Recognition of Isolated Musical Patterns Using Context Dependent Dynamic Time Warping

Aggelos Pikrakis, *Associate Member, IEEE*, Sergios Theodoridis, *Senior Member, IEEE*, and Dimitris Kamarotos

***Abstract*—Automatic recognition of musical patterns plays a crucial part in Musicological and Ethnomusicological research and can become an indispensable tool for the search and comparison of music extracts within a large multimedia database. This paper presents an efficient method for recognizing isolated musical patterns in a monophonic environment, using a novel extension of Dynamic Time Warping, which we call Context Dependent Dynamic Time Warping. Each pattern, to be recognized, is converted into a sequence of frequency jumps by means of a fundamental frequency tracking algorithm, followed by a quantizer. The resulting sequence of frequency jumps is presented to the input of the recognizer. The main characteristic of Context Dependent Dynamic Time Warping is that it exploits the correlation exhibited among adjacent frequency jumps of the feature sequence. The methodology has been tested in the context of Greek Traditional Music, which exhibits certain characteristics that make the classification task harder, when compared with Western musical tradition. A recognition rate higher than 95% was achieved.**

***Index Terms*—Dynamic time warping, signal processing for music.**

I. Introduction

DIGITAL sound processing tools offer new possibilities to the analysis of musical structures, the modeling of the acoustic characteristics of an instrument and the musical pattern comparison and recognition. The earliest and most well known survey of digital signal processing techniques for the production and processing of musical sounds was authored by [1] in 1976. Today, the computational efficiency of computers permits the research community to deal with tasks that were unrealistic to face before. Such an important task of great interest to the musicologist is the semi-automated search of specific sound patterns within a large number of stored sound files. These musical patterns have been shaped and categorized through practice and experience in many musical traditions.

This paper proposes a scheme for the recognition of such predefined musical patterns in a monophonic environment in the context of Greek Traditional music. It is assumed that the patterns to be recognized have been isolated from their context by means of a segmentation process, thus the term "isolated musical patterns." To this end a manual segmentation procedure was adopted. The term monophonic refers to a *single nonpolyphonic instrument, the clarinet, recorded under laboratory conditions with an ambient noise of less than 5 dB*. The reported research focuses on Greek Traditional music. The musical system of Greek Traditional music and the techniques of instrument players give the resulting sound material a radically different structure when compared with that of the western equal-tempered intervalic system (system of musical scales). Some major differences are:

- existence of transitional patterns between notes;
- use of larger, formalized transitory (melody) patterns, ranging typically from 2 s up to 30 s, as a main element of the musical structure and not as an ornamental one, as it is the case in the western musical tradition;
- intervalic system that contains many musical intervals, which are smaller compared to the intervals used in the "well-tempered" western tradition. For the signals under study, the smallest musical interval encountered was equal to one quarter of the tone. In many cases, musical intervals that were odd multiples of one quarter of the tone were also observed.

From a large number of types of transitory musical patterns, encountered in practice in different instrumental styles, we have selected the twelve most typical cases. The choice of the types of patterns was suggested by musicologists on the basis of a) their common use in practice and b) their respective time elasticity. The time elasticity of a musical pattern refers to the phenomenon of stretching its total length, up to five times in some cases, while retaining its musical function.

The recognition scheme that we propose consists of two stages. In a first stage, a feature generation algorithm converts the unknown musical pattern into a sequence of frequency jumps. The smallest allowed frequency jump is equal to one quarter-tone. At the heart of this stage lies a fundamental frequency tracking algorithm which generates a sequence of fundamental frequencies from the unknown musical pattern. A number of well known time-domain and frequency-domain algorithms were considered and tested [2]–[8]. In addition, a new low complexity frequency-domain algorithm was developed. This algorithm can be considered as a modification of Schroeder's histogram [2] and its performance is slightly inferior to that of Brown's autocorrelation method [5], yet much more efficient from a computational point of view. Each extracted fundamental frequency is subsequently mapped to a positive integer, equal to the distance (measured in quarter-tone units) between the fundamental frequency and the lowest possible frequency that the instrument can produce.

In the second stage, Context Dependent Dynamic Time Warping (CDDTW) is employed in order to match the previously extracted feature sequence to a set of twelve reference

Manuscript received July 20, 2000; revised August 29, 2002. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Bryan George.

