Time-Scale Modification of Audio Signals Using Enhanced WSOLA With Management of Transients

Shahaf Grofit, *Student Member, IEEE*, and Yizhar Lavner, *Member, IEEE*

***Abstract*—In this paper, we present an algorithm for time-scale modification of music signals, based on the *waveform similarity overlap-and-add* technique (WSOLA). A well-known disadvantage of the standard WSOLA is the uniform time-scaling of the entire signal, including the *perceptually significant transient* sections (PSTs), where temporal envelope changes as well as significant spectral transitions occur. Time-scaling of PSTs can severely degrade the music quality. We address this problem by detecting the PSTs and leaving them intact, while time-scaling the remainder of the signal, which is relatively steady-state. In the proposed algorithm, the PSTs are detected using a Mel frequency cepstrum nonstationarity measure and the normalized cross-correlation, with time-varying threshold functions. Our study shows that the accurate detection of PSTs within the WSOLA framework makes it possible to achieve a higher quality of time-scaled music, as confirmed by subjective listening tests.**

***Index Terms*—Mel frequency cepstrum, spectral variation, time-scale modification of audio and music signals, waveform similarity overlap-and-add (WSOLA).**

I. Introduction

TIME-scale modification (TSM) of audio signals is the process of modifying the duration of a signal, while maintaining other qualities, such as the pitch and the timbre, unchanged [1]–[4]. The purpose of time-scaling is to change the rate at which acoustic events are experienced, while retaining their perceived naturalness. In other words, we would like to produce sounds that are perceived as if they had been generated by the original physical mechanism operating at a modified rate. For example, time-scaling of a sound produced by a musical instrument would ideally be perceived as if the performer had been playing the same instrument at a different tempo.

Present techniques for TSM of audio signals are used for many applications; for instance, for synchronization between different sounds or between the audio and the video components in recording studios [5], for musical transformations [6], and for gestural control of music or spoken voice [7]. TSM of speech signals can be used in answering machines, where speeding up the speech can save both storage and the time of listening to messages, in learning new languages [8], and in applications for the hearing impaired [9].

Manuscript received June 21, 2006; revised September 6, 2007. This work was supported in part by a Guastella Fellowship of the Sacta-Rashi Foundation, and the JAFI Project. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Michael Davies.

S. Grofit was with the Department of Computer Science, Tel-Hai Academic College, 12210 Upper Galilee, Israel. He is now with the School of Computer Science, Tel-Aviv University, 69978 Tel-Aviv, Israel.

Y. Lavner is with the Department of Computer Science, Tel-Hai Academic College, 12210 Upper Galilee, Israel (e-mail: Yizhar_l@kyiftah.org.il).

Digital Object Identifier 10.1109/TASL.2007.909444

Various algorithms have been proposed in the past three decades for high-quality time-scaling of speech signals; for example, time-domain harmonic scaling (TDHS) [10], the harmonic plus noise model (HNM) [11], [12], linear prediction [13], and phase vocoder [1]–[4], [14]. Algorithms for TSM of speech based on time-domain *synchronized overlap-and-add* (SOLA) [3], [15]–[17] such as the *waveform similarity overlap-and-add* (WSOLA) [18], were shown to achieve very good results at a low computational cost, and thus suitable for real-time synthesis systems [17]. Unfortunately, applying the same algorithms to music signals often causes noticeable degradations in sound quality. One of the reasons for this degradation is the distortion caused by the OLA TSM to the temporal envelope of music signals, which is known to be important for perceptual quality [19], [20]. When time-scaling is performed, transients, such as attacks and decays, can be either smeared or removed; in both cases, introducing artifacts. An improvement may be achieved by keeping the transient sections without modifications. For this purpose, accurate detection of the transients is required.

Detection of nonstationarities was shown to be useful for the time-scaling of speech signals [21]–[23]. Transient detection in music and audio signals has also been addressed in several studies for different applications, such as segmentation and editing of audio recordings [24], improving the resynthesis stage in music analysis-resynthesis [25], audio effects [26], or TSM [27]–[29]. Various methods, using different parameters of the audio signal, are used for the detection. For example, in [25], a weighted energy ratio of the high-frequency content between successive frames is proposed as a measure. In [30], aggregates of significant peaks in different frequency bands are used to define the transients. In [31], the transients are classified based on the variance of the spectrum and on the time offset of the center of gravity of windowed frames. Both energy and phase information are used for onset detection in [26]. In the latter work, differences in phase increments between adjacent frames are used to indicate the presence of the transient components, in a framework of multiresolution analysis. In another study, a cross-entropy measure is applied [32].

In many of the studies where transient detection was used to improve algorithms for TSM of music signals [2], [27]–[29], [33], the time-scaling is performed using a variation of the phase vocoder [34], [35]. Bonada [27] proposed a frequency domain technique that processes the fast changes in the signal (assumed as attacks) differently from other components. The perceptual features of the original signal in these regions are preserved by keeping their original phases. Another phase vocoder-based al-

gorithm for time-scaling that considers the attack transients is proposed by Roebel [29], using the center of gravity of spectral energy for the detection of transient peaks and a method to preserve these peaks during time stretching. In both studies [27], [29], the quality of the resulting time-scaled music is very high.

In this paper, we present an algorithm for time-scale modification of audio signals, which is based on the SOLA approach [16]. Although this approach is simple, nonparametric, and computationally efficient, efforts to improve it and adapt it to music and other audio signals, by considering the transients, have been carried out only in few studies [21], [36]. For example, in the study of Lee *et al.* [36], an improved SOLA algorithm is suggested for the TSM of speech signals, where modification is applied to steady-state sections only. However, the nonnormalized cross-correlation between overlapping frames suggested in that study for transient detection may be sensitive to changes in the recording level. Another shortcoming of [36] is that the thresholds are fixed and should be manually preset for each set of speech recordings.

The basic assumption of SOLA is that the signal is quasi-periodic in the time domain, and its spectral characteristics are relatively invariant for short durations. For TSM, the signal is typically divided into short overlapping frames, and the time-scaled signal is synthesized by overlapping and adding the frames, according to the required time-mapping function, while preserving the original spectral parameters and their relative location for each frame, as in the original signal.

The quasi-periodicity assumption used in SOLA methods can fail not only during transients, but also in polyphonic sounds, where the necessary synchronization between the overlapping frames cannot be achieved for all sources. A possible solution to this problem is presented in [37], where a subband framework is used [37], [38].

In this paper, our efforts are restricted to dealing mainly with the problem of transients. We propose a new method for locating and selecting *perceptually significant transients* (PSTs), using a measure based on the Mel frequency cepstrum and the normalized cross-correlation, with time-variant threshold functions. Introducing a few other enhancements in the framework of the WSOLA algorithm, we demonstrate improved performance, especially for monophonic harmonic music.

This paper is organized as follows. In Section II, the time-domain WSOLA-based algorithm for TSM is presented, along with the adjustments for differential time-scaling. The methods for the PST detection are discussed in Section III. Finally, in Section IV, listening tests comparing the proposed algorithm with other well-known TSM algorithms are described.

