Efficient Pitch Detection Algorithms for Pitched Musical Instrument Sounds: A Comparative Performance Evaluation

Chetan Pratap Singh
Department of ECE
National Institute of Technology
Warangal, India
chtn.pratap@gmail.com

Dr. T Kishore Kumar
Department of ECE
National Institute of Technology
Warangal, India
Chtn.pratap@gmail.com

Abstract—**Pitch detection of an audio signal is an interesting research topic in the field of speech signal processing. Pitch is one of the most important perceptual features, as it conveys much information about the audio signal. It is closely related to the physical feature of fundamental frequency f_0. For musical instrument sounds, the f_0 and the measured pitch can be considered equivalent. In this paper four pitch detection algorithms have been proposed for pitched musical instrument sounds. The goal of this paper is to investigate how these algorithms should be adapted to pitched musical instrument sounds analysis and to provide a comparative performance evaluation of the most representative state-of-the-art approaches. This study is carried out on a large database of pitched musical instrument sounds, comprising four types of pitched musical instruments violin, trumpet, guitar and flute. The algorithmic performance is assessed according to the ability to estimate pitch contour accurately.**

Index Terms—**Autocorrelation, AMDF, LPC, Cepstrum, Mel, Standard deviation.**

I. INTRODUCTION

Pitch detection of sound signal has consistently been a interesting topic in speech signal processing. It is widely used in music information retrieval (MIR), automatic music transcription, automatic speech recognition (ASR) and speaker identification. Pitch is an attribute based on perception which allows the ordering of sounds on a frequency-based scale extending from low to high values [1]. Most musical instrument sounds have a periodic structure when viewed over short time intervals, and such sounds are perceived by the auditory system as having a quality known as pitch. Like loudness, pitch is a subjective attribute of musical instrument sound that is related to the fundamental frequency of the sound, which is a physical attribute of the acoustic waveform Pitch is one of the most important attributes of pitched musical instruments [1]. Pitch detection Algorithms which tolerate noisy audio input [3] are particularly desirable in musical instrument sound pitch tracking. To avoid initial pitch-detection errors in musical instrument sound applications the following requirements should be satisfied for real time algorithms [5]:

- Efficient to function in real time
- Minimum output delay (latency)
- Accurate in the presence of noise

Basically Pitch detection algorithms (PDAs) can be classified into methods which operate in time-domain, frequency domain or joint time-frequency domain [7]. Time-domain algorithms like Zero Crossing Rate, Autocorrelation utilized sound or audio waveform directly to estimate the pitch period. The frequency-domain algorithms utilize frequency spectrum of an audio signal. Thus simple measurements can be made by extracting more information in frequency spectrum of the audio signal [5]. The joint domain algorithms incorporate features of both the domains for pitch detection. In this paper, the principles of four PDAs Autocorrelation, Average Magnitude Difference Function, Cepstrum method and LPC based algorithm including extraction of pitch pattern techniques are summarized. Some experiments and discussions are presented.

II. THEORY OF PITCHED MUSICAL INSTRUMENTS

In the context of Pitched musical instruments 'Pitched' emphasizes the fact that the instruments are used to play melodies [1]. Percussive instruments like drum are unpitched and produce sounds of indeterminate pitch. Musical instruments utilize pitch relationships to make up melodies and chords. Melody is a combination of pitch and rhythm that produces pitched sounds with musically meaningful pitch. The degree of random variation in instrument sound pitch reveals information about the stability of the source excitation and the strength of its coupling to the resonant body. For example, brasswind instruments, having relatively weak excitation-resonance coupling, show wide pitch variation at onset. Similarly string instruments have a high degree of pitch jitter in tones because of the weak interaction between bow and string. The wide pitch variation of vibrato causes an instrument's harmonic contents to interact with the resonant