The authors are with the Department of Informatics and Telecommunications, University of Athens, Athens, Greece (e-mail: stheodor@di.uoa.gr).

Digital Object Identifier 10.1109/TSA.2003.811533

sequences (one reference sequence per musical type). The unknown pattern is determined based on the lowest matching cost. We propose CDDTW as a novel extension of the standard Dynamic Time Warping (DTW) methodology [9]–[14]. Its rationale possesses certain similarities to a variation of Hidden Markov Models known as segment modeling. Although standard DTW schemes assume that each feature in the resulting sequence is uncorrelated with its neighboring ones (i.e., its context), CDDTW permits flexible grouping of neighboring features (i.e., forming feature segments) in order to exploit possible underlying mutual dependence.

The problem we are dealing with can be considered to be equivalent to the isolated word recognition task, that has been addressed by the speech processing community. To our knowledge, this is the first time that the CDDTW methodology is proposed and applied to the recognition of isolated musical patterns in the time domain. Previous work by the authors employed standard DTW schemes [15] which present certain limitations as will be discussed in Section III. An approach based on Hidden Markov Models has been presented in [16]. A standard DTW scheme was also used in [17], but for signals available in MIDI format, which has severe limitations for the majority of real world signals and in particular for the case of Greek Traditional musical patterns. Most previously published literature related to music sound recognition has focused on MIDI representation, e.g. [18], [19].

Section II presents the aforementioned feature generation procedure, along with the newly developed fundamental frequency tracking algorithm and a list of the algorithms that were studied in the context of Greek traditional music. Section III presents CDDTW and compares it with standard DTW schemes. Section IV gives details of the application of our method to patterns from the Greek Traditional music. Conclusions and future work are presented in Section V.

II. Feature Generation

A. Fundamental Frequency tracking

During the first stage of feature generation, a sequence of fundamental frequencies from the musical pattern to be recognized are extracted. At this point, we must emphasize that in the literature, the terms "fundamental frequency" and "pitch" are often used interchangeably, although their values do not always coincide. The perception of pitch is a psychoacoustical phenomenon, whereas the fundamental frequency is a quantity that can be calculated algorithmically for periodic or quasiperiodic signals.

To accomplish the task of fundamental frequency tracking we experimented with several frequency domain and time domain methods and also developed a new frequency-domain algorithm of low complexity. The methods which were tested are

1) Frequency-domain approaches: Schroeder's histogram [2], Schroeder's Harmonic Product Spectrum [2], Piszscalski's method [3] and Brown's pattern recognition method based on the properties of a constant-Q transform [4].

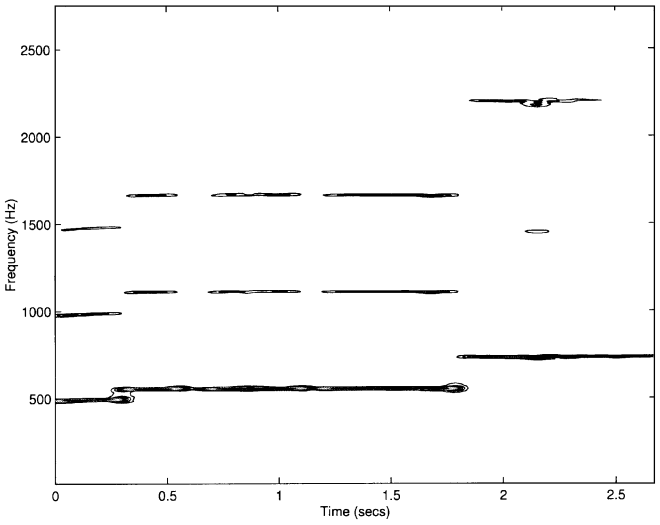


Fig. 1. Contour plot of the spectrogram of a musical pattern of type I.

2) Time-domain methods: Cooper and Kia's method [7] and Brown's narrowed autocorrelation method [5]. Also Tolonen's method was used [8].

After extensive experimentation with all the above algorithms, we concluded that Brown's narrowed autocorrelation method [5] and Tolonen's multipitch analysis model [8] gave the best results with respect to accuracy and frequency doubling, provided all required parameters were rightly tuned.

