Philippine Engineering Journal (1990) XI(2): 33-54

DIGITAL SIGNAL PROCESSING APPLICATIONS IN MONOPHONIC MUSIC ANALYSIS

Rowena Cristina L. Guevara, M.S.
Assistant Professor
Department of Electrical Engineering
University of the Philippines
Diliman, Quezon City

ABSTRACT

This paper presents an approach to monophonic music analysis using guided, continuous, overlapped fast Fourier transformation. This method is based on the physiological activity of the ear which can be modelled as a bank of contiguous bandpass filters. The analysis involves pitch detection, determination of note duration and identification of instrument limited to flute, clarinet and trumpet.

INTRODUCTION

Monophony in music pertains to the sound produced by a single instrumentalist playing one note at a time. The lowest level of music analysis deals with sound parameters: pitch, duration and timbre. Pitch is defined by the fundamental frequency of a tone. Duration is the length of time that a single tone is heard relative to other tones. Timbre is the tonal quality produced by an instrument which distinguishes it from other instruments.

"Digital signal processing is concerned with the representation of signals by sequences of numbers or symbols and the processing of these sequences." [6] In this case, the purpose of such processing will be to extract the sound parameters from monophonic musical signals.

The approach to music analysis discussed in this paper is the emulation of the ear's action, which is best described as follows:

> "Of clear significance in the auditory perception process are the signal processing operations performed by the cochlea. The cochlea has many outputs, with 30,000 neurons encoding 1500 to 2500 cochlear inner hair cell signals. Each neuron encodes a narrow band hair signal having a few hundred Hz of bandwidth, using a point process code, with the time between pulses coding the information being signalled into the neural network". [7].

From this description of the hearing mechanism, the signal processing that the musical signals should undergo could be thought of as a bank of band-pass filters followed by a fundamental frequency detector. The output of this bank gives directly the pitch of a tone and upon further analysis, estimates the duration of the tone and identifies the instrument producing the tone.

THEORETICAL BACKGROUND

The basic mathematical tool in signal processing is the Fourier Transform. It makes possible the analysis of a signal in either the time domain or the frequency domain. Some properties of a signal are not apparent in the time domain waveform; specially if it is distorted. In the case of music signals, inherent tremolo (amplitude modulation, see Fig. 1) and vibrato (frequency modulation) could distort the signal so much that the shape and period of the waveform will vary from one period to another. This is exemplified in Fig. 2, which shows the evolution of a C4 tone. In the first line, there are 2 local maxima and 2 local minima in each period. In the second line, there is 1 local maxima and 1 local minima. In the last line, there are 5 zero crossings in one period, compared to the single zero crossing per period in the previous portions.

In the frequency domain, the spectral line corresponding to the fundamental frequency or pitch of the signal will stand out with tremolo seen in the sidelobe, 0 - 6 Hz away from the fundamental and vibrato seen as the slight shifting of the fundamental spectral line.

The Fourier Transform is mathematically expressed as

$$S(f) = \int_{-\infty}^{\infty} s(t)\ e^{-j2\pi ft}\, dt \qquad (1)$$

where s(t) is the time domain waveform and S(f) is the Fourier Transform of s(t). The Fourier Transform decomposes s(t) into a sum of sinusoids. If the signal is periodic, S(f) will be discrete with nonzero components at frequencies which are multiples of the signal's frequency. If s(t) is nonperiodic, S(f) will be a continuous function of frequency.

For signals which cannot be represented analytically, the Discrete Fourier Transform (DFT) is used instead of the continuous Fourier Transform. The mathematical representation for the DFT is

$$S(n/NT) = \sum_{k=0}^{N-1} s(kT)\ e^{-j2pnk/N} \qquad (2)$$

$$n = 0,1,2,....,N-1$$

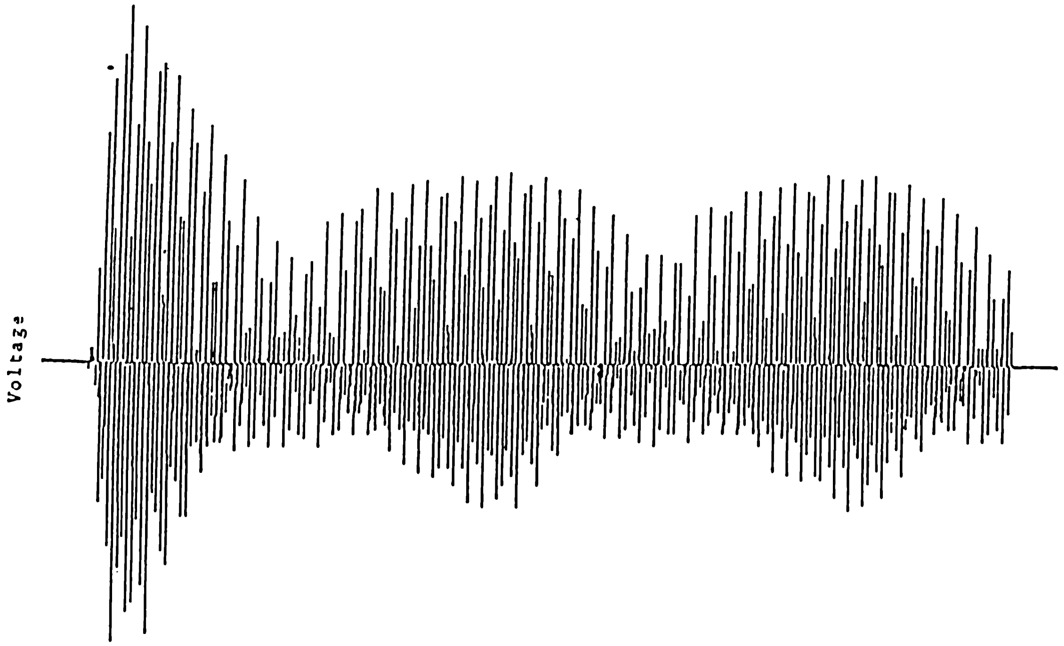


Fig. 1 - Decimation in time of a 1.23 sec C4 Flute tone, down-sampled to a frequency of 417 Hz, exemplifying amplitude modulation

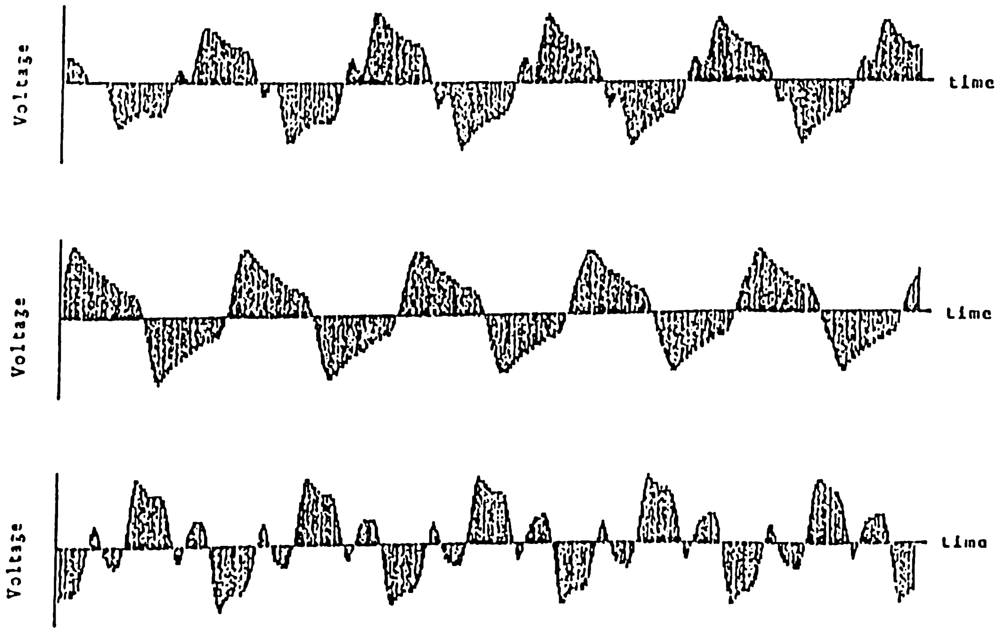


Fig. 2 - Steady-state portions of a synthesized Clarinet tone (C4)

where s(kT) is the sampled value of s(t) at t = kT, T being the sampling period. S(n/NT) is the DFT of s(kT). The DFT may be derived as a special case of the continuous Fourier Transform. A graphical development of this is found in BRIGHAM [2].