978-1-4799-3080-7/14/$31.00 ©2014 IEEE

modes of the instruments, producing amplitude modulation, and that provides a source of information about the instrument's resonant structure. The pitch of a musical instrument note is primarily determined by its fundamental frequency of oscillation as perceived by a human. The pitch of a sound signal is a feature based on perception. Western musical scale is used to define instrument note frequency that is set to 440 Hz. Pitch is measured in units called mels[2] and is related to frequency of a pure tone on a nonlinear scale (Mel scale) by equation:

Pitch in Mels = 1127loge (1 + f/700) (1)

Which is plotted in Figure 1.This expression is calibrated so that a frequency of 1000 Hz corresponds to a pitch of 1000 Mels? Below 1000 Hz, the relationship between pitch and frequency is nearly proportional. For higher frequencies, however, the relationship is nonlinear. For example, (1) shows that a frequency of f = 5000 Hz corresponds to a pitch of 2364 Mels.

III. PITCH DETECTION ALGORITHMS

Proposed pitch detection algorithms are Autocorrelation (AC), Average Magnitude Difference Function (AMDF), Cepstrum (CEP) based method and LPC based method.

A. Autocorrelation (AC) based pitch detection

Autocorrelation (AC) [5], [6] is one of the conceptually simplest time domain signal-processing techniques for estimating the pitch of instrument sound signal [3]. Autocorrelation can be obtained by multiplying the original signal with time-shifted version of itself and to estimate the period of the audio signal average energy in the shifted version of signal should be measured. Pitch is variant over time (fig.2), so the autocorrelation must be applied to short segment of signal (fig.3). This method is based on to detect the peak value of the autocorrelation function in the region of interest. The maxima of the autocorrelation function can be estimated at intervals of the pitch period of the original audio signal. The framewise autocorrelation function of a audio signal segment, *x*(*n*), of a discrete-time signal defined as quite accurate single f_0 estimation, can be estimated simply by an appropriate normalization of the framewise autocorrelation function ($A_c(\tau)$) over N length, defined as

$$A_c(\tau) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)x(n+\tau) \qquad (2)$$

The value of segment x(n) is zero outside the interval $0 \leq x(n) \leq N-1$ .The f_0 of the signal x(n) can be determined as the inverse of the time delay τ that corresponds to the maximum of $A_c(\tau)$ within a predefined range. This method is based on the observation that a sound having harmonic contents has an approximately periodic magnitude spectrum, the period of which is the f_0. This method has an advantage that the computations are somewhat more robust against the imperfection in harmonics of musical instruments.

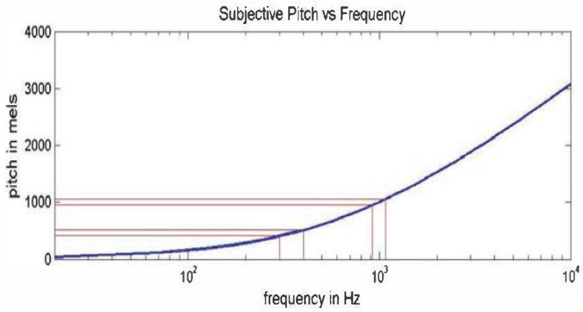

Fig. 1 Relation between subjective pitch and frequency of a pure tone.