Further to the above, we developed a simple, low-complexity frequency-domain algorithm that can be considered as a modification of Schroeder's histogram. Its drawback is that it fails to cope with the problem of missing fundamentals, but this is not a crucial issue for the signals of our study. The algorithm is based on the principle that the frequency content of a musical pattern is split into frequency components, i.e., a fundamental frequency component and its harmonics. This is illustrated in Fig. 1 for a musical pattern. Ideally, the fundamental frequency should always be identical to the largest Fourier peak. It would then suffice to choose the maximum Fourier peak from each frame as the respective feature. However, in many cases (in 23% of the frames of the signals that we studied) the largest Fourier peak is located in one of the harmonic frequency components. The fundamental frequency tracking algorithm should always follow the lowest distinguishable frequency component irrespective of where the largest Fourier peak lies.

Based on the above idea, our algorithm works as follows: A moving window Fourier Transform is performed on the musical pattern. For each window frame, the frequencies corresponding to the K (a preselected parameter, $K = 5$ for our experiments) most dominant peaks are selected as feature candidates. In turn, each one of the above frequencies is examined, starting from the lowest frequency, until a frequency is detected, which corresponds to a peak whose amplitude is higher than a (preselected) percentage threshold T_p, relative to the most dominant peak. The frequency that satisfies the above criterion is chosen as the respective feature for the specific frame. Furthermore, if the highest Fourier peak of a frame is less than a predefined threshold T_h, the frame is considered to be non periodic and the extracted frequency is set equal to one. For the signals that we

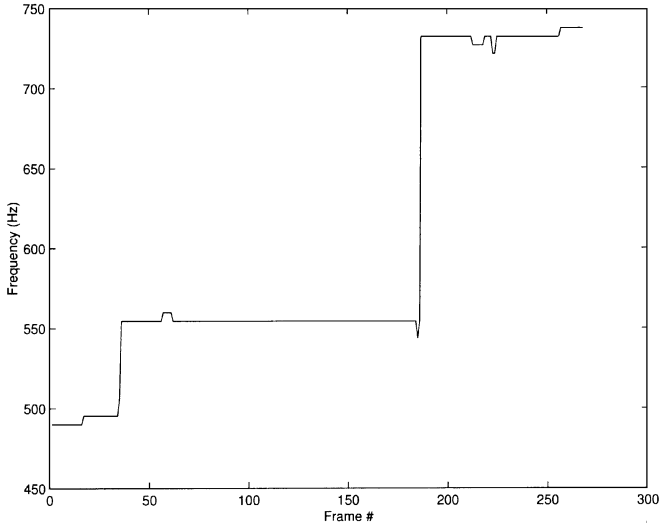


Fig. 2. Fundamental frequency tracking results for the pattern of Fig. 1 using the algorithm that we propose.

studied, only 2.5% of the window frames occurred to be non periodic. Fig. 2 shows the extracted sequence of frequencies corresponding to the pattern of Fig. 1. During this processing, errors are likely to occur in frames where the fundamental frequency is "absent." In such cases, the so called "pitch doubling" phenomenon is observed [20]. Some of the errors can be eliminated at a post-processing step, by setting the pitch period of a frame to be equal to the average of the pitch periods of its two neighboring frames. If the pitch doubling phenomenon spans more than one consecutive frames, then errors are inevitably propagated to the next stage of the recognition scheme. It must be stated that, Brown's autocorrelation method and Tolonen's multipitch analysis model also gave comparable results, however at a significantly higher computational cost.

B. Quantization of the Extracted Frequencies

Let $\mathbf{f} = \{f_i, i = 1 \cdots M\}$, be the generated sequence of fundamental frequencies corresponding to M successive frames of a pattern. The first goal of this stage is to map each f_i to a positive number, say k, equal to the distance (measured in quarter-tone units) of f_i from f_s, where f_s is the lowest frequency that the clarinet can produce (for the signals that we studied $f_s = 146.8$ Hz). Therefore

$$k = round\left(24 \log_2 \frac{f_i}{f_s}\right)$$

where $round(.)$ denotes the roundoff operation. As a result, the sequence of frequencies is mapped to a sequence of positive numbers, $\mathbf{L} = \{l_i, i = 1 \cdots M\}$. The goal of this step is to imitate some aspects of the human auditory system, which is known to analyze an input pattern using a logarithmic frequency scale. An alternative view for this mapping process is to consider it as a quantization step, where each f_i is quantized to a symbol from a finite and discrete alphabet. The alphabet consists of the positive integers in the range of 0 to 80. These range limits denote that the lowest frequency which can be accepted as a fundamental is equal to 146.8 Hz and the highest is $146.8 * 2^{80/24} = 1480$ Hz.