The effect of getting the N-point DFT of a signal is equivalent to passing the signal through a bank of N contiguous band pass filters (BPF), where the bandwidth of each filter is 1/NT (T is the sampling period) and the center frequency of the ith BPF is i/NT. The result of the DFT is scaled discrete version of the continuous Fourier Transform where the scaling factor is To/T (To is the length of the rectangular window and T is the sampling period). In connection with the previous discussion, it is obvious that the DFT could emulate the hearing action. If the DFT equation is implemented for a sequence of N samples, N2 multiplications must be performed. The number of multiplications required for the DFT increases exponentially with N. Since computer processing time is proportional to the number of multiplications to be performed, decreasing the number of multiplications would shorten the processing time. The Fast Fourier Transform (FET) algorithm is an implementation of the DFT that aims to do this.

After performing an FFT, pitch may be determined by looking at the resulting spectra. Fig. 3 shows the evolving spectrum of a trumpet tone C4. Visually, one can establish the fundamental and the harmonic lines to be the 11th, 21st, 31st, etc. lines. In between these lines are inharmonic smears which are inherent since 1) the analog-to-digital conversion introduced quantization errors; 2) the 1024-point FFT is being performed on a sequence that is not an integral multiple of the period; and 3) the 1024-point rectangular window which is multiplied to each 1024-point sequence is equivalent to a convolution in the frequency domain of a sine function and the harmonic spectral lines.

The first item is inherent to any system that includes analog-to-digital conversion. Increasing the number of bits will increase the bit-resolution, thereby decreasing the quantization error; but it is essentially economy that determines the choice. As the number of bits is increased, the price of the IC also increases.

The second item's contribution is uncrontrollable since the speed of the FFT algorithm rests on the fact that the number of samples being transformed is a power of 2. It is possible to lessen the smears by widening the rectangular window, but this would lengthen the data sequence and might result in an analysis that is too slow for the rate of note-changing.

The third item maybe corrected by applying a smooth-edged window, like the Blackman or Hamming window to the data sequence before performing the FFT. For these two windows, the main lobe is much wider, at the same time the energy in the side lobes is much lower when compared to that of the rectangular window. This is a consequence of the smooth edges of the Blackman and Hamming window. The effect of pre-multiplying the samples with either Blackman or Hamming window is the lowering of the energy in the smears; this is due to the lower energy in the side lobes of the Blackman and Hamming window.

Informal experiments using the Blackman and Hamming windows on the same trumpet tone produces very little improvement in the spectrum, since the smears or leakages

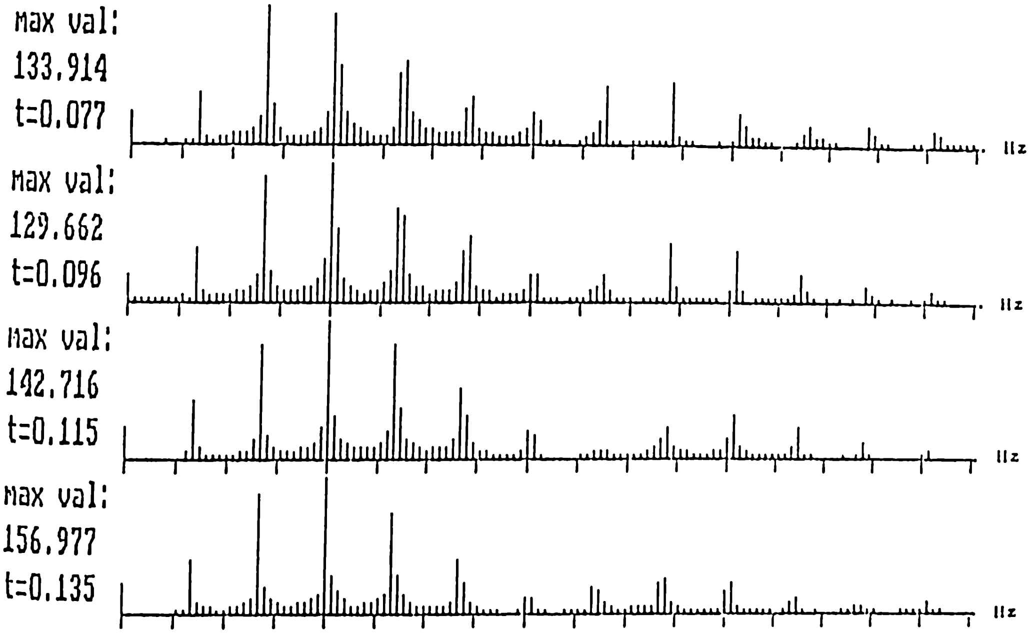


Fig. 3 - Periodogram of a C4 synthesized Trumpet tone, horizontal scale:]96 Hz per division