II. WSOLA Algorithm Adjusted for Differential TSM

The algorithm for TSM proposed here is based on detecting the events considered as PSTs, and preserving them unmodified, that is, without time-scaling. The remaining, relatively stationary sections are time-scaled using a WSOLA-like method, the main principles of which are presented in Section II-A, followed by the differential TSM algorithm in Section II-B. Section II-C discusses the special manipulations required for the first and last frames of the WSOLA time-scaled sections. Section II-D describes the corrections that have to be applied to the mapping function, to compensate for the nonscaled intervals. Finally, a method to achieve precise time-scale modification to a desired target length is presented in Section II-E.

A. WSOLA Algorithm

Most time-scaling algorithms that use the *overlap-and-add* (OLA) technique are based on minimizing a distance function between short-time Fourier transforms (STFTs) of the original signal and the time-scaled signal, in corresponding neighborhoods, according to a linear mapping function $\tau(m) = a \cdot m$, where m is the sample index and $a \in \mathbb{Q}$ is the time-scaling factor [1].

Unfortunately, this straightforward solution destroys the original phase relationships, and does not maintain the quasi-periodic structure of the signal. The WSOLA algorithm [17], [18] aims at ensuring continuity of the time-scaled signal, by selecting, in each synthesis step, an input frame, with the highest similarity to the natural continuity that exists in the original input signal [38].

Given $S_1, S_2 \in \mathbb{N}$, such that $\tau(S_1 \cdot k) = S_2 \cdot k$ for all k, and $a = S_2/S_1$ is the time-scaling factor, the WSOLA synthesis equation is

$$y(m) = \frac{\sum_{k=-\infty}^{\infty} x(m + S_1 \cdot k - S_2 \cdot k + \Delta_k) \cdot w(m - S_2 \cdot k)}{\sum_{k=-\infty}^{\infty} w(m - S_2 \cdot k)} \tag{1}$$

where S_1 and S_2 are the corresponding step sizes at the analysis signal $x(m)$ and the synthesis signal $y(m)$, respectively, $w(n)$ is a symmetric low-pass window of finite duration, k is the step index, and the time shift of Δ_k keeps the phase continuity of the original signal by allowing a local adjustment in the selection of analysis frames in the input signal.

The equation and the calculations can be simplified by defining the support of $w(m)$ to be $2 \cdot S_2$, and by denoting $v(m)$ the normalized window

$$v(m) = \begin{cases} \frac{w(m)}{w(m)+w(S_2+m)} & -S_2 \le m < 0 \\ \frac{w(m)}{w(m)+w(m-S_2)} & 0 \le m < S_2 \end{cases} \tag{2}$$

so that $\forall m \ \sum_{k=-\infty}^{\infty} v(m - S_2 \cdot k) = 1$.

Therefore, the WSOLA equation is reduced to the following [18]:

$$y(m) = \sum_{k=-\infty}^{\infty} x(m + S_1 \cdot k - S_2 \cdot k + \Delta_k) \cdot v(m - S_2 \cdot k). \tag{3}$$

In our study, Δ_k is selected to achieve the maximal normalized correlation between the overlapping parts of windowed frames

$$\Delta_k = \arg\max_{|\delta| \le \Delta_{\max}} \{C(k, \delta)\} \tag{4}$$

where $C(k, \delta)$ is defined, as shown in (5) at the bottom of the next page. The value of $\Delta_{\max}$ should be about half the assumed maximal pitch period for synchronization between ad-

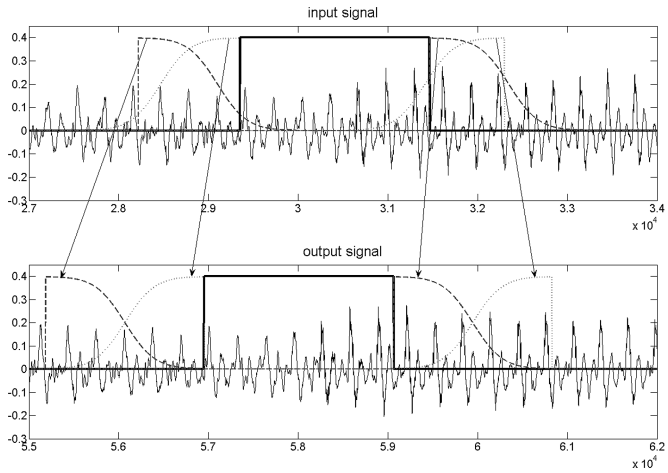


Fig. 1. Schematic representation of the time-scaling with $a > 1$ (slowing down) for sections containing significant transients. The left part of the signal demonstrates a TSM by overlap-and-add of two half-windows, where transients have not been detected. The middle part of the input signal (top) includes a transient, which is copied to the output signal (bottom) without overlapping. In the right part, TSM is performed again.

jacent pitch periods, and additionally, $\Delta_{\max} < S_1/2$ in order to prevent time reversal.

B. Differential TSM Considering PSTs

A straightforward way to implement the WSOLA (3) is to construct $y(m)$ by adding one windowed frame for each step k. In this paper, we use an alternative implementation, where in each step, the sum of two half-windows (previous right half-windowed frame and current left half-windowed frame) is added to the newly constructed output signal (a total of S_2 samples per step). Whenever a PST is detected, no overlap-and-add is carried out and a section of J samples (where J is the resolution of the PST detector, typically equivalent to 3 ms) is copied from $x(m)$ to $y(m)$, unmodified. The process is then repeated with the next pair of half-windows and so forth, until a PST is no longer detected. With this approach, the decision of whether to apply time-scaling or not is made in each step without affecting other synthesized sections. The OLA time-scaling procedure in the presence of PSTs is demonstrated in Fig. 1.

Detection of PSTs is performed using a combination of two criteria: the Mel cepstrum nonstationarity measure and the normalized cross-correlation, both of which are detailed in Section III. At each step of the algorithm, the cepstral distance measure $D(n)$ is computed for sample points inside the support of the frames that should be overlapped. Additionally, the maximum normalized cross-correlation $C(k, \Delta_k)$ is computed (as part of the WSOLA analysis process explained in Section II-A). Both $D(n)$ and $C(k, \Delta_k)$ are compared to their corresponding threshold functions $Th_D(n)$ and $Th_C(k)$, which vary over time.

