A pitch detection/tracking strategy for solo bowed-string and wind musical instrumental recordings is presented. To avoid the missing fundamental problem, we adopted the greatest common divisor method and modified it with a weighted-and-voting technique that can reveal more information of strong partials in the target signal. Moreover, a frame-based correction method with consideration of the performing aspects of the instruments is also proposed to emendate possible misjudgments in the transition from one note to the next note. Experimental results showed that the proposed strategy is superior to three popular methods for a pitch extraction/tracking task. The proposed method was also tested when the sound source is reverberant and the results were compared with other methods, too.

Keywords: pitch detection, pitch tracking, bowed-string instrument, wind instrument, weighted greatest common divisor and vote (WGCDV)

1. INTRODUCTION

Pitch detection, also referred to as fundamental frequency (F0) estimation, is a classical issue in the audio/speech processing areas. Many methods have been proposed in the literature and are still being researched nowadays. For example, zero crossings [1, 2], autocorrelation [3, 4], and harmonic product sum (HPS) [5, 6] are widely used ones. Systematic reviews of more details and other methods can be also found in [7-9]. Developing a “context-free” F0 estimator is a difficult task, whereas context-specific attempts work better in most cases. In most cases, to identify the exact pitch at every time instance may not be necessary because pitch resolution of human sensation is not very high for most people [10]. Even for those who have perfect pitch, they cannot identify the exact pitch every time they are asked for. If a clip of signal is too short, it is almost impossible to identify the pitch for listeners. In fact, many electronic instruments are unable to generate the required pitch for each note and they usually have 0 to 5 Hz deviation from standard pitches. Nevertheless, accurate pitch information is still necessary in applications such as structured audio coding and music information retrieval.
Since pitch is important for speech recognition/synthesis, numbers of pitch detection techniques were designed based on speech data. For examples, the Praat tool [11], developed by Boersma and Weenink aims at analyzing and manipulating digital speech data. Its pitch detection mechanism is practically a mixture of time-domain correlation methods. The STRAIGHT [12] proposed by Kawahara et al., based on the mono vocoder has had very good results for voice recognition and synthesis. Recently, a robust and accurate F0 estimation can be achieved by YIN estimator using the interplay between autocorrelation and cancellation [13]. They all contain a good F0 estimation tool. It is, however, not a trivial task to extract a set of usable pitch information to re-synthesizing recordings of the solos using the above methods [14]. In this paper, we propose a pitch detection/tracking strategy based on characteristics of such audio recordings of playing the instruments such as bowed-string instruments (violin and Erhu), brasses (trumpet) and woodwinds (oboe). They are all sustaining-driven musical instruments of unique and constantly changing timbres controlled by professional players.

The proposed method is basically categorized as a frequency domain approach. Frequency domain approaches can not only provide an estimated pitch contour but also acquire the timbre characteristics while the analysis procedure is done. In musical analysis and synthesis aspects, the pitch detection is not necessarily the first step toward building a synthesis database. Instead, a detailed spectra analysis may obtain both pitch and timbre parameters, especially considering specific instrumental characteristics [14]. Building a practical music synthesis database, however, lies outside the scope of this paper. So, we focus on extracting a set of useful pitch information. The basic procedure is illustrated in Fig. 1. The audio samples are first divided into analysis frames. Then, short-term Fourier transform (STFT) is adopted to convert the data into frequency domain. Based on the harmonic assumption of tones of target instruments, a method called weighted greatest common divisor and vote (WGCDV) is employed to find the possible pitch for each frame. By exploring the relationship among neighboring audio frames according to the knowledge of the instruments and performing aspects, a post-processing called frame based correction (FBC) is designed to correct the possible errors produced from the previous step. The simulation results showed that the proposed approach is more suitable for analyzing solo musical recordings of the target instruments than all previously given tools [11-13].

The rest of the paper is organized as follows. In section 2, the concept of WGCDV method is introduced and its detailed steps are given. FBC is presented in section 3. Computer simulation and case studies are given in section 4. The performances of different methods are also compared. Conclusions and future works are suggested in section 5.

