows three processors and the operations of one basic cell during its "software cycle". Each UPE consists of three inputs (A,X,W) and an output (Y). In each processor's software cycle, A and W are multiplied and added to X to form the new output Y. Y can then be sent via the news or routing network to another non-local processor. Figure 2 shows a 4th order filter implemented with 9 UPE's.

The 3 extra news network W to W transfers (a total of 4) were added to the software cycle to implement the 4th order recurrence relation and spread the last output value to the other processors.

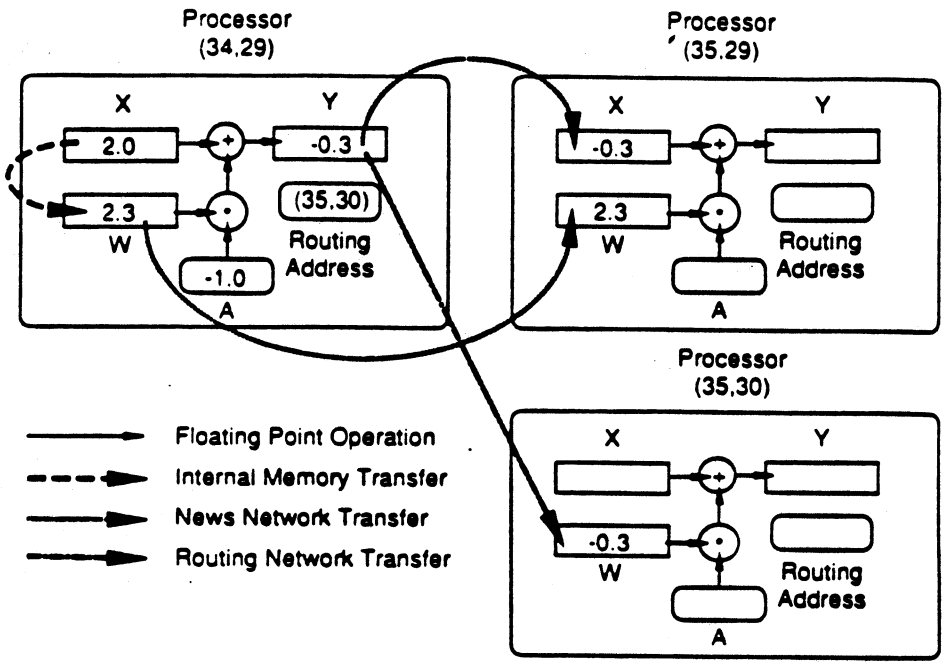

Figure 1. UPE operations available on each Software Cycle include three kinds of data transfer.

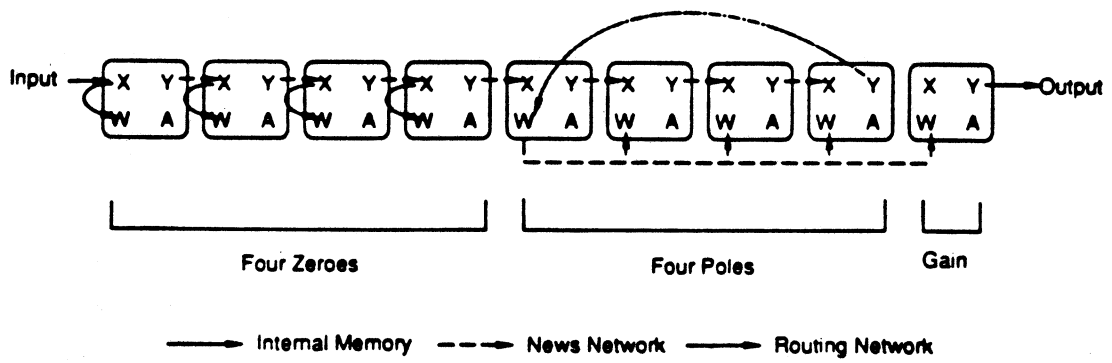

Figure 2. A four-pole IIR filter implemented with nine processors. It has a throughput of one sample per software cycle.

5. Signal Separation

We can now return to our strategy for polyphonic signal separation, outlined at the close of Section 2. We have completed the initial stages of this on the CM. The approach divides a complex mixed signal into sinusoidal components, and then identifies those whose similar micro-structure suggests a common source.