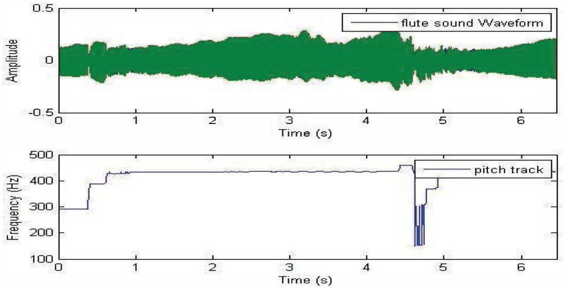

Fig. 2 waveform (flute) and pitch track with autocorrelation

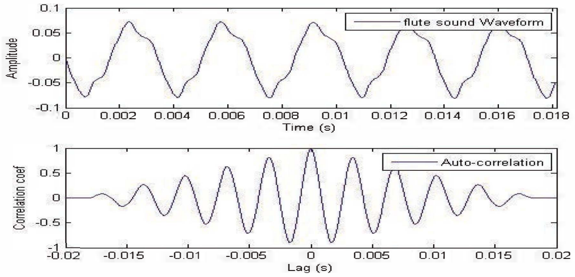

Fig.3 Autocorrelation for short segment of waveform

B. Average Magnitude Difference Function (AMDF)

AMDF is faster approach than ACF and has some similarities with ACF [5],[6].This method works on comparing original and shifted version of given audio signal using differences rather than multiplications .Multiplication is replaced by |s(n+m) – s (n)| .The pitch period is estimated as the value of the delay τ at which the minimum AMDF occurs. The AMDF and the ACF show very similar characteristics. The AMDF produces a notch in characteristics, while the autocorrelation function produces a peak. Higher estimation accuracy and reliability can be obtained using AMDF and the estimation cost with AMDF is less than that with ACF. AMDF computational cost can be reduced by involving multiplication in ACF, the values of AMDF coefficients for a frame of sound segment are set to either zero or one threshold level via a clipping, where the clipping threshold is chosen as 40% of the sum of maximum and minimum of the AMDF values. For periodic signal s(n) of period T the difference function x(m)=s(n+m)-s(n) will be small if m=0,±T,±2T,....

Based on the reason, the short-time average magnitude difference function of a frame of sound signal is defined as

$$x(m) = \frac{1}{N-m-1}\sum_{n=0}^{N-m-1} \left| s(n+m) - s(n) \right| \qquad (3)$$

Where s(n) is instrument's sound signal. N is the length of a frame of that signal. The range of m is between 0 and N. When a frame of sound signal is quasi-periodic,then the values of the AMDF coefficients for that frame of sound signal are also quasi-periodic.

C. Cepstral based method

Cepstrum is derived from Spectrum by reversing first four letters. This method gives the information about rate of change in different spectrum bands. Cepstral method gives a better way than AMDF and ACF for the calculation and tracking of pitch [5]. Pitch tracking is shown in fig. 4 and fig. 5. Cepstrum is nothing but a spectrum of a spectrum. The original sound signal in time domain is transformed into frequency domain using a Fast Fourier Transform (FFT) algorithm and then the obtained spectrum is converted to a logarithmic scale. This log scale spectrum is then transformed using the same FFT technique to obtain the power cepstrum. The power cepstrum reverts to the time domain and exhibits peak values corresponding to the period of the frequency spacings common in the spectrum. Cepstrum of a signal s[n] is defined as

$$c[n] = F^{-1}\left\{\log\left|F\{s(n)\}\right|\right\} \qquad (4)$$

Where F is DFT and F^{-1} is IDFT

For windowed frame of sound signal cepstrum is

$$c[n] = \sum_{n=0}^{N-1} \log\left(\left|\sum_{n=0}^{N-1} s[n] e^{-j\frac{2\pi}{N}kn}\right|\right) e^{j\frac{2\pi}{N}kn} \qquad (5)$$

If the log amplitude spectrum contains many regularly spaced harmonics, then the Fourier analysis of the spectrum will give a peak corresponding to the spacing between the harmonics as shown in fig.6 and fig.7: that is equivalent to fundamental frequency. Effectively we are using the spectrum of sound signal as another signal, and trying to calculate periodicity in the spectrum itself. The unit of x-axis of the cepstrum is quefrency, and peaks in the cepstrum (relating to periodicities in the spectrum) are called rahmonics. To calculate the fundamental frequency from the cepstrum we search for a peak in the quefrency region corresponding to typical speech fundamental frequencies (1/quefrency). Fundamental frequency is estimated in the same way as in the autocorrelation method.

D. LPC based method

Linear predictive coding (LPC) is a very powerful tool in speech processing, also known as auto-regressive (AR) modeling [5].