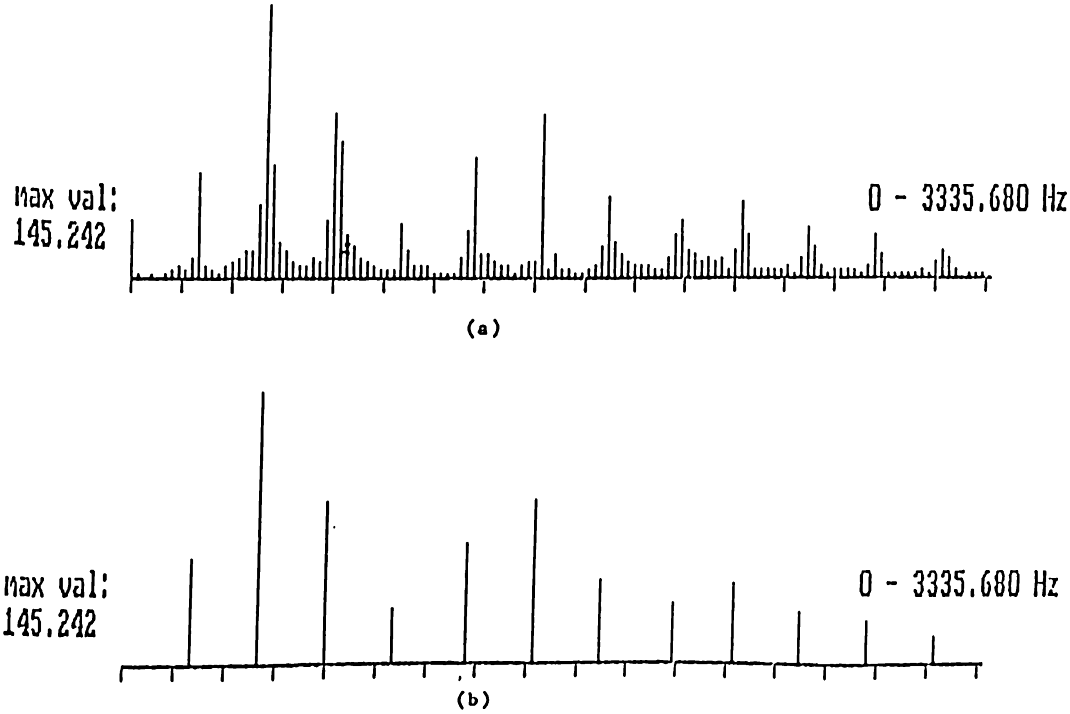


Fig. 4 - (a) Raw and (b) Flattened Spectrum of a C4 synthesized Trumpet tone

in the main lobe were amplified. Having exhausted proven means of sharpening the spectrum, a different approach was developed. One approach is to look for the line with the highest amplitude and designate it as the fundamental. The second, third and fourth line of Fig. 3 indicate that this approach fails since the line with maximum amplitude line could be the fundamental or any of the harmonics.

This is where the idea of spectrum flattening comes in. Spectrum flattening issued to emphasize the harmonic amplitudes by removing the smears or leakages and based on the distances between the harmonics, establish the fundamental frequency or pitch in speech analysis [21]. The routine developed for spectrum flattening consists of three stages:

1) threshold cancelling - where all spectral lines below a threshold (10% of the maximum amplitude) are zeroed.

2) the local maxima are located and these are designated as the harmonic spectral lines.

3) double checking is done to remove stray lines (these are lines of significant amplitude) falling near the harmonic spectral lines.

Figures 4A. and B. show the raw spectrum and the flattened spectrum of a tone, respectively. In the flattened spectrum the fundamental and the harmonic spectral lines are distinct, making fundamental frequency detection possible.

HARDWARE AND SOFTWARE

The data acquisition hardware consists of a buffer, an anti-aliasing filter and an analog-to-digital converter (ADC). The buffer is a unity gain voltage follower and the anti-aliasing filter is an 8th order Butterworth lowpass filter with a cutoff frequency of 8 kHz and an overall gain of 6.83. The 8 kHz cutoff frequency will allow up to the 4th harmonic frequency of the highest note pitch of interest, 2093 Hz. The sampling rate of the ADC is 26686 Hz. This surpasses the Nyquist requirement of 2 * 8 kHz = 16 kHz for sampling. Thus the digitized data is reliably undistorted. Control of the ADC and data recording is done through an assembly language program.

The sound sources that were used were a Yamaha Portasound Keyboard for the synthesized sounds and recorded live instruments for the natural sounds.

The FFT implementation used was developed by Bergland and Dolan [1] and is based on the original Cooley-Tukey FFT algorithm, memory-optimized, by taking advantage of the symmetry of the transform of real signals.

For N = 1024 and a sampling rate of Fs = 26686 Hz, the frequency resolution (frequency difference between two spectral lines) is 26. 06 Hz ($f_r = f_s/N$).

PITCH DETECTION

From a musical point of view, pitch is that attribute of auditory sensation in terms of which sounds may be ordered on a scale extending from low to high, such as a musical scale. Pitch depends on the fundamental frequency of the sound stimulus. The human ear is capable of hearing sounds with frequencies between 20 Hz and 20,000 Hz with the appropriate loudness [5]. For low frequencies to be heard, the peak-to-peak amplitude of the sound must be large enough to be detected by the ear. For high frequencies, the peak-to-peak amplitude of the sound must be small enough to prevent injury to the ear drum. At low frequencies, the ear can distinguish two consecutive tones or pitches with fundamental components which are 1- 2 Hz apart. At higher frequencies, the distinction between two consecutive tones is possible only if the frequency difference in the fundamental components is about 20 - 50 Hz [3].

In this paper, the musical scale used to define pitch is the equal-tempered musical scale of Western music. Table 1 shows the different pitches possible in this scale and the corresponding frequencies of the fundamental up to the fifth harmonic. The table shows the portion of the scale which contains the pitches relevant to this paper, D3 to C7. The whole scale spans from C0 to B9 with C4 termed as the middle C.

Pitch detection in music analysis involves the determination of the fundamental frequency of a tone and its corresponding pitch on the musical scale.

Given the time domain waveform of a signal emanating from a musical instrument (see Fig. 2), the fundamental frequency may be determined by taking the reciprocal of the period. One way of doing this is by taking the time duration between zero crossings. This method which works very well for strictly periodic waveforms, fails when applied to musical signals. Fig. 5 shows a plot of number of samples between zero crossings versus time for a single tone. The number of samples between zero crossings is not constant as expected. This phenomenon may be attributed to the natural vibrato of musical instruments. It is defined as the slight shifting of the fundamentál spectral line. Another possible culprit is waveform evolution (see Fig. 2).

Given the frequency domain representation of a musical signal, see Fig. 4 B., the fundamental is easily seen to be the lowest spectral line of significant amplitude. If the frequency resolution is known, then the horizontal axis of the Fourier Transform may be scaled accordingly, with the 0 Hz position corresponding to the dc component. Thus the frequency of the fundamental is easily determined.

The author's initial algorithm for pitch detection involved only the detection of the fundamental frequency of the tone. The lowest two notes in the frequency range of interest are D3 and D#3 of the clarinet which have fundamental frequencies of 146.83 Hz and 155.56 Hz respectively. The difference between these two frequencies is 8.73 Hz. This implies that the frequency resolution of the FFT to be used must be 8.73 Hz in order to distinguish between the two lowest pitches. With a sampling rate of 26686 Hz, this required at DFT of

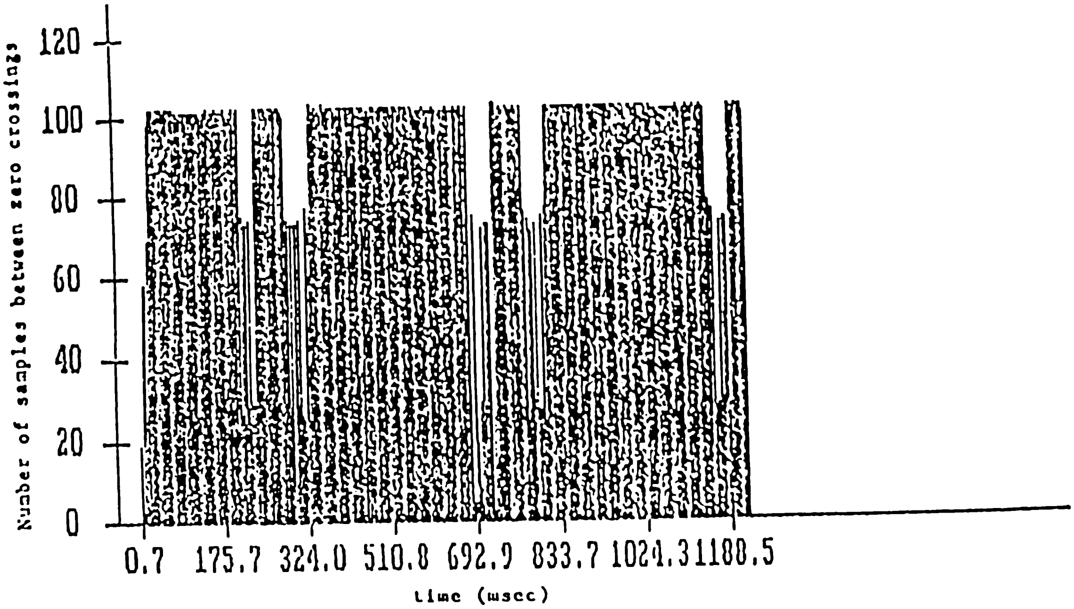


Fig. 5 - Number of samples between zero crossings versus time for a synthesized Flute tone (C4)