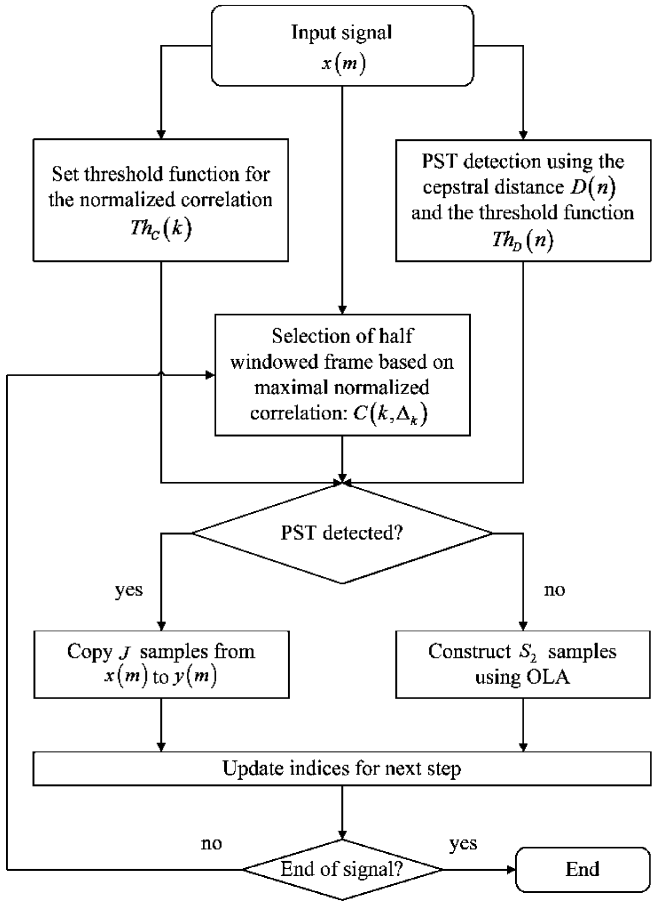


Fig. 2. Schematic description of the main algorithm. In each step, PST detection is performed using a combination of the Mel cepstrum nonstationarity measure and the normalized cross-correlation.

In step k, a section of the signal is considered to be a PST if at least one of the following holds: either $D(n) > Th_D(n)$ for any sample point inside the common support of the frames, or $C(k, \Delta_k) < Th_C(k)$. A schematic block diagram of the general algorithm is depicted in Fig. 2.

Since the cross-correlation, which is one of the measures used for transient detection, is inherently computed for the WSOLA procedure, the additional cost of the transient detection is just the computation of the cepstral distance measure, which is fairly low, even for real-time implementations.

$$C(k,\delta) = \frac{\sum_{m=0}^{S_2-1} x(m + S_1 \cdot (k-1) + \Delta_{k-1}) \cdot x(m + S_1 \cdot k - S_2 + \delta)}{\sqrt{\sum_{m=0}^{S_2-1} x^2(m + S_1 \cdot (k-1) + \Delta_{k-1})} \cdot \sqrt{\sum_{m=0}^{S_2-1} x^2(m + S_1 \cdot k - S_2 + \delta)}} \tag{5}$$

C. Manipulating the Edges

The WSOLA (3) does not provide a method for dealing with the first and the last windowed frames. In the implementation presented previously, the first S_2 samples are constructed by copying the first right half-windowed frame from the input to the output signal, and overlapping and adding the corresponding left half-windowed frame. It is reasonable to take the first right half-windowed frame starting at the first sample. Unfortunately, on many occasions (usually when slowing down the signal), the expected location of the left frame in the input signal may precede the input signal left delimiter. In order to avoid this problem, we start the algorithm after copying the first $\max\{S_2-(S_1-\Delta_{\max}),0\}$ samples (the largest possible overflow) from the input signal to the output signal.

The algorithm terminates when at least one of the next pair of half-windowed frames might exceed the input signal right delimiter. Then the remainder of the input signal (starting at the end of the last left half-windowed frame) is appended to the output signal.

This remainder is determined by the position of the last left half-windowed frame, and therefore its length (as the whole output signal length) may vary within a range of $2\Delta_{\max}$.

D. Correction of the Mapping Function to Compensate for Nonscaled Intervals

The existence of nonscaled intervals inevitably modifies the required mapping function. For example, assume a constant mapping function with a scaling factor $a_c = 1.5$, and suppose that 10% of the signal is selected for intact replication. The actual scaling ratio will be $a = 1.5 \cdot 0.9 + 0.1 \cdot 1 = 1.45$. Unfortunately, the total duration of the intact sections cannot be accurately predetermined, and a constant mapping function that provides the required time-scale factor cannot be evaluated in advance. Instead, the scaling factor, $a = S_2/S_1$, is adjusted locally in each frame. In our algorithm, we only change S_1 and leave S_2 constant, to avoid modifying the window function. The difference between the desired and the actual scaling is compensated gradually by changing S_1^k, so that for each frame index k

$$S_1^k = S_2 \cdot \frac{N_{\text{fix}}}{\left(a_c \cdot X_{\text{offset}}^k - Y_{\text{offset}}^k\right) + a_c \cdot N_{\text{fix}}} \tag{6}$$

where S_1^k is the modified S_1 corresponding to the step index k, Y_{offset}^k and X_{offset}^k are the respective durations of the already constructed output signal and the corresponding input signal (without taking into account the effect of Δ_k), N_{fix} is the compensation time-constant, and a_c is the desired constant scaling factor. The value of N_{fix} is set to be half a second. The value of S_1^k is modified according to the difference between the desired duration $a_c \cdot X_{\text{offset}}^k$ and the actual duration Y_{offset}^k. Fig. 3 demonstrates the short-time variations of the scaling factor.

E. Exact Time-Scale Modification With PSTs

Due to the required special handling of the edges (Section II-C), neither the original WSOLA algorithm [18] nor the above method can guarantee exact time-scaling. In this section, we propose an alternative technique that enables exact time-scaling of a given input signal of length X_{Len} into an output signal of length Y_{Len}, with precision of up to a few samples.

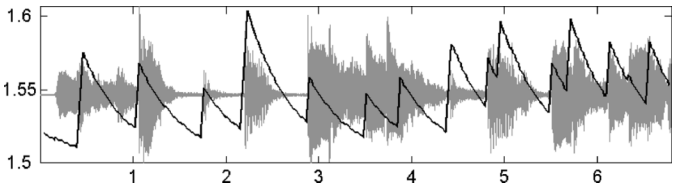


Fig. 3. Time-scale factor variation over time, to compensate for the changes induced by the intact sections. The original signal (gray) is depicted with the mapping function over time (dark line). Whenever a PST is detected and copied unmodified, the time-scale factor is increased and then slowly decreased. (Requested time-scale factor is $a_c = 1.5$.)

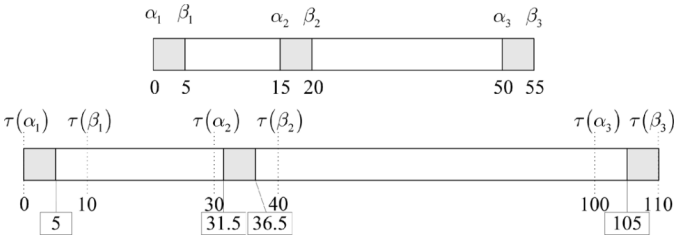


Fig. 4. Location of the nonscaled intervals when exact time-scaled modification is applied. The signal is stretched by a factor of 2 and contains three detected PSTs.