2. WGCDV PITCH DETECTION METHOD

Generally speaking, tones of most of sustaining-driven musical instruments, such as violin and trumpet, can have a longer lasting pitch than those of plucked or struck string instruments, such as guitar and piano. In this point of view, it seems an easier task to extract the pitch information from such specific musical instruments than general cases. However, some performing techniques, especially for sustaining-driven musical instruments, will introduce lots of obstructions to confuse most pitch detection strategies. For
example, there is no fret made for violin and Erhu. Thus, players can play fast trill, vibrato, portamento by taping or sliding the fingers on finger board and strings or applying the larger bowing pressure. All these are common in bowed-string instrument playing. The pitch variation may be over an octave for Erhu playing sometimes. There are other factors that reduce the accuracy of some F0 estimation algorithms. For example, the energy levels of the first two or three partials of Erhu are usually much weaker than higher partials. Based on our observation, such effects greatly bias the estimate. In our experience with different algorithms, if the pitch of a tone is misidentified, it is usually one octave higher or lower than the actual pitch. In fewer cases, it is 7 semitones higher than the actual pitch (1.5 times the actual fundamental frequency). If it falls into 1/2 semitone range, it is usually very close to the actual pitch identified by an invited Erhu player in advance.

As shown in Fig. 1, WGCDV estimates F0 in three steps: (a) locates the peaks of the transformed magnitude information; (b) finds a likely GCD value for each partial pair using a look-up-table method; (c) weights the likely GCD values according to the spectral energy and determine the final GCD by voting. In the following sub-sections, we will discuss each step in more details.

2.1 Locate Likely Partial Positions

Since our goal is to extract the pitch information from strong harmonic musical sig-
After a frame of audio data was transformed into frequency domain, we calculate a smoothed spectrum using the mean filter. In the smoothed spectrum, there are three kinds of points, peak point, valley point, and slope point, with respect to local maximum, local minimum, and others. Taking Fig. 2 as an example, the protrude value $P$ of peak point $A$ then can be defined by

$$ P = \frac{V_A}{\max(V_B, V_C)} $$

(1)

where $V_A$ is the magnitude of peak point $A$; $V_B$ and $V_C$ are the magnitudes of left valley point $B$ and right valley point $C$, respectively.

The protrude value shows how kurtosis a spectral peak is. To further reduce the number of possible partial positions, a protrude threshold $T_P$ ($T_P = 4$ is used in section 4.) is introduced to reject those small peaks. It is noted that to examine the whole spectrum is not necessary because the target instrument always has its only compass. It is applicable, for a target instrument, to analyze from the lowest compass frequency to the frequencies in two or three octaves higher than the highest compass frequency that covers the dominant partials. This principle will apply to most of the procedures described afterward.

2.2 GCD Look-up Table Method

For a pitch detection task, the greatest common divisor (GCD) method is more empirical than time-domain methods in the aspect of human understanding about how to determine the pitch of a sound. However, there are two problems that might decrease the efficiency. First of all, GCD is mathematically defined for positive integer numbers and is obviously limited by the frequency resolution of the transform. An excessive short or long window size introduces a larger offset from the possible pitch position. Secondly, most tones produced by musical instruments are quasi-periodic; the relation among their
partial components is usually inharmonic. For string instruments, the stiffness of the string causes the dispersion phenomenon [15]. It stretches the partial frequencies higher compared to harmonic frequencies. A better solution is to loosen the restriction of integer assumption. Without loss of generality, we can extend the GCD concept to the positive real numbers and use a look-up table (LUT) to map a floating-point quotient to its corresponding harmonic relation by examining which quotient in the harmonic relation table is the closest one to the quotient of a wait-for-examine partial pair. An implemented LUT is illustrated in Table 1. In this LUT, the quotient of any two partials can be calculated and matched with the closest one to determine the most possible harmonic relationship.