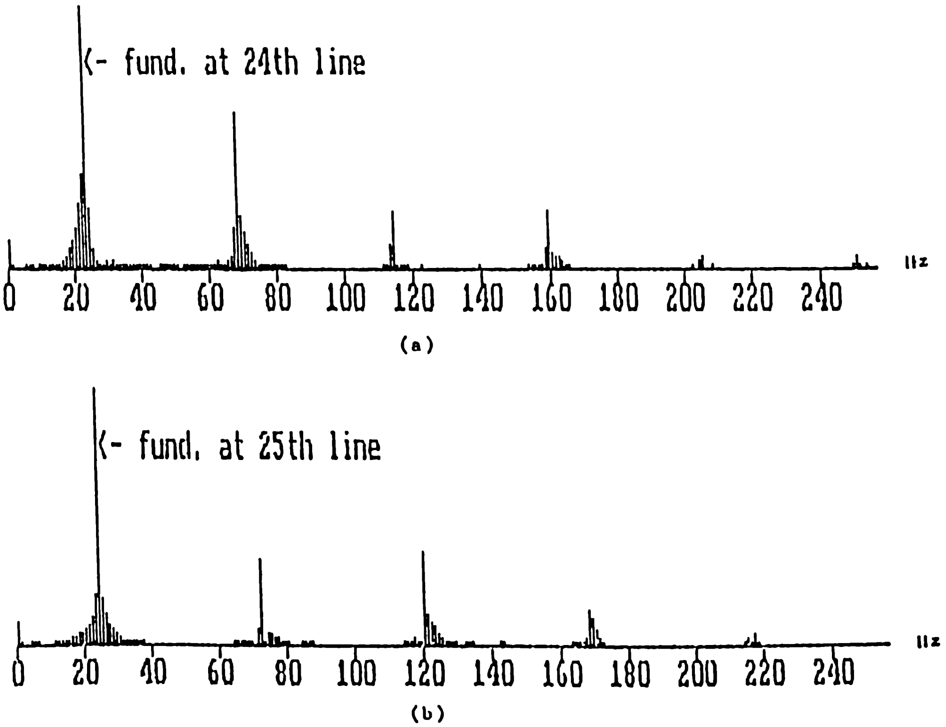


Fig. 6 - Spectrum of (a) D3' (b) D#3 synthesized Clarinet tone

$N = f_s/f = 26686/8.73 = 3056$ samples (3)

The nearest power of 2 from 3056 is 4096. (N must be a power of 2 when using the Fast Fourier Transform routine).

Figures 6 A. and B. show the results of 4096-point FFT performed on the D3 and D#3 tones of the clarinet. The fundamental of the note D3 falls on the 24th line while that of the note D#3 falls on the 25th line. Since the resolution of the 4096-point FFT works for the lowest pitches, it will work for the higher pitches where the fundamental frequencies are even farther apart.

On, a PC-XT, a 4096-point FFT will take about 50 sec. Aside from the long computational time, the analysis window was too long:

$t_w = 1/26686 \text{ X } 4096 = 0.1535$ sec. (4)

The FFT of a sound sample will result in a sharp spectrum if the tone was sampled during its steady-state portion. Otherwise the energy in between harmonic frequencies will be high. This means that the fastest note that a player may execute such that the pitch is perfectly detected should have a steady-state portion that lasts 2 X 0.1535 sec = 0.307 sec.

The multiplication by 2 is explained this way. If continuous short-time 4096-point FFT will be performed on the music to be analyzed, the 4096-point rectangular window's beginning and end will be randomized with respect to note beginnings and ends. It becomes random because the player's tempo is not synchronized with any metronome. Therefore for a note to be detected, it should last 2 X 0.154 sec, so that if the beginning of the 4096-point window falls anywhere in the 8192-point duration of the note, the pitch will still be detected. This allowable note speed, is equivalent to the instrumentalist playing a quarter note at 391 M.M. This is very slow, with the player limited to playing Andante, or moderate speed.

An improvement in the allowable note speed will result if instead of taking the short-time FFT continuously, an overlapping by 2048 samples of the 4096-point windows is done. This simple solution will double the note speed allowable to an eighth note at 391 M.M. But this is only Moderato tempo. This improvement would be at the cost of performing twice the number of FFT operations in the original method, so the processing time is doubled. This overlapping technique is equivalent to a person reviewing the recording a second, third or fourth time to catch a fast note.

The above scheme works well enough within its limitation, however it does not fully use the whole span of the spectrum. In an aural experiment made by Luce and Clark [15], they removed the fundamental of a tone and asked musically literate subjects to identify the pitch of the original and the modified tone. The subjects identified the two notes as having the same pitch. This implied that even without the fundamental, the pitch may be determined by just listening to the harmonics. An informal explanation to this could be that the ear, after hearing the harmonics, probably determines the least common multiple of the frequency of the harmonics and decides that this must be the frequency of the fundamental.

Among the three instruments, the flute has the least number of spectral components. In fact, beyond the third harmonic component, the magnitude of the spectral lines are negligible. To use the third harmonic for pitch distinction, the frequency resolution of the FFT must be 26.2 Hz. This is the frequency difference between the third harmonic components of the two lowest notes, D3 and D#3 (see Table 1). This frequency resolution will require a DFT of

$$N = f_s / f_r = 26686/26.2 = 1019 \text{ samples} \qquad (5)$$

or a 1024-point FFT. If the 1024-point FFT is overlapped by 512 samples, the shortest note for pitch analysis should last 1024/26686 = 38.3 msec (musically, this is a 16th note at 𝅗𝅥 = 390 M.M.).

Table 2 lists the spectral line positions of the fundamental up to the 8th harmonic of the different pitches from D3 to A4 for a 1024-point FFT. Pitch detection boils down to determining the position of the third harmonic spectral line and looking up the corresponding pitch from Table 2.

A decision tree fully based on Table 2 needs fine tuning because the frequency of the fundamental and the harmonics seldom fall exactly on the spectral line's discrete frequency. This is due to the discrete frequencies of the DFT. Fine tuning was done by determining the actual spectral line positions of all the notes (see Table 3).

When a 'rest' (absence of tone) is FFT'd, the resulting spectra will be that of equipment noise and will be of very low amplitude.

INSTRUMENT IDENTIFICATION

Timbre is the musical parameter which allows a listener to distinguish one instrument from another. Visibly, this distinction may be established from the time domain waveform.

Sounds produced by different instruments will have different waveforms. An approach to instrument identification in the time domain would be to make templates of the shape of the waveform produced by each instrument. However, as shown in Fig. 2, the sound waveform varies from one period to another, therefore several templates will be needed to identify one instrument.