First, the entire signal (or a segment of it) is processed by the PST detection algorithm, determining which sections contain transients and need to be copied intact. Suppose that M sections are chosen for intact replication. Denote these sections $\{[\alpha_1,\beta_1],[\alpha_2,\beta_2],\ldots,[\alpha_M,\beta_M]\}$, where α_i and β_i are the respective left and right delimiters of the ith intact section and $\alpha_1 < \beta_1 < \cdots < \alpha_M < \beta_M$. Each section $[\alpha_i,\beta_i]$ is mapped to its corresponding location $[\delta_i,\varepsilon_i]$ in the target signal, as follows:

$$\begin{aligned}\delta_i &= \tau(\alpha_i)\cdot(1-\phi_i) + (\tau(\beta_i) - (\beta_i-\alpha_i))\cdot\phi_i \\ \varepsilon_i &= \delta_i + (\beta_i - \alpha_i)\end{aligned} \tag{7}$$

where

$$\phi_i = \frac{\alpha_i}{X_{\text{Len}} - (\beta_i - \alpha_i)} \in [0,1] \tag{8}$$

is a weighting factor, determining the exact location of the PST starting point, δ_i inside the interval $[\tau(\alpha_i),\tau(\beta_i)]$ in the synthesis signal. The ending point ε_i is chosen to preserve the length of the PST. It can be observed that ϕ_i is rising with α_i, so the PSTs at the beginning of the signal are mapped to the beginning of their corresponding intervals in the synthesis signal, while the PSTs at the end are mapped to the end of the intervals, as demonstrated by Fig. 4.

In sections that are not selected for intact replication, time-scaling is performed with special manipulations in order to meet the precision requirement: enforced confidence segments of $\max\{S_2+\Delta_{\max}-S_1,0\}$ samples are copied from both edges of the original segment and the last Δ_k is chosen by considering both the last right half-windowed frame and the enforced left half-window (see Fig. 5). Unfortunately, the requirement for exact time-scale modification may cause tempo distortions due to the increased length of intact sections, or

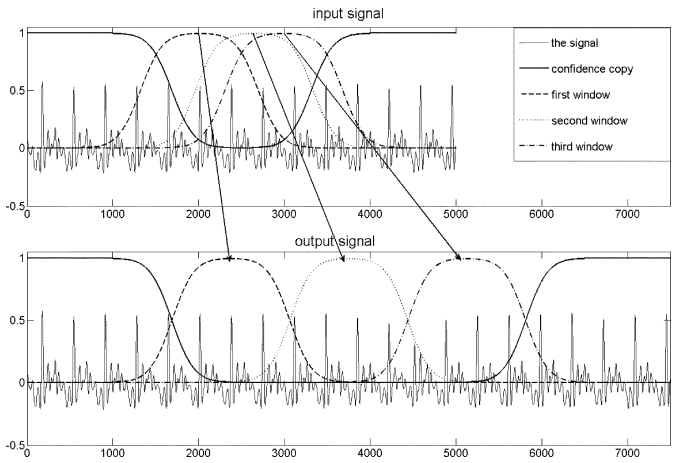


Fig. 5. Exact time-scale modification. The solid curve in both the input and the output graphs represents the enforced confidence segments and the half-windows determined by them. After setting the first (dashed) window and the second (dotted) window, the third (dash-dotted) window's Δ_k must be chosen with consideration of both the dotted line (previous frame) and the solid line (confidence segment). This situation might be problematic for phase synchronization. It is also clear that if the segment is short, the confidence segments will cause great changes to the time-scaling factor.

phase discontinuities. The problem is more difficult in sections with a high rate of transients.

III. Detection of Signal Transients and "Untouched" Sections

This section presents two methods for measuring nonstationarities. The first method uses a distance function based on the Mel frequency cepstrum coefficients (MFCC), described in Section III-A. The method itself is detailed in Section III-B, followed by a description of the threshold function in Section III-C. The second method, presented in Section III-D, uses the normalized correlation data, which is computed as part of the OLA process.

A. MFCC Computation

The Mel cepstrum [39] is one of the most common spectral representations of audio signals. It is based on characteristics of the human auditory system, such as the nonlinear frequency perception [40] and the existence of critical bands [41]. The MFCCs are known to be very efficient in various speech and speaker recognition algorithms [42]–[44].

The first step in the computation of the MFCC is calculating the discrete STFT $X(n,\omega_k)$ for short windowed frames of the audio signal

$$X(n,\omega_k)=\sum_{m=-\infty}^{\infty} x(m)\cdot w(m-n)\cdot e^{-jm\omega_k} \tag{9}$$

where $\omega_k = (2\pi/N)k$ and N is the length of the discrete Fourier transform (DFT).

The magnitude of the STFT is multiplied by a series of weighting bandpass filters, whose frequency response $V_i(\omega_k)$ has finite support and is shaped triangularly around the center frequency. The filter bank contains a total of 55 filters, spanning the frequencies from 66.6 to 16757 Hz. The lower 14 filters have bandwidths of 133.3 Hz each and are spaced linearly with 66.6 Hz between center frequencies. The upper 41 filters are spaced logarithmically, with a factor of 1.071 between center frequencies, each with a bandwidth of 0.1376 of the corresponding center frequency [45].

The frequency response of the filter bank has been shown to resemble that of the auditory critical band filters [41].

Next, the spectral energy is calculated for each filter i

$$E(n,i)=\frac{1}{S_i}\sum_{k=L_i}^{U_i}\left(|X(n,\omega_k)|\cdot V_i(\omega_k)\right)^2 \tag{10}$$

where L_i and U_i are the lower and upper indices of the frequencies contained in the support of the filter, $V_i(\omega_k)$ is the magnitude of the ith filter at frequency ω_k, and S_i is a normalization factor to compensate the variable bandwidth of the filters

$$S_i=\sum_{k=L_i}^{U_i}\left(V_i(\omega_k)\right)^2. \tag{11}$$

Finally, the discrete cosine transform is applied to obtain the MFCC

$$\mathrm{MFCC}(n,l)=\frac{1}{N_f}\sum_{i=0}^{N_f-1}\log\left(E(n,i)\right)\cdot\cos\left(\frac{2\cdot\pi}{N_f}\left(i+\frac{1}{2}\right)\cdot l\right) \tag{12}$$

for which N_f is the number of filters in the Mel filter bank.

B. Mel Cepstrum Nonstationarity Measure

The cepstral nonstationarity measure, used in this study to detect PSTs, is defined as the 2-norm of the MFCC difference between consecutive analysis points

$$D(n)=\sqrt{\sum_{l=0}^{N_L-1}\left|\mathrm{MFCC}(n\cdot J,l)-\mathrm{MFCC}((n+1)\cdot J,l)\right|^2} \tag{13}$$

where N_L is the number of MFC coefficients, $n\cdot J$ is the center of the nth frame, and J is the hop size, typically 3 ms.

Other distance functions, based on the average distance or the geometric mean, were tested but yielded inferior results. We choose $N_L < N_f$ (typically $N_f = 55$, $N_L = 10$), since it is known that the lower coefficients contain the important characteristics of the spectrum and are less sensitive to the analysis conditions [44].

C. Threshold Function for the Mel Cepstrum Nonstationarity Measure

We define a threshold function $Th_D(n)$, so that segments where $D(n) > Th_D(n)$ are considered PSTs. In order to avoid possible tempo distortions due to copying excessive segments without OLA, the threshold (Fig. 6) is set (both globally and locally) so that no more than a predefined percentage of the original signal is copied intact.