The signal is first analyzed using the Short Time Discrete Fourier Transform (STDFT), given by the equation,

$$S(\omega)=\sum_{n=0}^{N-1} x(n)w(n)e^{-j\omega n} \quad ,$$

where $x(n)$ is the sampled sound, and $w(n)$ is a window function. The purpose here is to separate the signal into sums of slowly varying partials,

$$s(t)=\sum_{n=1}^{N} (A_n+B_nt)sin(\omega_nt+\phi_n)$$

As can be seen, this model of sampled sound is different from the usual, adding an amplitude growth term and presuming no relationship between the frequencies of the partials. The sound is modeled as a sum of N sinusoids with constant frequency, phase offset, and a linearly varying amplitude over the analysis window. The growth term B greatly reduces the error of estimating signals, and allows signals to be better separated. For components whose peaks overlapped in the spectrum, an iterative subtraction technique was also developed that further aided separation [Cumming].

The idea is to compile a list of the frequency, amplitude, phase, and rate of amplitude change for each sinusoid in the frame. By comparing sequential frames of data, and noting the trends in magnitude, frequency-shift and phase, we can form consistent *histories* of the sinusoids over time. On the basis of common fluctuations in amplitude and frequency, the sinusoids can then be grouped into distinct sound sources.

As a check on the methods under development the combined sounds of a cello and flute were analyzed by researcher David Cumming during work on his thesis [Cumming]. Figure 3 shows the spectrum and waveform of the instruments playing an octave and a sixth apart. The flute (intentionally louder) has peaks visible at multiples of 550Hz, but there are several instances of interfering harmonics. Figure 4 shows the result of semi-automatic subtracting out of the flute, leaving the much quieter sound of the cello alone.

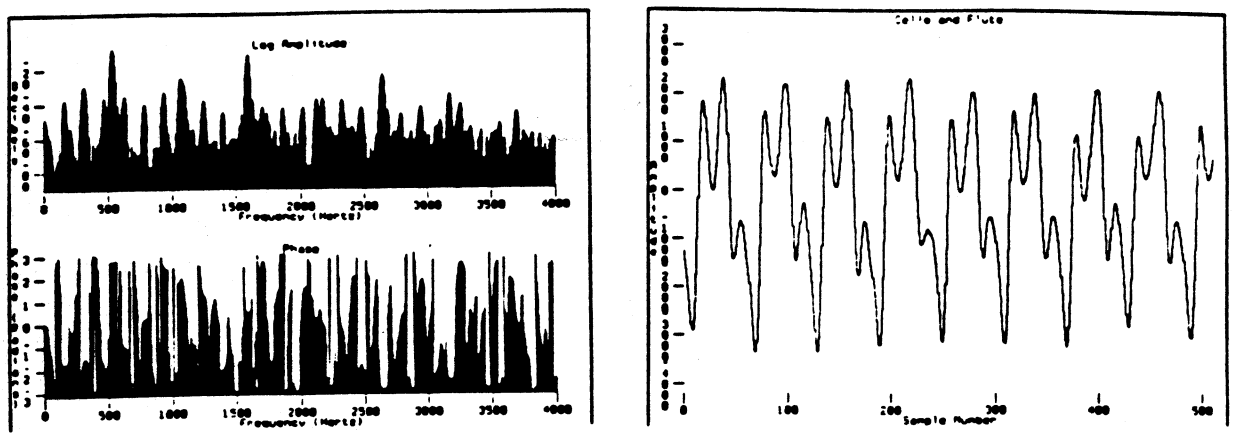

Figure 3. Spectrum and waveform of cello and flute an octave and sixth apart.