Instrument identification is best performed in the frequency domain because the spectral content of a tone plays the most important role in determining its timbre. The presence of high frequency components would make a tone sound bright and the lack of it would make the tone mellow. The presence of strong inharmonic frequencies would make a sound unpitched. The absence of even harmonics would make a tone sound reed-like. The presence of very few harmonic components would make a tone sound very simple. The presence of all frequency components (white noise) would sound like a rush of air. White noise passed through a comb filter will sound like a jet passing by [3].

Table 1: Frequencies (in Hz) of the Fundamental to the Fourth Harmonic in the Scale of Equal Temperament

note	1st	2nd	3rd	4th
D3	146.8	293.7	440.5	587.3
D#3	155.6	311.1	466.7	622.3
E3	164.8	329.6	494.4	659.3
F3	174.6	349.2	523.8	698.5
F#3	185.0	370.0	555.0	740.0
G3	196.0	392.0	588.0	784.0
G#3	207.7	415.3	623.0	830.6
A3	220.0	440.0	660.0	880.0
A#3	233.1	466.2	699.2	932.3
B3	246.9	493.9	740.8	987.8
C4	261.6	523.3	784.9	1046.5
C#4	277.2	554.4	831.5	1108.7
D4	293.7	587.3	881.0	1174.7
D#4	311.1	622.3	933.4	1244.5
E4	329.6	659.3	988.9	1318.5
F4	349.2	698.5	1047.7	1396.9
F#4	370.0	740.0	1110.0	1480.0
G4	392.0	784.0	1176.0	1568.0
G#4	415.3	830.6	1245.9	1661.2
A4	440.0	880.0	1320.0	1760.0
A#4	466.2	932.3	1398.5	1864.7
B4	493.9	987.8	1481.7	1975.5
C5	523.3	1046.5	1569.8	2093.0

(cont'd)

Table 1

note	1st	2nd	3rd	4th
C#5	554.4	1108.7	1663.1	2217.5
D5	587.3	1174.7	1762.0	2349.3
D#5	622.3	1244.5	1866.8	2489.0
E5	659.3	1318.5	1977.8	2637.0
F5	698.5	1396.9	2095.4	2793.8
F#5	740.0	1480.0	2220.0	2960.0
G5	784.0	1568.0	2352.0	3136.0
G#5	830.6	1661.2	2491.8	3322.4
A5	880.0	1760.0	2640.0	3520.0
A#5	932.3	1864.7	2797.0	3729.3
B5	987.8	1975.5	2963.3	3951.1
C6	1046.5	2093.0	3139.5	4186.0
C#6	1108.7	2217.5	3326.2	4434.9
D6	1174.7	2349.3	3524.0	4698.6
D#6	1244.5	2489.0	3733.5	4978.0
E6	1318.5	2637.0	3955.5	5274.0
F6	1396.9	2793.8	4190.7	5587.7
F#6	1480.0	2960.0	4439.9	5919.9
G6	1568.0	3136.0	4703.9	6271.9
G#6	1661.2	3322.4	4983.7	6644.9
A6	1760.0	3520.0	5280.0	7040.0
A#6	1864.7	3729.3	5594.0	7458.6
B6	1975.5	3951.1	5926.6	7902.1
C7	2093.0	4186.0	6279.0	8372.0

Table 2 - Spectral Line Positions for N = 1024

note	1st	2nd	3rd	4th	5th	6th	7th	8th
D3	7	12	18	24	29	35	40	46
D#3	7	13	19	25	31	37	43	49
E3	7	14	20	26	33	39	45	52
F3	8	14	21	28	35	41	48	55
F#3	8	15	22	29	36	44	51	58
G3	9	16	24	31	39	46	54	61
G#3	9	17	25	33	41	49	57	65
A3	9	18	26	35	43	52	60	69
A#3	10	19	28	37	46	55	64	73
B3	10	20	29	39	48	58	67	77
C4	11	21	31	41	51	61	71	81
C#4	12	22	33	44	54	65	75	86
D4	12	24	35	46	57	69	80	91
D#4	13	25	37	49	61	73	85	97
E4	14	26	39	52	64	77	90	102
F4	14	28	41	55	68	81	95	108
F#4	15	29	44	58	72	86	100	115
G4	16	31	46	61	76	91	106	121
G#4	17	33	49	65	81	97	113	128
A4	18	35	52	69	85	102	119	136

Table 3 - Actual Spectral Line Positions for N = 1024 for notes which are identifiable from the position of the fundamental

NOTE	FUNDAMENTAL	2nd HARMONIC	3rd HARMONIC
C6	41	82	122, 123
B5	39	77, 78	115, 116
A#5	37	73	108, 109
A5	35	69	102, 103
G#5	33	65	96, 97
G5	31	61, 62	91, 92
F#5	30	58	87
F5	28	55	82, 83
E5	26, 27	52	77, 78
D#5	25	49, 50	73, 74
D5	24	46, 47	69, 70
C#5	22, 23	44	65, 66
C5	21	41	61, 62
B4	20	39	58, 59
A#4	19	37	54, 55, 56
A4	18	35	52
G#4	17	33	49
G4	16	31	46, 47

(Cont'd)

Table 3 - Actual Spectral Line Positions for N = 1024 for notes which are identifiable from the position of the 2nd and 3rd harmonic

NOTE	FUND.	2nd HARM.	3rd HARM.	4th HARM.	5th HARM.
F#4	15	29, 30	44	58	72, 73
F4	14, 15	28	41, 42	55	68, 69
E4	14	26, 27	39, 40	52	64, 65
D#4	13	25	37	49	61
D4	12	24	35	47	57, 58
C#4	12	22, 23	33	44	54, 55
C4	11	21	31	41	51, 52
B3	11	20	29, 30	39	48, 49
A#3	10	19	28	37	46
A3	9, 10	18	26, 27	35	43, 44
G#3	9	17	25	33	41
G3	9	16	24	31	39
F#3	8	15	22, 23	30	36, 37
F3	8	14	21	28	35
E3	7	14	20	27	33
D#3	7		19		31
D3	7		18		29

Analysis of wind instrument tones by STRONG and CLARK [23] indicated the interplay of the spectral envelope (from the plot of the relative amplitudes of the harmonics versus frequency) and the temporal envelope (from the plot of the amplitude envelope versus time, see Fig. 1) in characterizing the timbre of an instrument. In order to identify the instrument producing a tone, a comprehensive analysis of all orchestral instruments is called for. Even STRONG and CLARK [22] indicated that methods other than the auditory process would entail complicated schemes. Narrowing the choice to three instruments makes the problem of instrument identification manageable. A look at the spectrum of the three instruments after spectrum flattening makes the job easier. Samples of flattened spectrum for the note C4 for each instrument are shown in Fig. 7.

After examining the spectra of all the pitches that may be produced by each instrument, the author concluded the following. The clarinet tone is identified with the absence of even harmonics. The flute tone is identified with the absence of high frequency harmonics. The trumpet is identified when the number of harmonic spectral line present is greater than 4.

DETERMINATION OF NOTE DURATION

The output of the continuous short-time FFT used in pitch detection is the pitch of a 1024-point sample lasting 38 msec. The exact beginning of the note producing this pitch cannot be ascertained since the beginning of the 1024-point sample is random with respect to the occurence of notes. Since the problem of determining duration is basically time measurement, the first approach should be in the time domain.