<table>
<thead>
<tr>
<th>numerator</th>
<th>denominator</th>
<th>numerator/denominator</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>0.5000</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>0.3333</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>0.2500</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>0.2000</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>0.4000</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>0.1667</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>0.1429</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>0.2857</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
<td>0.4286</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>0.1250</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>0.3750</td>
</tr>
<tr>
<td>1</td>
<td>9</td>
<td>0.1111</td>
</tr>
<tr>
<td>2</td>
<td>9</td>
<td>0.2222</td>
</tr>
<tr>
<td>4</td>
<td>9</td>
<td>0.4444</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>0.1000</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>0.3000</td>
</tr>
<tr>
<td>1</td>
<td>11</td>
<td>0.0909</td>
</tr>
<tr>
<td>2</td>
<td>11</td>
<td>0.1818</td>
</tr>
<tr>
<td>3</td>
<td>11</td>
<td>0.2727</td>
</tr>
<tr>
<td>4</td>
<td>11</td>
<td>0.3636</td>
</tr>
<tr>
<td>5</td>
<td>11</td>
<td>0.4545</td>
</tr>
<tr>
<td>1</td>
<td>12</td>
<td>0.0833</td>
</tr>
<tr>
<td>5</td>
<td>12</td>
<td>0.04167</td>
</tr>
</tbody>
</table>

In section 2.1, it is not accurate enough if peak A in Fig. 2 is used as a partial point. Therefore, one has to estimate a floating-point peak position from the integral positions such as point A, B and C in Fig. 2. In this paper, a simple approximation by using 2nd order polynomial (a parabolic function) is adopted and the detailed algorithm can be found in the appendix. Let $\alpha_i$ represent the estimated floating-point position of the $i$th peak. Before we use Table 1 to calculate a likely GCD for the $(\alpha_i, \alpha_j)$ pair, we need to keep $2\alpha_i < \alpha_j$ because the table was designed in optimal storage requirement and only contained the terms whose denominator is no less than two times of the corresponding numerator. For $2\alpha_i > \alpha_j$ cases, we simply replace $\alpha_i$ with $\alpha_j - \alpha_i$ since the GCD of $(\alpha_i, \alpha_j)$ is mathematically equivalent to the GCD of $(\alpha_j - \alpha_i, \alpha_j)$. 

In section 2.1, it is not accurate enough if peak A in Fig. 2 is used as a partial point. Therefore, one has to estimate a floating-point peak position from the integral positions such as point A, B and C in Fig. 2. In this paper, a simple approximation by using 2nd order polynomial (a parabolic function) is adopted and the detailed algorithm can be found in the appendix. Let $\alpha_i$ represent the estimated floating-point position of the $i$th peak. Before we use Table 1 to calculate a likely GCD for the $(\alpha_i, \alpha_j)$ pair, we need to keep $2\alpha_i < \alpha_j$ because the table was designed in optimal storage requirement and only contained the terms whose denominator is no less than two times of the corresponding numerator. For $2\alpha_i > \alpha_j$ cases, we simply replace $\alpha_i$ with $\alpha_j - \alpha_i$ since the GCD of $(\alpha_i, \alpha_j)$ is mathematically equivalent to the GCD of $(\alpha_j - \alpha_i, \alpha_j)$.
Now we can determine a possible harmonic relation for each \( (\alpha_i, \alpha_j) \) pair from the LUT. A likely GCD \( \gamma_{ij} \) can be calculated directly from divided \( \alpha_i \) by the denominator in the LUT. For example, if the quotient of \( (\alpha_i, \alpha_j) \) is close to 0.4, its harmonic relation will be \( (2, 5) \) and the likely GCD of this pair will be \( \alpha_j/5 \).

### 2.3 Energy Weighted and Voting

After we calculate likely GCDs from all partial pairs in section 2.2, one needs to choose from them to determine \( F_0 \). Since the critical partials are always of higher energy than most other frequency components, we design a weight factor corresponding to each partial pair according to their magnitudes. The advantage is to further reduce the effects of inharmonicity and noise. Let \( \beta_i \) be the corresponding magnitude of \( \alpha_i \) and a weight factor \( w_{ij} \) for \( \gamma_{ij} \) is defined by

\[
w_{ij} = \sqrt{\min(\beta_i, \beta_j)}.
\]

(2)

To start a voting procedure, all likely GCDs are roughly assigned to several musical note partitions determined by a quantization factor \( Q \).

\[
c_{ij} = \text{floor}\left(\frac{\gamma_{ij}}{Q} + 0.5\right).
\]

(3)

Moreover, an indicator function can be defined by

\[
\theta_{ij}(k) = \begin{cases} 1 & \text{if } c_{ij} = k \text{ or } c_{ij} = k + 1 \\ 0 & \text{otherwise} \end{cases}
\]