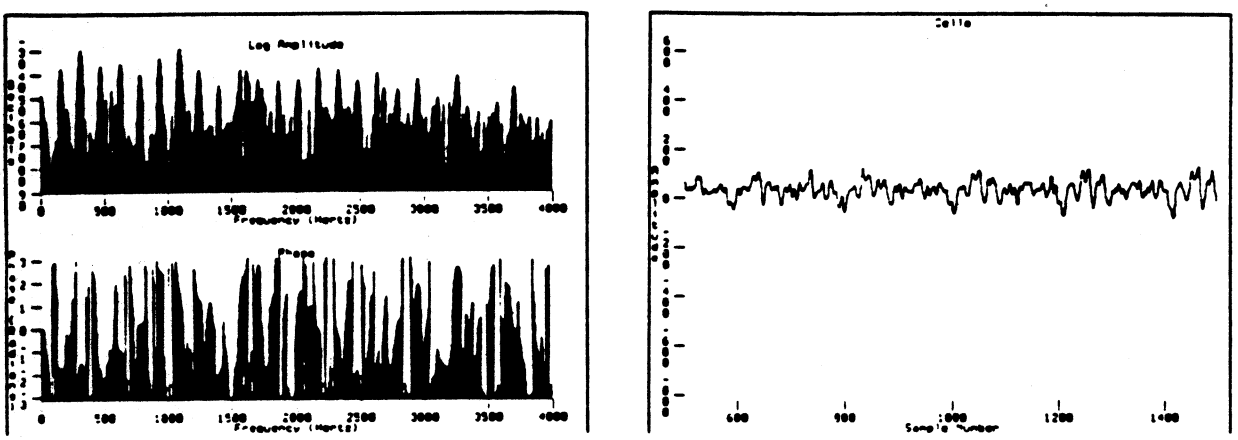

Figure 4. Spectrum and waveform of the cello alone after the flute has been subtracted out.

The information necessary for a fully automated version of this can vary. Common onset times are a strong grouping factor, as are constant frequency ratios, common phase jitter, or common amplitude modulation or periodicity (vibrato). Once the components have been grouped in this way, we can always test the grouping procedures by artificially reconstructing each source. However, the real objective of this separation—pitch and envelope estimation for each separated source—does not imply reconstruction, but further recognition. We next consider an approach to that based on even more parallelism.

6. A Neuronal Speculation

Although UPE's have enabled efficient modelling of auditory peripheral processing, the later stages described above have been less natural for the CM. Because the brain manages the grouping so effortlessly, a neural network approach to *post-peripheral* processing has seemed promising. There has been a recent resurgence in this field [Rumelhart & McClelland]. There has been notable progress in speech generation trained through error back-propagation [Sejnowski & Rosenberg], and these algorithms have also been implemented on a CM [Blelloch & Rosenberg]. Our interest here, however, lies in those processes immediately following spectral analysis by which the brain does grouping and signal separation.

We suspect that a key to this is early activation of *motion detectors* in both pitch and amplitude. The existence of neuron groups sensitive to motion is well established in vision [Hubel & Wiesel]. More recently, the auditory cortex of a cat was found to contain neurons sensitive to *frequency modulated* pitch, but not to the same pitch unmodulated [Whitfield & Evans]. Due to [McAdams] we can infer that these are critical to the perception and parsing of audio. We can also reason that output from the peripheral auditory system is sent to neural layers that implement *difference detectors* and *difference matchers*, and that these in turn will group the components into distinct percepts (separate spectra) ready for selective attention and recognition. How might such computation be accomplished simply? From the above we could postulate that the auditory cortex consists largely of AM and FM detectors, with interconnections that sense cross-correlations over time.

We have speculated on how such processing might be organized. In the schematic of figure 5, an auditory filter bank is followed by a layer of sum-and-difference nodes across local filter pairs. We can think of the difference units as primitive FM detectors, and the sums (integrated over time) as weighted amplitude sensors. Using the sum as gain control, the difference values can represent equivalent signed velocities of motion across the pairs. The velocities are then fed into a matrix of comparators. These are thresholding difference units, whose activation states (one-bit on/off output) are sensors of a brief moment of parallel motion.

The states are next preserved in a stack of *delay* planes (a bank of clocked shift registers?) whose current states are integrated over time in a separate *summing* plane. The sums represent the degree of cross-correlation, and the summing plane would control the routing of frequency-magnitude pairs to different perceptual spaces. We can speculate on the delay constants and the number of delay planes as follows: since auditory onsets are resolved by humans only to within 5 milliseconds,