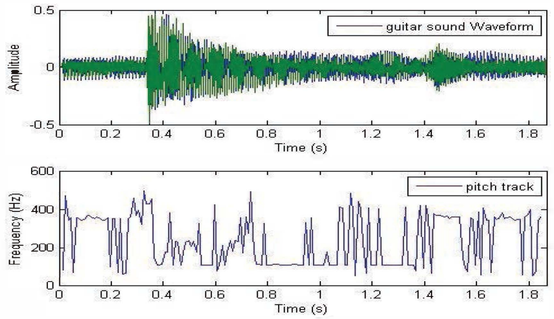

Fig.4 Guitar pitch tracking waveform

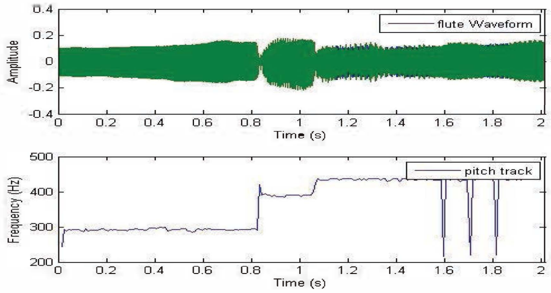

Fig.5 flute pitch tracking waveform

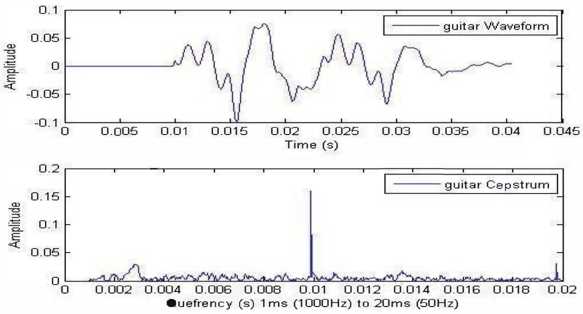

Fig.6 Guitar power cepstrum using cepstrum method

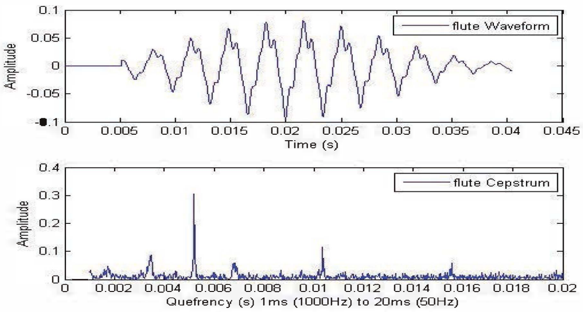

Fig.7 Flute power spectrum using cepstrum method

This method is widely used in speech processing because it is fast and easy to use and it is an effective way to estimate the main parameters of sound signals like musical instrument sounds. LPC behaving like the spectrum estimation method computes

the linear predictive (LP) coefficients of a single frame. From these coefficient values it is possible to generate another synthesized signal which has its spectral contents close to the original one. An all-pole filter with a sufficient number of poles is a good approximation for sound signals. Thus, we could model the filter H(z) as

$$H(Z)=\frac{S(Z)}{E(Z)}=\frac{1}{1-\sum_{k=1}^{p} a_k z^{-k}}=\frac{1}{A(z)} \qquad (6)$$

Where p is order of the LPC model. Taking inverse z transform in (6) results in

$$s(n)=\sum_{k=1}^{p} a_k s[n-k]+e[n] \qquad (7)$$

Linear predictive coding as the name itself tells that it predicts the current sample as a linear combination of its past p samples. By plotting H(e^{jw}), we can see peaks at the roots of the denominator. From this fact, we can detect formant frequencies. The number of LPC coefficients p is computed in (8)

$$p = 2 + \text{sampling frequency}/1000 \qquad (8)$$

In this experiment instrument sound samples are used at 8000Hz sampling frequency, so LPC model order is 10 for pitch and formants analysis. Fig. 8 shows pitch and formants track using LPC and fig. 9 shows frequency spectrum of guitar sound.