The resolution of the cepstrum-based PST detection is equal to the hop size J, which is typically 3 ms. However, because the OLA mechanism considers complete half-windowed frames in

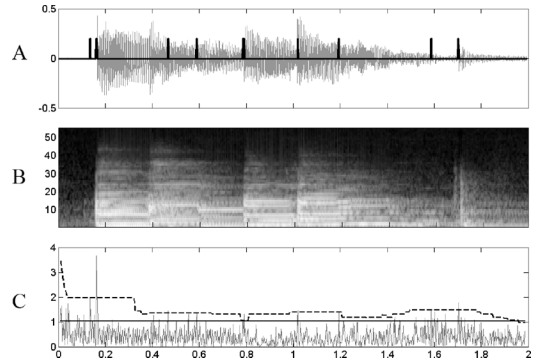


Fig. 6. Selection of "untouched" sections using local and global thresholds. (A) The original signal (gray) and the sections chosen for intact replication (dark). (B) Mel scale spectrum over time. (C) The Mel cepstrum nonstationarity $D(n \cdot J)$ (gray line), local threshold (dashed) and global threshold (dark).

each step, the duration of the segment that is excluded from time-scaling whenever a PST is detected is typically much longer than the PST itself. For example, if $S_1 = 20$ ms and $S_2 = 40$ ms, then the common support of a single overlap is about 60 ms, and for every PST (even if it is as short as a few milliseconds) a whole section of 60 ms might be copied intact. Therefore, if $D(n)$ is used to compute the threshold, the percentage of untouched segments in practice will be much higher than intended. To avoid this, the threshold is computed with respect to an adjusted measure $D_{ol}(n)$, which is defined for every point as the maximum value of the nonstationarity measure in a neighborhood of length T_{ol}

$$D_{ol}(n) = \max\left\{ D(n+i) \,\middle|\, 0 \le i \le \left\lceil \frac{T_{ol}}{J} \right\rceil \right\} \tag{14}$$

where T_{ol} is the common support of two half-windowed frames intended for OLA operation (assuming that the average value of Δ_k is 0),

$$T_{ol} = \begin{cases} 2 \cdot S_2 - S_1 & S_1 < S_2 \\ S_1 & S_1 \ge S_2 \end{cases} \tag{15}$$

(see also Fig. 1).

The threshold $Th_D(n)$ combines a local time-varying threshold $Th_{DL}(n)$ and a global threshold Th_{DG}. The local threshold is set so that only α_L fraction of the signal in a local neighborhood of duration T_{ln} is above it, where T_{ln} is chosen to include about three untouched sections ($T_{ln} = 3 \cdot T_{ol}/\alpha_L$). Setting T_{ln} to include less than two untouched sections may yield a threshold that is too high, for example when a series of attacks with increasing nonstationarities is present.

Determining the value of α_L involves a certain tradeoff. If it is too low, only a few transients will be detected, and many others may be time-scaled, leading to audio degradation. If it is too high, a large portion of the signal will be copied without scaling, causing tempo distortions. The reader should refer to Section IV for further discussion.

Denote by $p(S, \alpha)$ the function that returns the value of α percentile for a given set S, and let

$$S(n) = \left\{ D_{ol}(n+i) \;\middle|\; \left\lceil -\frac{T_{ln}}{2J} \right\rceil \le i \le \left\lceil \frac{T_{ln}}{2J} \right\rceil \right\} \tag{16}$$

be the set of values attained by D_{ol} in the local neighborhood. Then

$$Th_{DL}(n) = p(S(n), (1 - \alpha_L)). \tag{17}$$

The local threshold is intended to select the most important sections for intact replication within a local interval. In order to prevent intact replication in long stationary segments, where it is not necessary, a global threshold is defined in a similar way, where the values of D_{ol} across the entire signal are considered

$$Th_{DG} = p\left(\left\{ D_{ol}(n) \,|\, 1 \le n \le \left\lceil \frac{X_{\mathrm{Len}}}{J} \right\rceil \right\}, (1 - \alpha_G) \right) \tag{18}$$

where typically $\alpha_G = 3 \cdot \alpha_L$ to obtain a global threshold that is more permissive than the local threshold.

The final value of the threshold is chosen as the maximum of the two

$$Th_D(n) = \max\{Th_{DL}(n), Th_{DG}\}. \tag{19}$$

In step k, the signal is considered a PST if one of the half-windowed frames, intended for overlap, contains samples for which $D(n) > Th_D(n)$.

D. Normalized Correlation Criterion for Selection of Intact Sections

The normalized cross-correlation, computed as part of the WSOLA process (see Section II-A), can be used as an additional measure for detection of significant transients. Recall that for each pair of overlapped half-windowed frames, Δ_k is selected according to the maximal normalized correlation between the frames $C(k, \Delta_k)$. High correlation indicates that the frames are quite similar, and the quality of the resulting time-scaled signal will be high.

A mechanism for controlling the rate of untouched sections, similar to the one described in the previous section, is also used here. Two thresholds are defined: the local correlation threshold $Th_{CL}(k)$ and the global threshold Th_{CG}, calculated as above, so that only a certain percentage of the frames are below the threshold (α_{CL} locally and α_{CG} globally).

The final threshold function for normalized correlation is

$$Th_C(k) = \min\{Th_{CG}, Th_{CL}(k)\}. \tag{20}$$

In step k, the signal is considered to contain a PST if $C(k, \Delta_k) < Th_C(k)$.

The advantages of using the cross-correlation are that no additional computations are required, and that it is closely tied to the relation between the input and the constructed output signal. On the other hand, unlike the MFCC-based nonstationarity measure, it does not take the properties of the auditory system into account and tends to ignore high frequencies in signals where the low frequencies are dominant. As such, the cross-correlation is used only as a secondary detection measure, to detect transients that were missed by the MFCC-based measure. Accordingly, the thresholds are set more conservatively: $\alpha_{CL} = 0.2 \cdot \alpha_L$ and $\alpha_{CG} = 0.6 \cdot \alpha_L$, to detect only sections with very low correlation values.

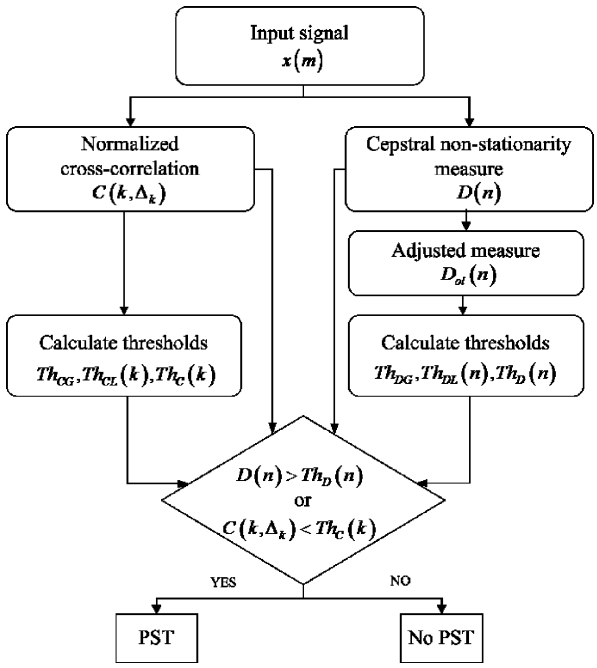


Fig. 7. Schematic description of the PST detection algorithm, using Mel cepstrum nonstationarity and normalized cross-correlation.