An issue that has to be carefully treated is the fact that instances of the same musical type may have different starting frequencies. According to our quantization process, such cases imply that each instance of the musical type is mapped to a different symbol sequence, although they are all just shifted versions of the same pattern. A solution to this problem is to extract a sequence of frequency jumps from the symbol sequence $\mathbf{L}$. This is achieved by calculating the difference $\mathbf{D}$ of $\mathbf{L}$, i.e.,

$$\mathbf{D} = \{d_{i-1} = l_i - l_{i-1}, i = 2 \cdots M\}.$$

It is obvious that d_i can be positive, negative or zero, depending on whether l_i is greater, less than or equal to l_{i-1}. Since each note, in a musical pattern, is most likely to span more than one consecutive frames, most of the time l_i is equal to l_{i-1}. This is important and it means that $d_i = 0$ for most of the frames (i's). Calculating differences can be alternatively viewed as transforming the sequence of positive numbers to a sequence of symbols (frequency jumps) falling in the range of $-G$ to G, where G corresponds to the maximum allowed frequency jump ($G = 60$ quarter-tones, i.e., 15 tones for the signals that we studied).

III. Context Dependent Dynamic Time Warping

In the sequel, the resulting (from the unknown pattern) sequence $\mathbf{D} = \{d_i, i = 1 \cdots M - 1\}$ is matched against a set of twelve reference patterns (one reference pattern per musical type) using CDDTW. The rest of this section presents, at first, how the reference patterns are chosen. The CDDTW methodology is then introduced starting from a presentation of certain limitations of standard DTW schemes.

A. Choice of the Reference Patterns

The choice of reference patterns is based on the fact that all musical patterns of a specific type can be considered as variations of a theoretically established model. Such models are the result of musicological research in the context of Greek Traditional music and describe the ideal structure that should be present in all patterns of a specific type. Following the notation adopted so far, each model is translated to a reference sequence $R_l = \{0, S_1, 0, S_2, 0, \ldots, S_{R_l}, 0\}$, $l = 1, \ldots, 12$, where $\{S_1, \ldots, S_{R_l}\}$ are positive or negative frequency jumps, multiples of one quarter-tone. It is important to notice that there is only one zero separating successive S_i's. This is because, as we will soon discuss, successive zeros do not contribute to the cost. For example, the reference pattern for musical type II is $R_2 = \{0, -2, 0, -4, 0, -4, 0, 4, 0\}$. Table I presents the representative reference patterns for the twelve musical types that we studied.

B. From DTW to CDDTW

In order to demonstrate the advantages of CDDTW, we first show how patterns of the same musical type may differ from the respective reference model and how such differences affect the performance of standard DTW schemes. Following the feature generation stage, described in the previous section, it is expected

TABLE I
REPRESENTATIVE REFERENCE PATTERNS FOR THE TWELVE MUSICAL TYPES

Model Type	Reference Sequence
1	$\{0, 10, 0, 8, 0\}$
2	$\{0, -2, 0, -4, 0, -4, 0, 4, 0\}$
3	$\{0, 8, 0, 2, 0\}$
4	$\{0, 6, 0, 4, 0\}$
5	$\{0, 14, 0, -2, 0, -12, 0, 6, 0, -6, 0, 14, 0, -2, 0, -12, 0\}$
6	$\{0, -4, 0, -4, 0, -2, 0, -4, 0\}$
7	$\{0, 10, 0, -4, 0, 4, 0, -4, 0, 4, 0, -4, 0, 4, 0, -4, 0, 4, 0, -4, 0, 4, 0, -4, 0, 4, 0, -4, 0, 4, 0\}$
8	$\{0, 14, 0, -14, 0, 14, 0, -14, 0, 14, 0, -14, 0, 14, 0, -14, 0, 14, 0, -14, 0, 14, 0, -14, 0\}$
9	$\{0, 10, 0, -2, 0, -4, 0\}$
10	$\{0, 2, 0, -2, 0, 2, 0, -2, 0, 2, 0, -2, 0, 24, 0\}$
11	$\{0, 4, 0, 2, 0, -2, 0, 2, 0, -2, 0, -4, 0, 4, 0, -4, 0, 4, 0\}$
12	$\{0, -6, 0, -4, 0, 4, 0, -4, 0, 4, 0, -4, 0, 4, 0, -4, 0, 4, 0, -4, 0, 4, 0, -4, 0\}$