The sound produced by musical instruments may be divided into three parts: attack, steady-state and decay. In between the steady-state of two notes is the decay portion of the previous note and the attack portion of the succeeding note. This is manifested in the plot of the average magnitude or the temporal envelope as valleys. The extraction of note duration thereby reduces to determining the time between these valleys.

The formula for the average magnitude is

$$M_n = \sum_{m=-s/2}^{s/2} |x[m]| w[n-m] \qquad (6)$$

The averaging window used, w[n-m] is a rectangular window of unity value; s is the window size and x[m] is the sequence to be averaged.

Since the rate of note changing is much less than the sampling rate, decimation in time may be done without losing information. This is implemented by taking every N^{th} sample and discarding the N-1 samples in between, for an Nth decimation. As for the window size, the longer it is, the flatter the average becomes. When the window used is too

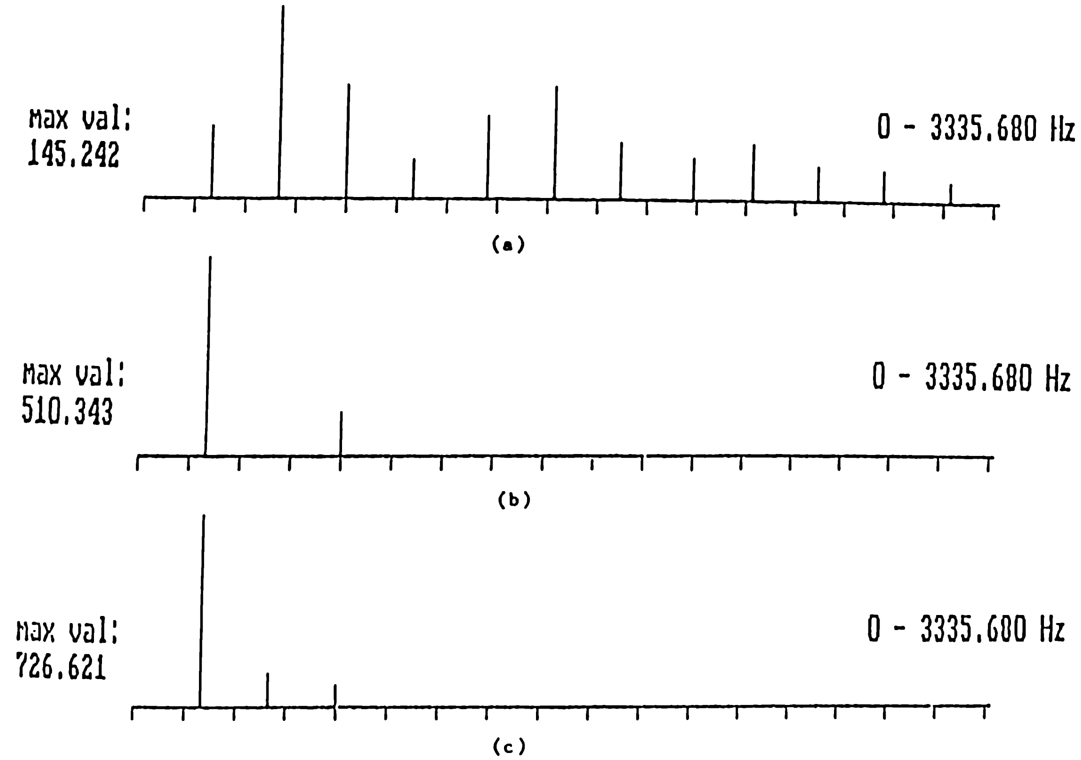


Fig. 7 - Flattened Spectrum of a C4 (a) Trumpet (b) Clarinet (c) Flute Tone

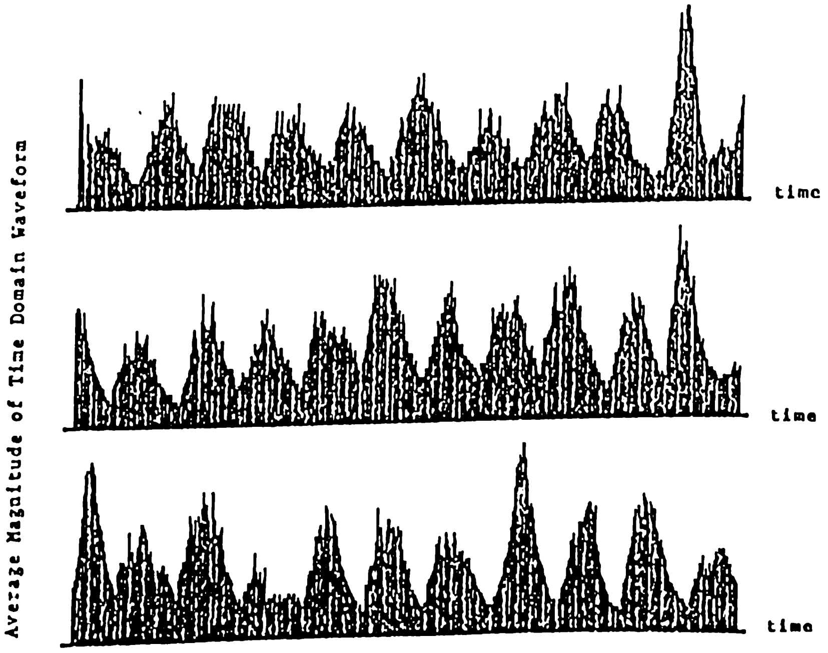


Fig. 8 - Temporal Envelope of 32 successive Trumpet tones, played staccatto from E3 to B5, f_s=256 Hz, window size: 13 samples, duration: 1.6 seconds

long, the valleys between notes disappear, making it impossible to determine the start and end of a note.

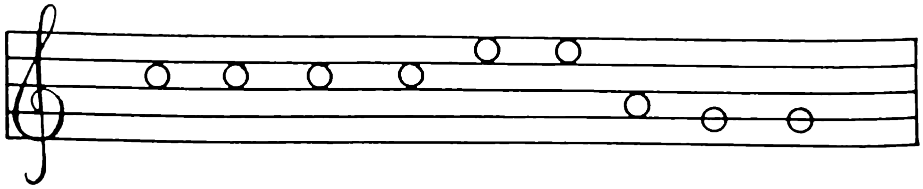
The effect of decimating and getting the average magnitude is equivalent to the operation of an envelope detector implemented using a peak detector followed by a lowpass filter (LPF). The LPF is the equivalent of decimation since the frequency of note changing is in the band pass and the higher frequencies containing pitch information is band stopped. Getting the average magnitude is equivalent to peak detection. Figures 8 and 9 show the result of applying this method to a chromatic passage consisting of 32 equal duration. In Fig. 8, the music was played by the trumpet, staccatto[1] style. The 32 notes played are easily seen to be of equal duration since the valleys and peaks are equally spaced and very prominent. In Fig. 9, the music was played by the clarinet, legato[2] style. Notice that the peaks and valleys are no longer equally spaced and the expected 32 peaks and valleys are not prominent anymore.

From the preceding discussion it is obvious that the playing style of the instrumentalist affects the effectivity of using decimation and magnitude averaging in determining the duration of a note.

Another factor that will affect this method is the amplitude modulation (tremolo) that is an inherent characteristic of wind instruments. Figures 10 to 12 show the temporal envelope for low, medium and high pitches for each instrument. Each plot represents a long note lasting 1.6 seconds. This is done to emphasize the amplitude modulation that occurs.