A section is considered to contain a transient if it is detected either by the cepstral nonstationarity measure or by the normalized cross-correlation.

A block diagram of the PST detection process is shown in Fig. 7.

IV. Evaluation and Comparison With Other Algorithms

The algorithm was evaluated through three subjective listening tests. In the first test, the listeners were asked to compare the quality of time-scaled signals produced by the present algorithm with corresponding signals, time-scaled using uniform WSOLA, with the same time-scale factor and window length. Seven listeners, all professional sound technicians or amateur musicians, participated in the test, which was conducted using Yamaha RH-40-M professional headphones on signals sampled at 44 100 Hz. Each session included five comparisons of tracks, chosen to cover a wide range of instruments and conditions. Most of the tracks in this test were multipitch. The subjects listened to the original music section, followed by the time-scaled versions, in random order, without being informed which track corresponded to which algorithm. The listeners were guided to indicate their preferred version according to the quality and the naturalness compared to the original section.

The results are summarized in Table I. It can be seen that most of the listeners preferred the present algorithm to WSOLA. Examples of the operation of the algorithm versus WSOLA can be found at http://spl.telhai.ac.il/speech/.

The parameters of the algorithm were $S_1 = 20$ ms, and $J = 3$ ms. It was found empirically that a convenient default value for setting the local threshold is $\alpha_L = 0.1$. Although this value yields satisfactory results, it was fine-tuned to achieve optimal results for each signal separately. Note that the other parameters

TABLE I
Listening Tests and Results

Track Name	Original Length [sec]	Stretch Ratio	Non-Scaled Ratio	Present Algorithm Preferred	Cannot Choose	WSOLA Preferred
"Pipa"	8	x 2	11.6 %	7	0	0
"Dylan"	7	x 2	6.9 %	6	1	0
"Drums"	13	x 1.7	11.0 %	6	0	1
"Janis"	6	x 2	12.7 %	6	0	1
"Piano"	9	x 2	8.2 %	3	3	1

The fourth column (nonscaled ratio) describes the percentage of transients that were not time-scaled by the algorithm. "Pipa" – 4-stringed guitar-like plucked instrument; "Dylan" – "Blowin' in the Wind," sung by Bob Dylan alone with guitar and harmonica; "Drums" – "Ghost Train," played by Counting Crows, mostly on drums and bass guitar; "Janis" – "Trouble in Mind," sung by Janis Joplin, electric guitar; "Piano" – Track of electric piano.

(α_G, α_{CL}, and α_{CG}) are derived directly from α_L and need not be fine-tuned.

The algorithm was validated both by comparing the PST locations detected by the algorithm to a manual demarcation, and by carefully listening to the time-scaled signal. The procedure was repeated by three users for consistency. The values of the parameters were selected after testing the algorithm with α_L in the range [0.08,0.2], and S_1 between 12 and 25 ms. The setting of S_1 is a tradeoff between keeping high resolution to achieve local similarity between corresponding neighborhoods of the input and the output (small values) and ensuring the phase relations during the OLA procedure (for which large values are preferred, especially in low-pitch signals).

Additionally, a MUSHRA-like double blind test using a hidden reference [46] was carried out to compare the quality of the presented algorithm with WSOLA and the phase vocoder. In this test, three music tracks were compared, subjected to stretching by a factor of 2 ("Pipa" and "Janis") and 1.7 ("Drums"). Nineteen listeners, most of them either musicians or experienced in the field of audio evaluation, participated in this test. Each listener could hear the reference (original) section and the modified sections multiple times, switching between them at will, and was asked to adjust sliders to match their subjective evaluation score using a scale of 0–100, where qualitative ranks ("bad," "poor," "fair," "good," and "excellent"), were used for guidance. The results of the tests are summarized in Fig. 8. As can be seen, the presented algorithm received the highest average score, significantly higher than the other algorithms.

The third test compared the quality of the proposed algorithm with two modern algorithms, developed by Bonada [27], and Roebel [29] (see Section I). Two sessions were conducted: the first using a stretching factor of 1.3 (14 listeners), and the second with a stretching factor of 2.0 (12 listeners). The results are summarized in Fig. 9. The scores of the proposed algorithm in this test were somewhat lower but still comparable with these algorithms. While the advantages of the latter algorithms were more noticeable when polyphonic music was tested, in the case of a monophonic example (such as "Pipa ") the differences are quite small.

This is most probably due to the quasi-periodicity assumption of the WSOLA algorithm (see Section I), which is more applicable to monophonic harmonic sounds than to polyphonic

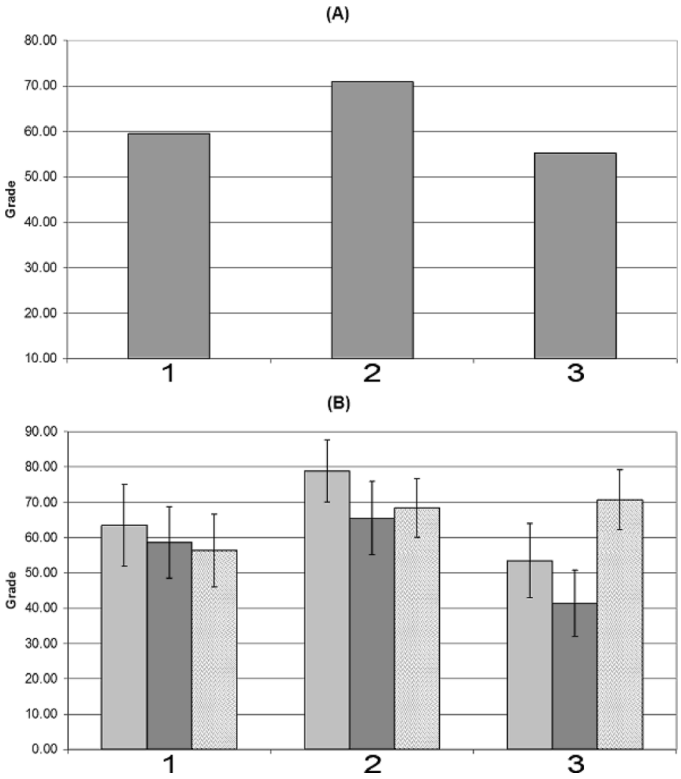


Fig. 8. MUSHRA test results for evaluation of the subjective quality of each of the algorithms. 1—WSOLA; 2—the proposed algorithm; 3—phase vocoder. (A) Average score of all tracks. (B) MUSHRA test results for each track separately. The vertical lines denote a 95% confidence interval. The left column in each of the methods represents the score of the "Pipa " track, the middle represents the average score of the "Drums," and the right is for the average score of the "Janis" track.