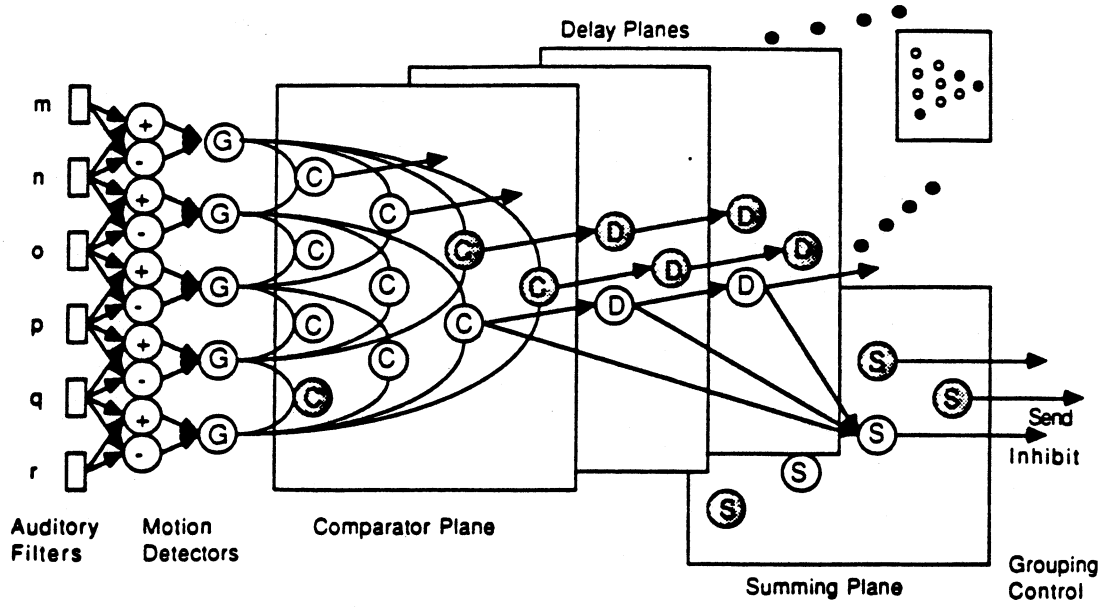

Figure 5. A network for grouping correlated motion. G nodes pass magnitude difference / average magnitude to a matrix of thresholding comparators, whose states are summed over time to determine grouping. Shaded on-states will send magnitudes of m,n,p,q,r to a single spectrum space; other transfers will be inhibited.

the planes are about 5 milliseconds apart; and since we can perceive no more than 15 sequential events per second (about 60 ms per), it apparently takes about 12 consecutive planes of correlated motion to register an integral event.

7. Implementation on the CM

The computation structure best suited to the above is an artificial neural net. Its primitive operations can be described as fan-out, weighting, fan-in, and an activation function, and these are organized into *layers* of processing. The implementation of such a layer on the CM is shown in figure 6. To propagate an output value with various weightings we use the *segmented copy-scan* operation, which replicates any value onto a string of contiguous processors. These processors hold the weights and target addresses defining a pattern of connectivity. All fan-out weights are now applied in parallel and the results sent to a structured destination—a set of contiguous receiving processors from which a high-speed *plus-scan* can sum all the inputs belonging to each single target. Each target then determines its state of activation as a nonlinear function of its input; connection weights are also modified.

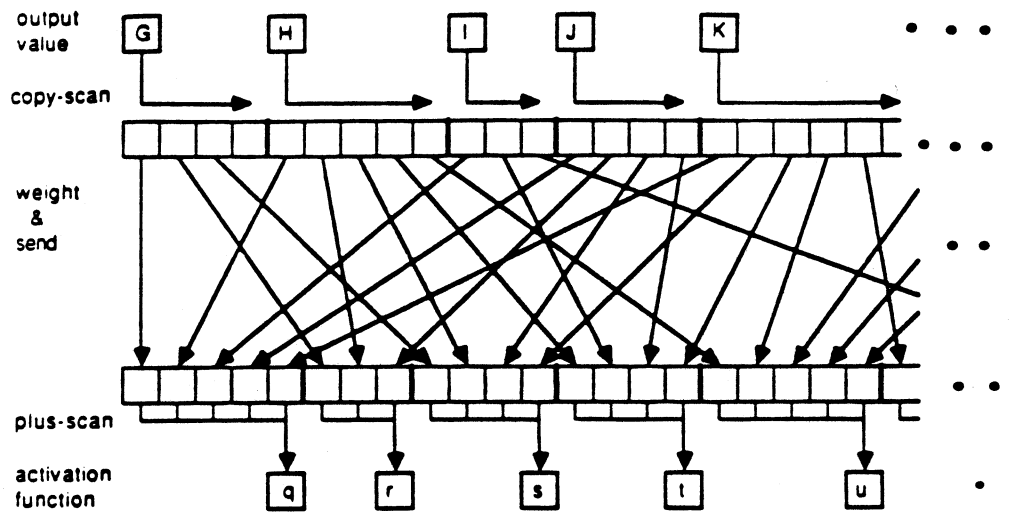

Figure 6. A layer of neuron-like communication (shown in part). Values G, H, .. are fanned out to adjoining processors; connection weights are applied and the results distributed to a staging area; fan-in then sums these for each target.