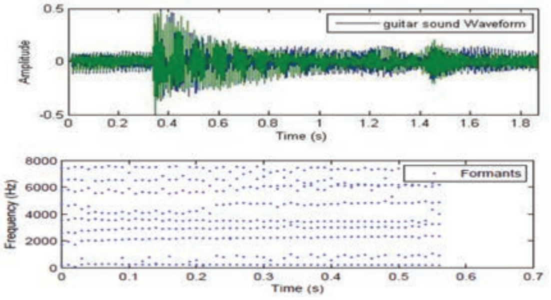

Fig.8 Waveform and formants track with LPC

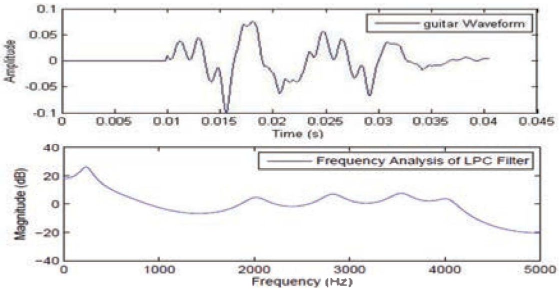

Fig.9 Waveform and Frequency spectrum of LPC filter for frame 5

IV. DATA BASE FOR EVALUATION

To analyse the performance of the four PDAs, a database of musical instrument notes collected from MUMS Library. Samples of MUMS database have a close approximation of the fidelity of real musical instruments. Four pitched musical instruments guitar, flute, trumpet and violin were used for experiment. MUMS samples consist of a tone played by an instrument repeating the same note for a total length of 5 seconds. For each instrument three notes of distinct frequencies were used to develop tonal models for training so the data set consisted of 12 (four instruments three notes) instrument music samples used for analysing the performance of pitch detection algorithms. The instrument music notes sampled at a rate of 44100 Hz stored in WAV format. For experiment purpose the music notes were downsampled to 8000 Hz. To obtain more accuracy in pitch tracking we apply Praat [4] to the synthesized signal of an instrument. The lowest frequency is called the fundamental and the pitch estimated by this frequency is used recognize the instrument note. However, the melodious sound is produced by the combination of this fundamental with a series of harmonics. Most pitched musical instruments produce harmonic sounds, and these can be calculated by multiplying the fundamental by an integer. Therefore, when the pitched instrument tuning note is played, the sound is a combination of 440 Hz, 880 Hz, 1320 Hz, and 1760 Hz and so on.

V. RESULT

The following criteria are used to evaluate the performance of proposed pitch detection algorithms. Accuracy of different PDAs is estimated based on these criteria [5]:

A. *Mean Squared Finite Pitch Error (MSFPE)*

If P_1 is true pitch contour of instrument sound segment and P_2 is detected pitch contour by algorithm then pitch error for nth frame is defined in (9)

$$e(n)=P_1-P_2 \qquad (9)$$

Now if |e(n)|< 10 samples or less than 1.25 ms in estimating pitch period, the error is classified as Fine Pitch Error(FPE)[4],[5].To show the results in terms of Mean Squared FPE we computed standard deviation of fine pitch errors, σ_e as given in (10)

$$\sigma_e=\sqrt{\frac{1}{N_k}\sum_{i=1}^{N_K} e^2(n_i)-m_e^2} \qquad (10)$$

n_i is ith interval in sound segment for which e(n)<10 samples and N_k is total number of such intervals in sound segment. m_e is mean of fine pitch errors and computed in (11)

$$m_e=\frac{1}{N_k}\sum_{i=1}^{N_k} e(n_i) \qquad (11)$$

The standard deviation of the fine pitch errors is a measure of the accuracy of the pitch detection algorithm.

B. *Pitch Error Ratio (E_{rr})*

The pitch error ratio E_{rr} for evaluating the pitch detection performance is defined as the difference between estimated pitch frequency f_t and the true pitch frequency f_0 of musical instrument note over all the frames and is divided by the true pitch frequency f_0.E_{rr} is computed in (12).

$$E_{rr}= \frac{|f_t - f_0|}{f_0} \times 100 \qquad (12)$$

The error analysis discussed above was performed on the data base of Section IV, and the major results are presented in Tables I and II.