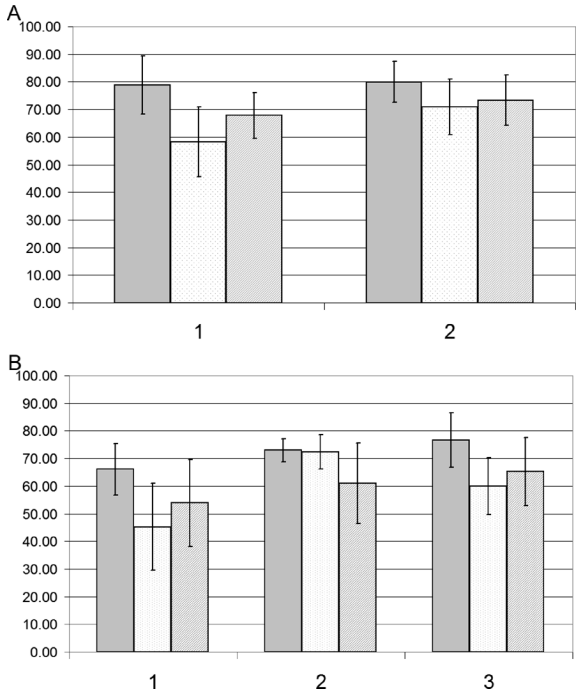


Fig. 9. MUSHRA test results for evaluation of the subjective quality of each of the algorithms. 1—the proposed algorithm; 2—Bonada's TSM, 3—Roebel's phase vocoder. The lines denote a 95% confidence interval. (A) Using a stretching factor of 1.3. (B) Using a stretching factor of 2.0. The left column in each of the methods represents the score of the "Pipa" track, the middle represents the average score of the "Dylan" track, and the right is for the average score of the "Janis" track.

or nonharmonic music. Several tests conducted with a drum beat sequence, which contains a nonharmonic noise-like sound source, further supported this conclusion: Ten listeners participated in this experiment, using a stretching factor of 2.0. Our algorithm achieved an average score of 57%, versus 61% and 82% for the algorithms of Bonada and Roebel, respectively.

Thus, our results suggest that the proposed algorithm, while being relatively simple, achieves a high-quality modified audio, improving upon the performance of most OLA-based algorithms.

V. Conclusion

In this paper, we have presented a method for TSM of music signals, which consists of a WSOLA-based algorithm for time-scaling and a mechanism for detecting PSTs. The transients may be subject to degradation when TSM is performed. In order to prevent this, the PSTs are left intact, while other sections are time-scaled according to the required modification factor. The proposed algorithm yields high quality time-scaled music signals, both stretched and compressed. Three listening tests were carried out, which showed that the algorithm outperforms similar algorithms, such as the regular WSOLA, and is comparable to more complicated, parametric algorithms. This suggests that the differential manipulation of transients is crucial for achieving high-quality time-scaled music with the overlap-and-add approach.

Another fundamental issue with OLA-based TSM algorithms is that they are less adequate to polyphonic music. While for monophonic signals, especially harmonic sources, the necessary synchronization between the overlapping frames can be achieved to ensure proper phase relations during the OLA procedure, this may not be the case with polyphonic signals, especially when there is no strong quasi-periodic element. This explains some of the results in Section IV.

Adapting the algorithm to polyphonic sounds could be done using a subband framework [37], [47]. With this approach, the signal is decomposed into a set of simpler waveforms and each subband signal is time-scaled separately, using a different set of parameters. It would be interesting to study whether an integration of the principles, namely an implementation of our algorithm in a subband framework, with differential treatment of transients, will yield better results compared to each of the algorithms separately.

Acknowledgment

The authors would like to thank Prof. D. Malah, the Head of SIPL, Technion, Haifa, Israel, for reading the manuscript and for his valuable comments, and Y. Yakir for valuable technical support. They would also like to thank the listeners who participated in the tests. Special thanks to I. Neoran, Director of R&D of Waves Audio, for his assistance in the subjective listening tests. They would like to thank Dr. J. Bonada from MTG at Pompeu Fabra University, Barcelona, Spain, and Dr. A. Roebel from IRCAM, Paris, France, for their generous help in providing

the authors the results of their algorithms as part of the evaluation tests. Special thanks to Mr. D. Ruinskiy for his helpful comments and for valuable assistance in revising the paper. The authors are grateful to the anonymous reviewers for valuable comments and suggestions.

REFERENCES

[1] D. W. Griffin and J. S. Lim, "Signal estimation from modified short-time Fourier transform," *IEEE Trans. Audio, Speech, Signal Process.*, vol. ASSP-32, no. 2, pp. 236–243, Apr. 1984.

[2] J. Laroche and M. Dolson, "Improved phase vocoder time-scale modification of audio," *IEEE Trans. Speech Audio Process.*, vol. 7, no. 3, pp. 323–332, May 1999.

[3] E. Moulines and J. Laroche, "Non-parametric techniques for pitch-scale and time-scale modification of speech," *Speech Commun.*, vol. 16, no. 2, pp. 175–206, 1995.

[4] M. R. Portnoff, "Time scale modification of speech based on short-time Fourier analysis," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, no. 3, pp. 374–390, Jun. 1981.

[5] G. Pallone, "Dilatation et transposition sous contraintes perceptives des signaux audio: Application au transfert cinéma-vidéo," Ph.D. dissertation, Laboratoire de Mécanique et d'Acoustique, Université de la Méditerranée – Aix-Marseille II, Aix-Marseille, France, 2003.

[6] X. Amatriain, J. Bonada, A. Loscos, J. L. Arcos, and V. Verfaille, "Content-based transformations," *J. New Music Res.*, vol. 32, no. 1, pp. 95–114, 2003.

[7] D. Arfib and V. Verfaille, "Driving pitch-shifting and time-scaling algorithms with adaptive and gestural techniques," in *Proc. Int. Conf. Digital Audio Effects (DAFX)*, London, U.K., 2003, pp. 106–111.

[8] M. Demol, K. Struyve, W. Verhelst, H. Paulussen, P. Desmet, and P. Verhoeve, "Efficient non-uniform time-scaling of speech with WSOLA for CALL applications," in *Proc. InSTIL/ICALL 2004 Symp. Comput. Assisted Learn., NLP Speech Technol. Adv. Lang. Learn. Syst.*, Venice, Italy, 2004, paper 007.

[9] Y. Nejime, T. Aritsuka, T. Imamura, T. Ifukube, and J. Matsushima, "A portable digital speech-rate converter for hearing impairment," *IEEE Trans. Rehabil. Eng.*, vol. 4, no. 2, pp. 73–83, Jun. 1996.

[10] D. Malah, "Time-domain algorithms for harmonic bandwidth reduction and time scaling of speech signals," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, no. 2, pp. 121–133, Apr. 1979.

[11] J. Laroche, "Autocorrelation method for high quality time/pitch scaling," in *Proc. IEEE Workshop Applicat. Signal Process. Audio Acoust.*, New Paltz, NY, 1993, pp. 131–134.

[12] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," *IEEE Trans. Speech Audio Process.*, vol. 6, no. 2, pp. 131–142, Mar. 1998.

[13] B. S. Atal and S. L. Hanauer, "Speech analysis and synthesis by linear prediction of the speech wave," *J. Acoust. Soc. Amer.*, vol. 50, pp. 637–655, 1971.

[14] J. L. Flanagan and R. M. Golden, "Phase vocoder," *Bell Syst. Tech. J.*, vol. 45, no. 9, pp. 1493–1509, 1966.

[15] E. Moulines and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech Commun.*, vol. 9, no. 5-6, pp. 453–467, 1990.