(4)

Next, the weighted sum of each partition is evaluated by

\[
S(k) = \sum_{i,j} w_{ij} \times \theta_{ij}(k).
\]

(5)

The most probable pitch position will fall into the partition of the greatest weighted sum. The centroid method [16] is then used to calculate a more accurate pitch position \( r \) by involving all the likely GCDs in that partition, i.e.,

\[
r = \frac{\sum_{i,j} \gamma_{ij} \times w_{ij} \times \theta_{ij}(k)}{S(k)}.
\]

(6)

With the window size \( W \) and sampling frequency \( F_s \), the estimated fundamental frequency \( f_p \) is obtained by

\[
f_p = \frac{r}{W} \times F_s.
\]

(7)
3. FRAME BASED CORRECTION METHOD

In some occasions, very weak and unstable tones are produced because of light and uneven bowing or blowing pressure. In such cases, fundamentals may disappear or the tones are too weak to be detected for many pitch detection algorithms including the proposed WGCDV method. No matter how accurate a F0 estimation method based on a single audio frame is, its accuracy can be improved if the context information from consecutive frames is involved. The basic assumption of the pitch correction procedure is that the pitch from a note of any musical performance won’t (may not) change abruptly. Thus, the first step is to segment the source into different note regions. In general, the spectrum has a large change in both timbre and energy in the transition region between two notes. A measure is defined in Eq. (8) to determine the degree of change for two successive frames.

\[ d = \sum_{f} \left| \frac{A_i(f) - A_{i-1}(f)}{A_i(f)} \right| \]  

where \( f \) is the frequency index and \( A_i(.) \) is the spectral magnitude function of the \( i \)th frame.

It is worthy to note that \( d \) is equal to zero only when the spectra of two adjacent frames are identical. The degree of change will increase, whether the energy steeply varies or the timbre is reshaped. When \( d \) is greater than 0.7, a note change is considered.

Another measure is that the duration of one note cannot be shorter than the human physical reaction time. Because of the skill limitation of a human performer, two changing points should not occur in a very short time, said less than one semiquaver or one eighth second. In such a situation, one of the changing points can be eliminated to get a clear cut between two notes. A reference pitch of each note region can be decided by using the median of all estimated pitches of the frames in the note region after note regions are segmented. As shown in Fig. 3, the note region designated between changing points \( g \) and \( h \) is shorter than 0.125 second (about 5 hop sizes if the hop size is 1024 at 44.1 kHz sampling rate). The changing point \( h \) should be removed because the estimated pitch of point \( h \) is different from the estimated pitches of its adjacent frames.

As mentioned above, we suppose that there should not be any abrupt and large pitch change within a note region. Small changes in pitch are, however, allowed due to that vibrato and portamento are common playing techniques for the target musical instruments in this paper. Fortunately, the pitch changes caused by vibrato and portamento within a short period of time are usually less than one octave. Thus, if the pitch variation of adjacent frames is larger than an octave or the estimated pitch of one frame is an octave offset from the reference pitch in this note region, FBC assumes that there is an error to be corrected. The new pitches of misjudged frames will be interpolated from those of neighboring frames as the example shown in Fig. 4. Although the proposed FBC was designed according to specific characteristics of some musical instruments, it can be modified for other situations, such as human voices, by taking consideration of vocal features. It is also noted that the FBC method is developed independently of the WGCDV method and can be applied to other pitch detection schemes, too.
4. EXPERIMENTAL RESULTS AND DISCUSSION

Recordings of solo performance using Erhu, trumpet and violin are adopted to test the proposed strategy. A synthesis song produced by a Wavetable synthesized oboe is also provided as a contrast set. Some mono sound materials are sampled in 44.1 kHz with 16 bits resolution and available on [17]. The experimental results of WGCDV, HPS, Praat, and YIN are listed in Table 2. Each of the methods combined with FBC are also tested. The frame size is 2,048 with 50% overlap between two adjacent frames and the STFT window type is Hamming.