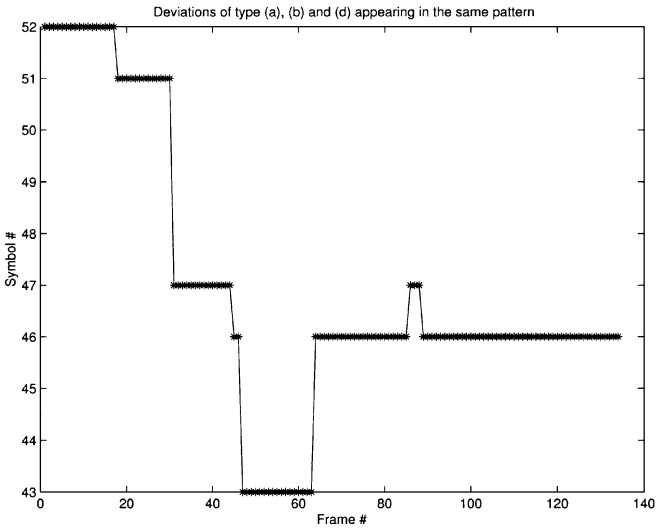


Fig. 3. Symbol sequence L of a musical pattern of type II, prior to calculating differences. Deviations of type a), b), and d) can be observed. If differences are calculated, the resulting feature sequence is $\mathbf{D} = \{\mathbf{0}_{z_1}, -1, \mathbf{0}_{z_2}, -4, \mathbf{0}_{z_3}, -1, \mathbf{0}_{z_4}, -3, \mathbf{0}_{z_5}, 3, \mathbf{0}_{z_6}, 1, \mathbf{0}_{z_7}, -1, \mathbf{0}_{z_8}\}$.

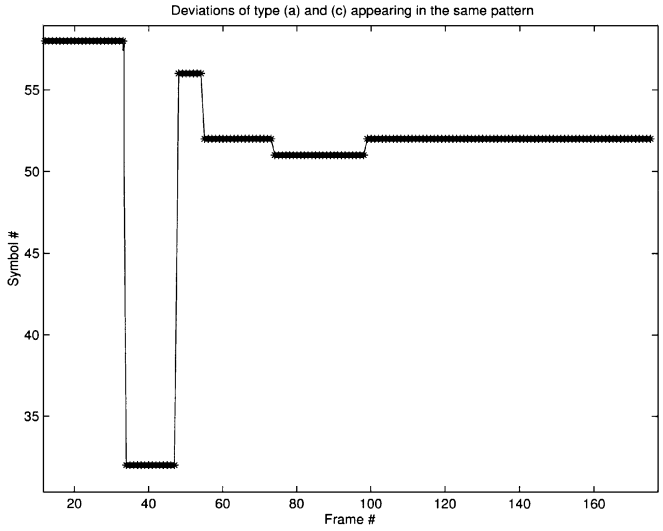


Fig. 4. Symbol sequence L of another musical pattern of type II, prior to calculating differences. Deviations of type a), and c) can be observed. The resulting feature sequence is $D = \{\mathbf{0}_{z_1}, -26, \mathbf{0}_{z_2}, 24, \mathbf{0}_{z_3}, -4, \mathbf{0}_{z_4}, -1, \mathbf{0}_{z_5}, 1, \mathbf{0}_{z_6}\}$. In this case, each deviation of type a) is equal to 3QT's, which is an extreme case. It is also worth noticing that $-26 + 24 = -2$, equal to S_1 of R_2.

that feature sequences, corresponding to patterns of the same musical type, should possess the following structure:

$$\{\mathbf{0}_{z_1}, S_1, \mathbf{0}_{z_2}, S_2, \mathbf{0}_{z_3}, \ldots, \mathbf{0}_{z_{R_l}}, S_{Rl}, \mathbf{0}_{z_{R_l+1}}\}$$

where $\mathbf{0}_{z_k}$ stands for z_k successive zeros. In other words, due to the phenomenon of time elasticity, such feature sequences should, ideally, differ only in the number of successive zero-valued d_i's, separating any two S_i's. However in practice, the following deviations from this ideal situation are often encountered:

a) Some S_i's can be one quarter-tone higher or lower than what one would expect. This is due to variations among instrument players and/or to errors during the feature generation stage (Figs. 3 and 4).
b) Negative or positive jumps, equal to one quarter-tone, *usually encountered in pairs,* are likely to appear in the feature sequence due to errors in the feature generation stage. Such pairs manifest themselves as sub-sequences of d_i's of the form $\{-1, \mathbf{0}_{k_1}, 1, \mathbf{0}_{k_2}\}$ or of the form $\{1, \mathbf{0}_{k_1}, -1, \mathbf{0}_{k_2}\}$ (Fig. 3), in place of the expected sequence $\mathbf{0}_{k_1+k_2+2}$ of $k_1 + k_2 + 2$ zeros.
c) Large pitch estimation errors, generated by the fundamental frequency tracker, (pitch doubling or pitch halving errors spanning more than one consecutive frames), are also likely to appear. Such errors usually manifest themselves as a large negative (positive) frequency jump P_1 followed by a number of zeros and a large positive (negative) jump Q_1 followed by a number of zeros. This is demonstrated in Fig. 4, where $P_1 = -26$, $Q_1 = +24$ and 13 successive zeros separate P_1 from Q_1. The sum of P_1 and Q_1 is equal to -2 which coincides with the "true" symbol S_1 of the reference sequence for patterns of musical type II and is also equal to the frequency jump perceived by the human ear in this case. In general, $|P_1| > 20$, $|Q_1| > 20$ and $P_1 + Q_1$ is equal to the respective S_i (a quarter-tone difference might be observed) or equal to zero. In some more complicated situations,

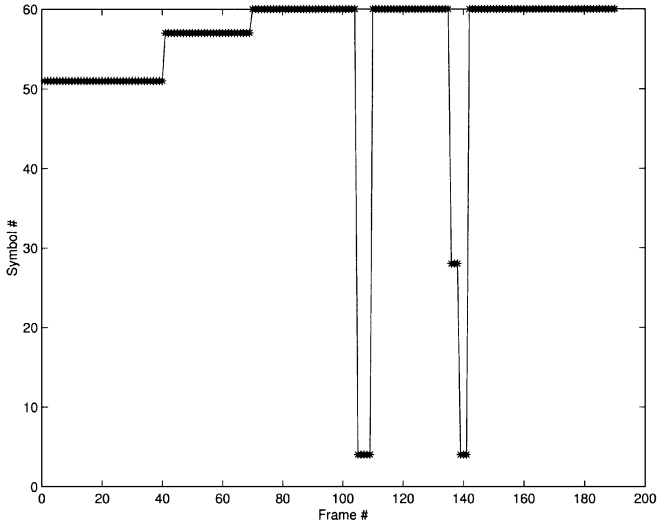


Fig. 5. Symbol sequence of a musical pattern with two cases of large fundamental frequency tracking errors.

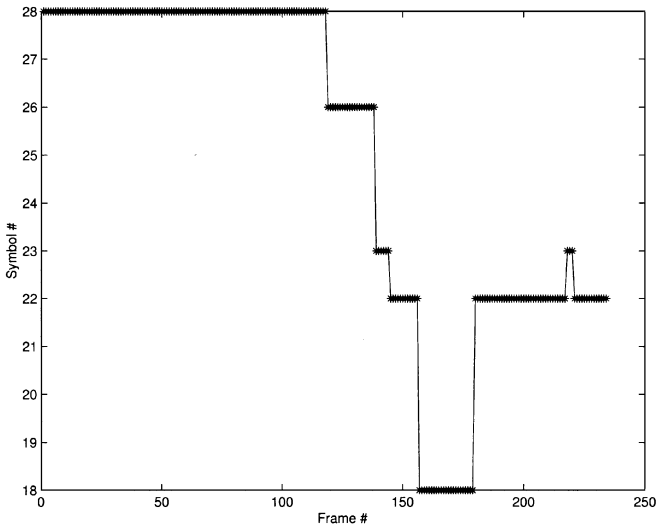


Fig. 6. L sequence for the musical pattern used to demonstrate the limitations of standard DTW schemes. The resulting feature sequence is $D = \{\mathbf{0}_{z_1}, -2, \mathbf{0}_{z_2}, -3, \mathbf{0}_{z_3}, -1, \mathbf{0}_{z_4}, -4, \mathbf{0}_{z_5}, 4, \mathbf{0}_{z_6}, 1, \mathbf{0}_{z_7}, -1, \mathbf{0}_{z_8}\}$.