[16] S. Roucos and A. M. Wilgus, "High quality time-scale modification for speech," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Tampa, FL, 1985, pp. 493–496.

[17] W. Verhelst, "Overlap-add methods for time-scaling of speech," *Speech Commun.*, vol. 30, no. 4, pp. 207–221, 2000.

[18] W. Verhelst and M. Roelands, "An overlap-add technique based on waveform similarity (WSOLA) for high quality time-scale modification of speech," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Minneapolis, MN, 1993, pp. 554–557.

[19] S. Furui, "On the role of spectral transition for speech perception," *J. Acoust. Soc. Amer.*, vol. 80, no. 4, pp. 1016–1025, 1986.

[20] S. Handel, *Listening*. Cambridge, MA: MIT Press, 1993.

[21] M. Covell, M. Withgott, and M. Slaney, "Mach1: Nonuniform time-scale modification of speech," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Seattle, WA, 1998, pp. 349–352.

[22] M. Demol, W. Verhelst, K. Struyve, and P. Verhoeve, "Efficient non-uniform time-scaling of speech with WSOLA," in *Proc. 10th Int. Conf. Speech Comput. (SPECOM)*, Patras, Greece, 2005, pp. 163–166.

[23] D. Kapilow, Y. Stylianou, and J. Schroeder, "Detection of non-stationarities in speech signals and its application in time-scaling," in *Proc. 6th Eur. Conf. Speech Commun. Technol. (Eurospeech)*, Budapest, Hungary, 1999, pp. 2307–2310.

[24] A. Klapuri, "Sound onset detection by applying psychoacoustic knowledge," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Phoenix, AZ, 1999, pp. 3089–3092.

[25] P. Masri and A. Bateman, "Improved modelling of attack transients in music analysis–resynthesis," in *Proc. Int. Comput. Music Conf. (ICMC)*, Hong Kong, 1996, pp. 100–103.

[26] C. Duxbury, M. E. Davies, and M. B. Sandler, "Separation of transient information in musical audio using multiresolution analysis techniques," in *Proc. Int. Conf. Digital Audio Effects (DAFX) Limerick*, U.K., 2001, pp. 1–4.

[27] J. Bonada, "Automatic technique in frequency domain for near-lossless time-scale modification of audio," in *Proc. Int. Comput. Music Conf. (ICMC)*, Berlin, Germany, 2000, pp. 396–399.

[28] C. Duxbury, M. E. Davies, and M. Sandler, "Improved time-scaling of musical audio using phase locking at transients," in *Proc. Audio Eng. Soc. 112th Conv.*, Munich, Germany, 2002, paper 5530.

[29] A. Roebel, "A new approach to transient processing in the phase vocoder," in *Proc. Int. Conf. Digital Audio Effects (DAFX)*, London, U.K., 2003, pp. 344–349.

[30] X. Rodet and F. Jaillet, "Detection and modeling of fast attack transients," in *Proc. Int. Comput. Music Conf. (ICMC)*, Havana, Cuba, 2001, pp. 30–33.

[31] S. Hainsworth, M. Macleod, and P. Wolfe, "Analysis of reassigned spectrograms for musical transcription," in *Proc. IEEE Workshop Applicat. Signal Process. Audio Acoust.*, New Paltz, NY, 2001, pp. 23–26.

[32] G. Peeters and X. Rodet, "SINOLA: A new method for analysis/synthesis using spectrum distorsion, phase and reassigned spectrum," in *Proc. Int. Comput. Music Conf. (ICMC)*, Beijing, China, 1999, pp. 153–156.

[33] M. S. Puckette, "Phase-locked vocoder," in *Proc. IEEE Workshop Applicat. Signal Process. Audio Acoust.*, New Paltz, NY, 1995, pp. 222–225.

[34] M. Dolson, "The phase vocoder: A tutorial," *Comput. Music J.*, vol. 10, no. 4, pp. 14–26, 1986.

[35] J. A. Moorer, "The use of the phase vocoder in computer music applications," *J. Audio Eng. Soc.*, vol. 26, no. 1, pp. 42–45, 1978.

[36] S. Lee, H. D. Kim, and H. S. Kim, "Variable time-scale modification of speech using transient information," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Munich, Germany, 1997, pp. 1319–1322.

[37] D. Dorran and B. Lawlor, "An efficient time-scale modification algorithm for use within a subband implementation," in *Proc. Int. Conf. Digital Audio Effects (DAFX)*, London, U.K., 2003, pp. 339–343.

[38] G. Spleesters, W. Verhelst, and A. Wahl, "On the application of automatic waveform editing for time warping digital and analog recordings," in *Audio Eng. Soc. 96th Conv.*, Amsterdam, The Netherlands, 1994, preprint 3843.

[39] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-28, no. 4, pp. 357–366, Aug. 1980.

[40] S. S. Stevens and J. Volkman, "The relation of pitch to frequency: A revised scale," *Amer. J. Psychol.*, vol. 53, pp. 329–353, 1940.

[41] T. F. Quatieri, *Discrete-Time Speech Signal Processing*. Upper Saddle River, NJ: Prentice-Hall, 2001.

[42] J. P. Campbell, "Speaker recognition: A tutorial," *Proc. IEEE*, vol. 85, no. 9, pp. 1437–1462, Sep. 1997.

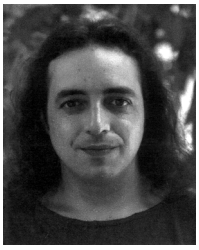
[43] R. Mammone, X. Zhang, and R. Ramachandran, "Robust speaker recognition: A feature-based approach," *IEEE Signal Process. Mag.*, vol. 13, no. 5, pp. 58–71, Sep. 1996.

[44] L. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[45] M. Slaney, "Auditory Toolbox," Interval Res. Corp., Palo Alto, CA, 1998, #1998-010.

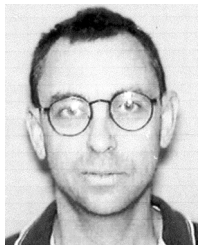
[46] G. Stoll and F. Kozamernik, "EBU listening tests on internet audio codecs," *EBU Technical Rev.*, no. 283, pp. 20–33, 2000.

[47] D. Dorran and B. Lawlor, "Time-scale modification of music using a synchronized subband/time domain approach," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Montreal, QC, Canada, 2004, pp. 225–228.



Shahaf Grofit (S'07) received the B.A. degree in computer science, specializing in signal processing, from Tel-Hai Academic College, Upper Galilee, Israel. He is currently pursuing the M.Sc. degree in computer science at Tel-Aviv University, Tel-Aviv, Israel. His thesis focuses on nonlinear stable connectivity between brain regions.

His research interests include signal processing, usually of audio, and system identification.



Yizhar Lavner (M'01) received the Ph.D. degree from The Technion, Israel Institute of Technology, Haifa, in 1997.

He has been with the Computer Science Department, Tel-Hai Academic College, Upper Galilee, Israel, since 1997, where he is now a Senior Lecturer. He also has been teaching in the Signal and Image Processing Lab (SIPL), Electrical Engineering Faculty, The Technion, since 1998. His research interests include audio and speech signal processing, voice analysis and perception, and genomic signal processing.