TABLE I. MEAN SQUARED FINITE PITCH ERROR

	AC				CEP			
	note 1	note 2	note 3	Average	note 1	note 2	Note 3	Average
Guitar	1.23	1.22	2.10	1.51	1.24	2.33	2.14	1.90
Violin	0.98	1.12	2.13	1.41	0.98	1.13	2.19	1.43
Trumpet	0.97	2.11	1.21	1.43	1.01	2.17	1.22	1.47
Flute	0.99	0.95	1.15	1.03	0.94	1.16	2.10	1.40
Average				**1.35**				**1.55**
	AMDF				LPC			
	note 1	note 2	note 3	Average	note 1	note 1	note 3	Average
Guitar	1.34	2.43	2.45	2.07	1.27	1.22	2.43	1.64
Violin	1.22	1.25	2.12	1.53	1.13	1.32	2.33	1.59
Trumpet	1.98	0.97	1.11	1.35	1.22	2.28	1.07	1.52
Flute	1.19	1.30	2.29	1.59	0.91	2.19	2.44	1.85
Average				**1.64**				**1.65**

TABLE II. PITCH ERROR RATIO

	AC				CEP			
	note 1	note 2	note 3	Average	note 1	note 2	note3	Average
Guitar	0.78	0.91	2.01	1.23	0.98	1.12	2.15	1.42
Violin	1.02	1.09	1.08	1.06	1.04	1.21	2.27	1.51
Trumpet	1.27	2.19	2.17	1.88	1.13	2.20	2.10	1.81
Flute	0.96	1.03	1.04	1.01	1.30	1.34	1.45	1.36
Average				**1.30**				**1.53**
	AMDF				LPC			
	note 1	note 2	note 3	Average	note 1	note 1	note 3	Average
Guitar	1.22	1.29	2.26	1.59	0.89	1.18	2.47	1.51
Violin	1.03	1.08	2.11	1.41	0.99	1.26	1.18	1.14
Trumpet	0.77	1.39	1.17	1.11	1.87	1.45	2.01	1.78
Flute	1.29	1.27	2.37	1.64	1.67	1.73	2.21	2.42
Average				**1.44**				**1.71**

VI. CONCLUSION

In this paper the performance of four pitch detection algorithms using two error measurements has been analysed.The performance of Autocorrelation based algorithm is better than other three algorithms namely CEP, AMDF and LPC. But for trumpet sound AMDF has shown the better results.

VII. REFERENCES

[1] Anssi Klapuri, “Signal Processing: Method for Music Transcription”, Tampere University of Technology, Tampere Finland, eISBN:0-387-32845-9,©2006 Springer Science+ Business Media LLC.

[2] Lawrence R. Rabiner, “Introduction to Digital Speech Processing”, Rutgurs University and University of California, USA, Vol. 1, Nos. 1-2(2007) 1-194.

[3] Tetsuya Shimamura, Member, IEEE, and Hajime Kobayashi, “Weighted Autocorrelation for Pitch Extraction of Noisy Speech”,IEEE Transactions on Speech and Audio Processing,Vol. 9,October 2001.

[4] Onur Babacan, Thomas Drugman, Nicolas d’Alessandro, Nathalie Henrich, Thierry Dutoit, “A Comparative Study of Pitch Extraction Algorithms on A Large Variety of Singing Sounds”, IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013.

[5] Lawrence R.Rabiner,Michael J.Cheng,Aaron E. Rosenburg and Carol A McGonegal, “Comparative Performance study of Several Pitch Detection Algorithm”,IEEE Transactions on Acoustics Speech and Signal Processing,October,24(5):399-418, 1976.

[6] Li Hui, Bei-qian Dai and Lu Wei, “A Pitch Detection Algorithm Based on AMDF and ACF”, IEEE,ICASSP 2006.

[7] Savitha S Upadhya, “Pitch Detection in Time and Frequency Domain”,IEEE International Conference on Communication, Information and Computing Technology,October 2012