An estimation error rate is designed to evaluate their performances and can be calculated by

$$e = \frac{F_{\text{error}}}{F_{\text{total}}} \times 100\%,$$

where $F_{\text{total}}$ is the total number of non-silence audio frames and $F_{\text{error}}$ is the number of frames where wrong estimates occur. The actual pitch of each frame is identified manually by a musician who is an Erhu player. When the estimated pitch falls within a half of a semitone distance from the actual pitch (about 2.973% margin), it is denoted as a correct estimate. Table 2 shows the performances of the methods. While most methods are quite good for signals that are easy to analyze, such like the synthetic sound, there are
some occasions that can make the detectors confused. To illustrate the reliability of these methods in more details, we discuss three special cases as follows.

The first case is the missing fundamental problem. The second tone shown in Fig. 5 is a typical missing fundamental sound in which the fundamental’s energy is far below the other partials’. Spectrogram clearly shows that the energy of the fundamental component (~ 300 Hz) stays below the noise floor. The actual (solid line) and estimated pitch contours are indicated in bottom subplot.

Most detectors mentioned in this paper work well except some reasonable errors due to the strong energies of the second and fourth partials. After FBC is applied, most errors can be corrected. It is noted that Praat failed the test in the later half of the tone. In
addition, perceptual based detectors should be good performers in this regard, too [18].

The second case is the under-estimated case. The top subplot in Fig. 6 shows its spectrogram in which we can observe that there is strong energy appearing in the regions around $0.5 \cdot F_0$ and $1.5 \cdot F_0$ as well. This often appears when Erhu is played with low bowing speed and small bowing pressure. For most frequency-domain methods, detection errors easily occur because of its seemingly harmonic structure. Compared to HPS, the proposed WGCDV method prevents some misjudgments due to the weighted and voting strategy.

![Spectrum](image1)

(a) Spectrum.

![Pitch Contours](image2)

(b) Actual (solid line) and estimated pitch contours.

Fig. 6. Under-estimated case.

The third case is the reverberation case. We use the reverb function of Adobe Audition 2.0 to add different degrees of reverberation (delay time = 50, 100, and 150 ms, respectively). Fig. 7 shows the spectrograms of a synthesis signal and a processed signal. Table 3 shows the results of all the methods. One can see that the proposed method performs better in the high reverberant case, but YIN and PRAAT outperform the proposed method in the other two cases. An oboe song synthesized with wavetable synthesis method is used for example. It is noted that a clear harmonic structure remained due to the lingering sound of the preceding tone. This phenomenon confuses all pitch detectors and delays the correct estimation of the new pitch. The WGCDV method is again benefit from the weighting and voting strategy and has the best average performance.

The last experiment is to test the accuracies of all methods. Synthesis signals of different pitches are produced. The pitches are 440 Hz, 450 Hz, 460 Hz and 470 Hz, respectively. Table 4 shows the average results of 80 frames. It is found that PRAAT is the best performer. WGCDV and YIN performs less well at 460 Hz, but the error is still much less than a semitone.

Similar experiments and analysis are performed over various bowed-string and wind musical instruments. In our experiments, bowed-string instruments are more difficult
than wind instruments. The reasons why these methods produced unsatisfactory results are quite similar. That is, the testing samples are extracted from commercially available compact disks, and they usually contain certain degree of reverberation. The proposed WGCDV + FBC method performs well in the provided samples. However, all methods performed poorly if the signals are overly reverberant. One overly reverberant example can be heard from [17]. More investigation is required in this aspect.
5. CONCLUSION

A pitch detection method called weighted greatest common divisor and vote (WGCDV) for recordings of solo bowed-string and wind instruments is presented. The proposed method was tested over a wide range of audio recordings extracted from commercially available compact disks. The idea of GCD look-up table method makes the GCD approach detour its mathematical restriction and provide a more intuitive estimate than the traditional way. Based on the performing aspects of the target instruments, a frame-based correction (FBC) method is also proposed to track the pitch contour and improve the existing methods. The proposed strategy is also compared favorably to several pitch tools and achieves a better performance in most test recordings. As mentioned in [14], tracking the rapid pitch variation more accurately may be more important than finding the exact hertz of a tone. Most listeners do not feel the pitch problem with the re-synthesis results if there is no large pitch tracking error. This re-synthesis software is also available at [17] for reference. The lightweight computation makes the proposed strategy a practical approach to design a real-time analysis and synthesis application for solo bowed-string and wind instruments.