For low clarinet tones, the valleys are not deep enough to be mistaken as the start of a new note, but for the rest of the samples, specially for high notes, the valleys are deep enough to be mistaken as the start of a new note.

In a musical score, the duration of a note or a rest (the absence of sound) is relative rather than absolute. If the score is presented in the manner shown below,



[1] Staccatto style of playing a wind instrument is characterized by the staggered blowing action of the instrumentalist, separating the notes from one another and accentuating each one.

[2] Legato style of playing a wind instrument involves the single blowing action for a string of notes, merging one note to another.

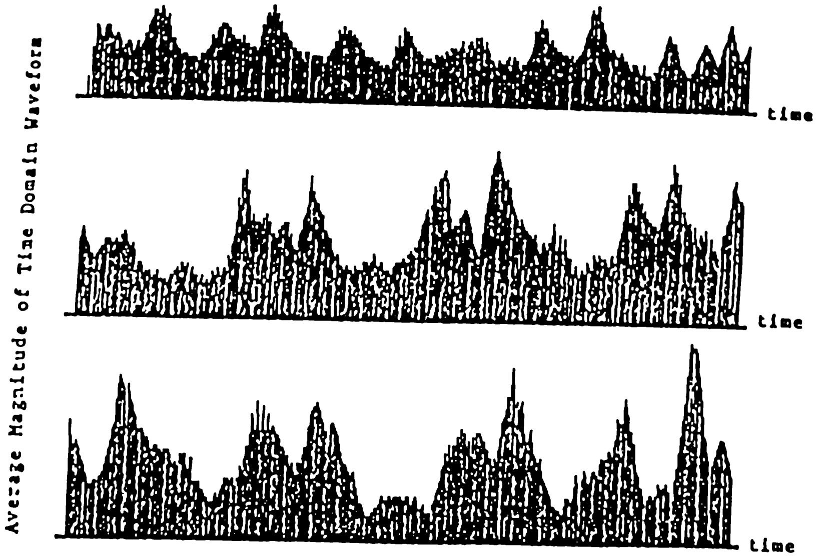


Fig. 9 - Temporal Envelope of 32 successive Clarinet tones, played legato from D3 to A5, f_S=256 Hz, window size: 13 samples, duration: 1.6 seconds

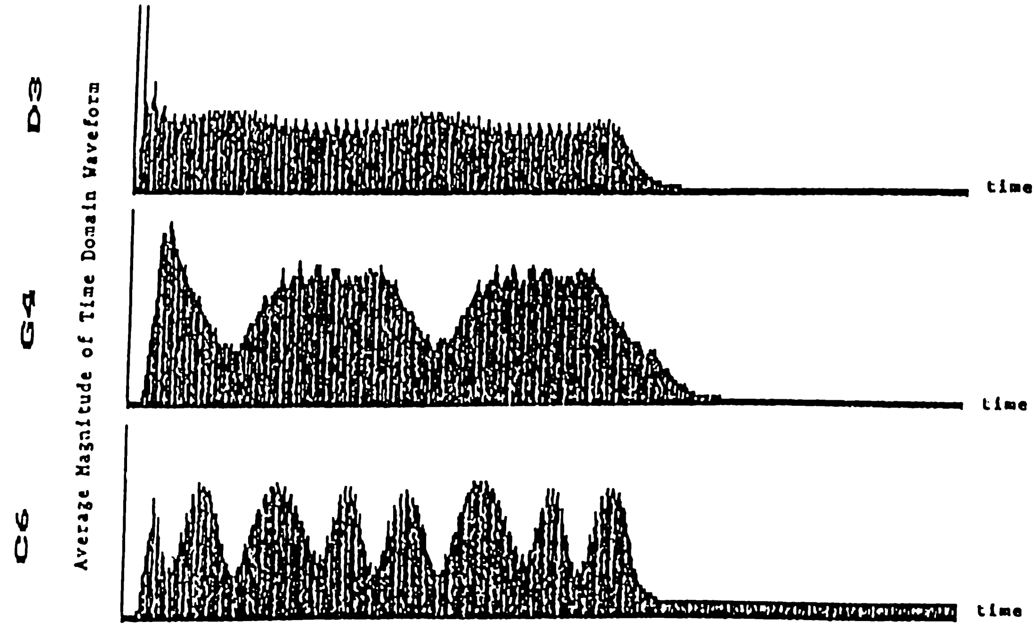


Fig. 10 - Temporal Envelope for 3 long Clarinet tones: D3, G4 and C6, sampled at 256 Hz, window size: 13 samples, duration: 1.6 seconds

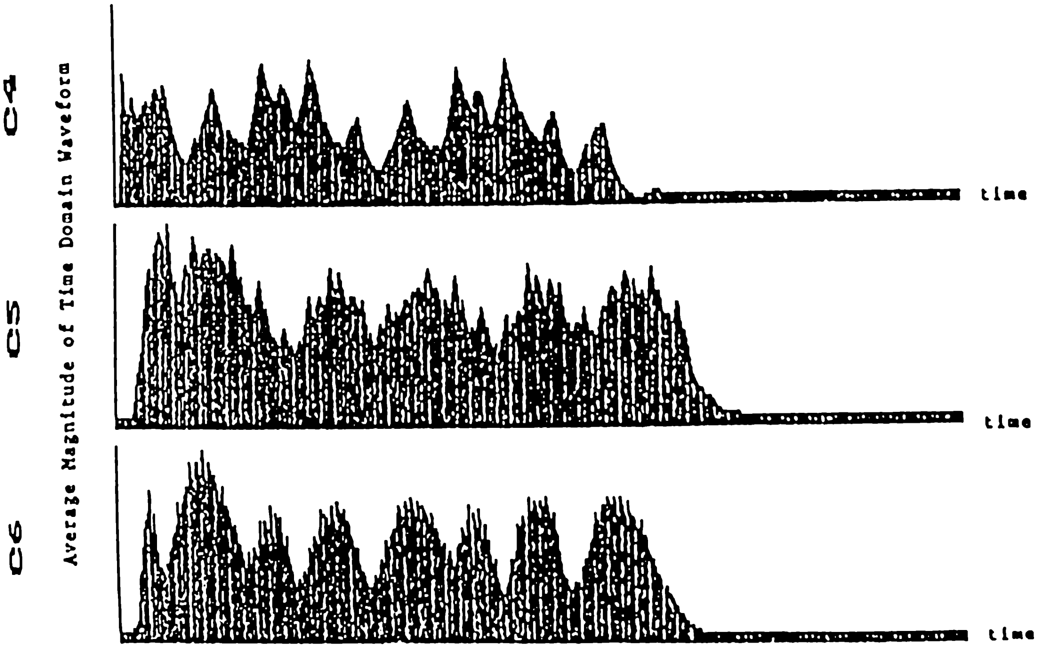


Fig. 11 - Temporal Envelope for 3 long Flute tones: C4, C5 and C6 sampled at 256 Hz, window size: 13 samples, duration: 1.6 seconds