8. Conclusions and Prospects

This research has aimed to implement auditory parsing of music by porting classical signal processing (CSP) into the domain of massive parallelism, and smoothly linking it with artificial neural networks (ANN). The approach enables SIMD data-level parallelism to host a range of SP procedures and feature-detecting repertoires that are essential components of music perception and cognition. The polyphonic pitch detection problem is well served by massive parallelism: the peripheral auditory functions we already understand can be represented by CSP, and the less understood processes of perceptual grouping can be experimented on with ANN processing.

We sense that approaches which are cognizant of how biological systems solve audio problems are the best hope for machine understanding of music. For instance, we can look at how auditory parsing skills are developed during post-natal experience as a lesson on developing musically intelligent machines. We eventually hope to experiment with self-defing systems in which specific signal exposures would create a machine competence biased towards a specific skill, such as sonar, music, or speech separation, as a means of providing more robust tools for the recognition of semantic and expressive content.

9. References

Amuedo, J. "Periodicity Estimation by Hypothesis-Directed Search," *ICASSP 1985, Tampa, FL.*

Blelloch, G. and Rosenberg C. "Network Learning on the Connection Machine," Internal Document, Thinking Machines Corp., Cambridge, MA, Jan. 1987.

Childers D.G. and Lee, C.K. "Co-Channel Speech Separation," *ICASSP 1987,* 6.4.1-6.4.4.

Cumming, David. *Parallel Algorithms for Polyphonic Pitch Tracking* M.S. Thesis, Media Lab & E.E. Dept., MIT, 1988.

Danisewicz, R.G. and Quatieri, T.F. "An Approach to Co-Channel Talker Interference Suppression using a Sinusoidal Model for Speech," *Lincoln Laboratory Technical Report No. 794,* MIT, Cambridge MA, 1988.

Hanson, B.A. and Wong, D.Y. "The Harmonic Magnitude Suppression Technique for Intelligibility Enhancement in the Presence of Interfering Speech," *ICASSP 1984,* 18A.5.1-18A.5.4

Hillis, W.D., *The Connection Machine.* MIT Press, Cambridge MA, 1985.

Hubel, D.H., and Wiesel, T.N., "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *Journal of Physiology* (London), v. 160, 106-154, 1962.

McAdams, S.J. *Spectral Fusion, Spectral Parsing and the Formation of Auditory Images,* Ph.D dissertation, Stanford University, 1984.

Minsky, M. *The Society of Mind,* N.Y. Simon & Schuster, 1987.

Moorer, J.A. *On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer,* Ph.D dissertation, Stanford University, 1975.

Naylor J.A. and Boll, S.F. "Techniques for suppression of an Interfering Talker in Co-channel Speech," *ICASSP 1987,* 6.12.1-6.12.4.

Parsons, W. "Separation of speech from interfering speech by means of harmonic selection," *JASA 60 (4),* pp. 911-918, 1976.

Rumelhart, D. and McClelland, J. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition,* 2 vols, The MIT Press, 1986.

Schwede, Gary *Algorithms and Architectures for Constant-Q Fourier Spectrum Analysis,* PhD Thesis, U. of Cal., Berkeley, Nov 1983.

Sejnowski, T. and Rosenberg, C. "NETtalk: A Parallel Network that Learns to Read Aloud," *Johns Hopkins Univ. Tech. Report JHU/EECS-86/01,* 1986.

Vercoe, B. "The Synthetic Performer in the context of Live Performance," *ICMC Proceedings 1984,* pp. 199-200.

Vercoe, B. and Puckette, M. "Synthetic Rehearsal: Training the Synthetic Performer," *ICMC Proceedings 1985,* pp. 275-278.

Wawrzynek, I., Mead, C., Tzu-Mu, L., and Dyer, L., "A VLSI approach to sound synthesis," *ICMC Proceedings 1984,* pp. 53-64.

Whitfield, I. and Evans, E. "Responses of auditory neurons to stimuli of changing frequency," *J. Neurophysiol.* 28: 655-672, 1965.