APPENDIX

In this appendix results required for the parabolic approximation are derived. First of all, we try to find a peak from three adjacent points \((x_1, y_1), (x_2, y_2), \) and \((x_3, y_3)\) with the following relationships,

\[
\begin{align*}
    x_1 &= x_2 - 1 \\
    x_2 &= x_2 + 1 \\
    y_2 &= y_1 \\
    y_2 &= y_3
\end{align*}
\]

(10)

The first two relationships indicate that the parabolic function can be centered upon the second point and the corresponding coordinates can be rewritten as \((-1, y_1), (0, y_2), \) and \((1, y_3)\) in the first place. Now we start from a generic parabolic function Eq. (11) to interpolate the peak point \((x', y')\) as illustrated in Fig. 8.

![Fig. 8. Three-point parabolic approximation.](image-url)
\[ y = ax^2 + bx + c. \quad (11) \]

Substituting the given three points for the equation variances, we have

\[
\begin{align*}
    y_1 &= a - b + c \\
    y_2 &= c \\
    y_3 &= a + b + c
\end{align*}
\]

We can derive \( a \), \( b \), and \( c \) from Eq. (12),

\[
\begin{align*}
    a &= \frac{y_1 + y_3 - 2y_2}{2} \\
    b &= \frac{y_3 - y_1}{2} \\
    c &= y_2
\end{align*}
\]

As we know, the peak point exists where the first-order derivative is zero,

\[
\frac{\partial y}{\partial x} = 2ax + b = 0. \quad (13)
\]

The peak position will be \(-b/2a\) and its value will be \(c - (b^2/4a)\). As a result, the formula solutions can be written as

\[
\begin{align*}
    x' &= -\frac{b}{2a} = \frac{y_3 - y_1}{2(y_3 + y_1 - 2y_2)} \\
    y' &= c - \frac{b^2}{4a} = y_2 - \frac{(y_3 - y_1)^2}{8(y_3 + y_1 - 2y_2)}.
\end{align*}
\]

It is noted that the approximated peak position is denoted as an offset related to the second given point.

REFERENCES


Yi-Song Siao (蕭義松) received his M.S. and B.S. degrees in Computer Science and Information Engineering from National Cheng Kong University, Tainan, Taiwan, in 2003 and 2005 respectively. He began learning the Erhu at the age of thirteen, and extended this interest to his study. In 2004, he proposed the JavaOL (120th Convention AES, May 2006) concept which improves the performance and flexibility of the MPEG-4 Structured Audio. In 2005, he applied the additive synthesis method on synthesizing the Erhu sound and built an interactive analysis/synthesis tool. His research interests include computer music, audio signal processing, GUI design, and computer graphics.
Wei-Chen Chang (張維城) was born in Taipei, Taiwan, R.O.C., in 1975. He received the B.S. degree in Mathematics, the M.S. degree and the Ph.D. degree in Computer Science Information Engineering from National Cheng Kung University, Taiwan, in 1997, 2002 and 2008, respectively. From 2007 to 2008, he was a visiting scholar in IRCAM, Paris, where he worked on polyphonic estimation and tracking. His research activities include data compression, signal processing, model-based music synthesis, and machine learning.

Alvin W. Y. Su (蘇文鈺) received his B.S. degree in Control Engineering from National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 1986. He received his M.S. and Ph.D. degrees in Electrical Engineering from Polytechnic University, Brooklyn, New York, in 1990 and 1993, respectively. From 1993 to 1994, he was with CCRMA, Stanford University, Stanford, California. From 1994 to 1995, he was with CCL (Computer and Communication Lab.), ITRI, Taiwan. In 1995, he joined the Department of Information Engineering, Chun Hwa University, Hsinchu, Taiwan. In 2000, he joined the Department of Computer Science and Information Engineering of National Cheng Kung University (NCKU), where he served as an Associate Professor. He is the director of Campus Information System Group of NCKU. He is the director of SCREAM (Studio of Computer REseArch on Music and Multimedia), NCKU. His research interests cover the areas of digital audio/video signal processing, physical modeling of acoustic instruments, multimedia data compression, P2P multimedia streaming systems, embedded systems, VLSI signal processor design and ESL (Electronic System Level) tool design.