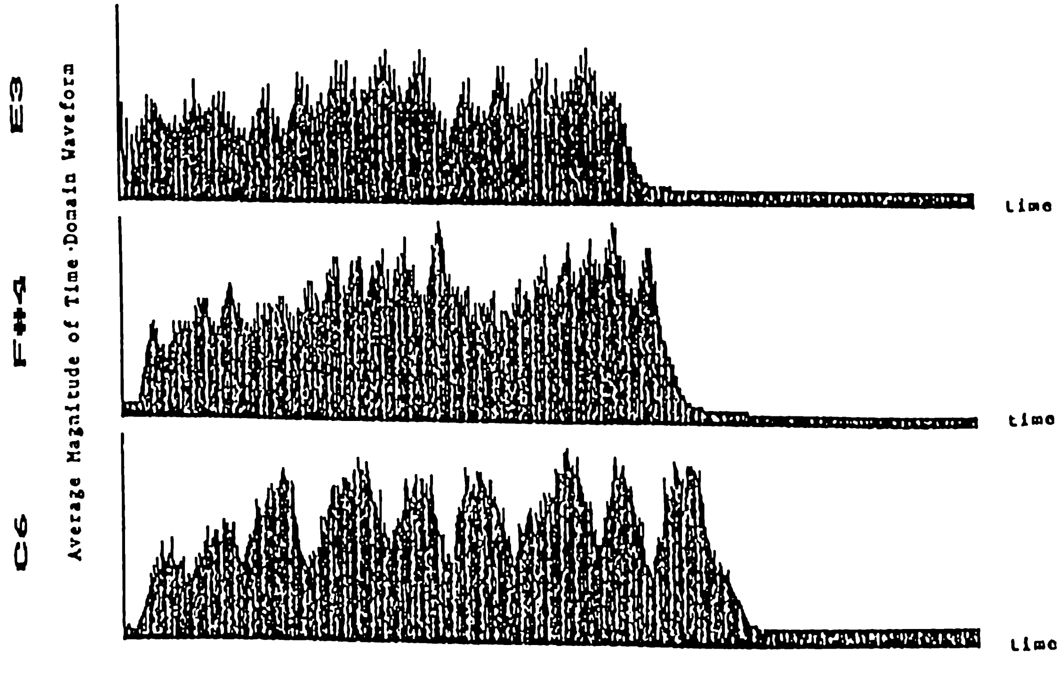
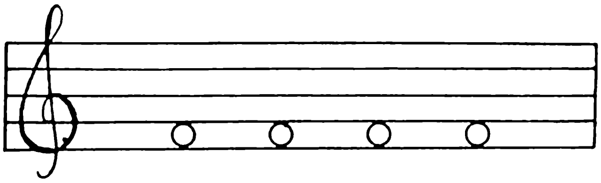


Fig. 12 - Temporal Envelope for 3 long Trumpet tones: E3, F#4 and C6, sampled at 256 Hz, window size: 13 samples, duration: 1.6 seconds

the musician could infer the relative durations of the notes. If the duration of the third note is used as a basis, the ratio of note duration would be 4:2:1:2. When the result of the overlapped 1024-point FFT is transcribed in this manner the duration of the notes can be estimated. A musical score presented this way completely characterizes the musical passage. The problem with this representation is that one cannot differentiate a note of long duration from a series of short notes having the same pitch, i.e., the following score,



could mean an F4 note played 4 units of time or 4 F4 notes played 1 unit of time each, or 2 F4 notes played 2 units of time each, or 2 F4 notes, one played 1 unit of time, the other played 3 units of time.

o remove this ambiguity, information about the maximum spectral magnitude (MSM) of each 1024-sample frame is needed. Since each tone is composed of an attack, steady state and decay portion, then the successive MSMs of a tone will be composed of a high amplitude MSM (corresponding to the steady state) in the middle of 2 low amplitude MSMs (corresponding to the attack and decay).

CONCLUSION

In the preceding discussion, DSP was used basically for extracting certain features in musical signals to establish pitch, instrument identity and note duration. The algorithm discussed can fully extract the necessary features needed to decompose (the opposite or composing) music. It emulates the hearing process. And just as the brain guides the hearing process, DSP must be guided by an "expert system" which consists of several decision levels, each of which is fed with the output of a specific DSP routine.

REFERENCES

1. BERGLAND, G.D. and DOLAN, M.T. (1979), Programs for Digital Signal Processing - Fast Fourier Transform Algorithm, IEEE Press, New York.

2. BRIGHAM, E.O. (1988), The Fast Fourier Transform and its Applications, Prentice Hall Inc., Englewood Cliffs, New Jersey.

3. CHAMBERLIN, H. (1980), Musical Applications of Micropro-cessors, Hayden Book Co. Inc., Rochelle Park, N.J.

4. KUC, R. (1988), Introduction to Digital Signal Processing, McGraw-Hill Co., Inc., Singapore.

5. OLSON, H. (1952), Musical Engineering, McGraw-Hill Co., Inc., York, Pa.

6. OPPENHEIM, A. V. and SCHAFER, R. W. (1975), Digital Signal Processing, Prentice-Hall Inc., Englewood Cliffs, New Jersey.

7. ALLEN, J. B. (1985), "Cochlear Modelling," IEEE ASSP Magazine, pp. 3-29.

8. BLESSER, B. (1981), "Perceptual Issues in Digital Processing of Music," IEEE, pp. 583-586.

9. CAHN, F. (1983), "Pitch Translation of Trumpet Tones," IEEE, pp. 1380-1383.

10.. FOSTER, S. and ROCKMORE, A. J. (1982), "Signal Processing for the Analysis of Musical Sound," IEEE, pp. 89-92.

11. FREEDMAN, M.D. (1967), Analysis of Musical Instrument Tones," Journal of Acoustical Society of America, v. 41, no. 4, pp. 793-806.

12. GOLD, B. (1962), "Computer Program for Pitch Extraction," J. Acoustical Society of America, v. 34, no. 7, pp. 916-921.

13.. GOLD, B. and RABINER, L. R. (1969), "Parallel Processing Techniques for Estimating Pitch Periods of Speech in the Time Domain," Journal of Acoustical Society of America, v. 46, no. 2.

14. JUSTICE, J. H. (1979), "Analytic Signal Processing in Music Computation," IEEE Trans. on Acoustics, Speech, and Signal Processing, v. ASSP-27, no. 6, pp. 670-684.

15. LUCE, D. and CLARK, M. (1967), "Physical Correlates of Brass Instrument Tones," Journal of Acoustical Society of America, v. 42, no. 6, pp. 1232-1243.

16.. MIYASAKA, E. (1982), "Timbre of Complex Tone Bursts with Time Varying Spectral Envelope," IEEE, pp. 1462-1465.

17. MOOG, R. (1986), "Digital Music Synthesis," Byte, v. 11, no. 6, pp. 155-170.

18. MOORER, J. A. (1977), "Signal Processing Aspects of Computer Music: A Survey," Proc. of the IEEE, v. 65, no. 8, pp. 1108-1137. .pa

19. POWELL, R. (1986), "The Challenge of Music Software," Byte, v. 11, no. 6, pp. 145-154.

20. RISSET, J.C. and MATTHEWS, M.V. (1969), "Analysis of Musical Instrument Tones," Physics Today, v. 22, no. 2, pp. 23 -30.

21. SONDHI, M. M. (1986), "New Methods of Pitch Extraction," IEEE Trans. on Audio and Electroacoustics, v. AU-16, no. 2, pp. 262-266.

22. STRONG, W. and CLARK, M. (1967), "Synthesis of Wind Instrument Tones," Journal of Acoustical Society of America, v. 41, no. 1, pp. 39-52.

23. STRONG, W. and CLARK, M. (1966), "Perturbations of Synthetic Orchestral Wind Instrument Tones," Journal of Acoustical Society of America, v. 41, no. 7, pp. 277-285.