P_1 might appear to be broken to k_1 jumps $P_1, \ldots, P_{k_1}$ and the same may happen to Q_1, i.e., the symbols $Q_1, \ldots, Q_{k_2}$ are likely to appear. Such a complicated phenomenon generates the following sub-sequence of $\mathbf{D}$

$$\{P_1, \mathbf{0}_{P_1}, \ldots, P_{k_1}, \mathbf{0}_{P_{k_1}} Q_1, \mathbf{0}_{Q_1}, \ldots, Q_{k_2}, \mathbf{0}_{Q_{k_2}}\}$$

where $|\sum_{i=1}^{k_1} P_i| > 20$, $|\sum_{i=1}^{k_1} Q_i| > 20$ and $\sum_{i=1}^{k_1} P_i - \sum_{i=1}^{k_2} Q_i$ is approximately equal to zero or some S_i. Fig. 5 presents the symbol sequence of a musical pattern (before calculating differences), where two cases of large fundamental frequency tracking errors can be observed. In the first case $P_1 + Q_1 = 0$ and in the second case, P_1 is broken into two jumps, $P_1 = -32$ and $P_2 = -24$, whereas $Q_1 = 56$. In this case $Q_1 = P_1 + P_2$.

d) In some cases, certain S_i's are "broken" into two successive jumps whose sum is equal to the original S_i (Fig. 3).

It must be emphasized that, with the exception of variations of type a) and d), all these phenomena are due to errors in the feature generation process and have no relation whatsoever with what the ear perceives.

1) An Example Using Standard DTW: In order to show how the above deviations affect the performance of a standard DTW scheme, we consider, as an example, the feature sequence of Fig. 6 that corresponds to a musical pattern of Type II. The feature sequence for this pattern is

$$\mathbf{D} = \{\mathbf{0}_{z_1}, -2, \mathbf{0}_{z_2}, -3, \mathbf{0}_{z_3}, -1, \mathbf{0}_{z_4}, -4, \mathbf{0}_{z_5}, 4, \mathbf{0}_{z_6}, 1, \mathbf{0}_{z_7}, -1, \mathbf{0}_{z_8}\}.$$

To start with, $\mathbf{D}$ is matched against the reference pattern R_2, using a standard DTW scheme [15]. Any DTW scheme demands taking decisions regarding at least the following parameters:

1) The local and global path constraints.
2) The endpoint constraints.
3) The function that constructs the cost grid.
4) The function that assigns costs to transitions from one node of the cost grid to another.

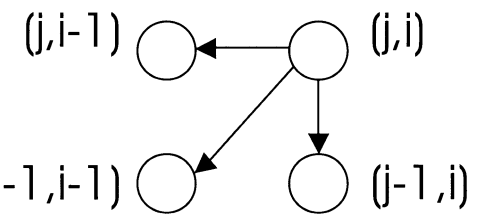


Fig. 7. Popular case of Sakoe–Chiba local path constraints.

Without loss of generality, we focus on a DTW scheme where

1) the Sakoe–Chiba constraints of Fig. 7 are adopted and no global path constraints are used;

2) the best path is restricted to start at node (1, 1) of the grid and end at node (J, I), where J is the length of R_l and I is the length of $\mathbf{D}$;

3) a Euclidean distance determines the costs assigned to the nodes of the cost grid;

4) the cost of a transition $(i_k, j_k) \rightarrow (i_l, j_l)$ from node (i_k, j_k) to node (i_l, j_l) depends only on the cost that has been assigned to (i_l, j_l).

If cost normalization is ignored, the resulting matching cost C in our example is equal to 4. Careful observation reveals that this cost does not depend on the number of successive zeros separating any two nonzero d_i's. This is due to the fact that adjacent zeros result in long horizontal sub-paths in the matching grid (Fig. 8), with each node in the sub-path contributing a zero cost. Therefore, it makes sense to replace each sequence of adjacent zeros with one zero only and rewrite $\mathbf{D}$ as

$$\mathbf{D} = \{0, -2, 0, -3, 0, -1, 0, -4, 0, 4, 0, 1, 0, -1, 0\}.$$

This justifies our decision to represent reference patterns with a single zero, every time a sequence of zeros is encountered.

The matching cost previously extracted is expected to stem from certain deviations of $\mathbf{D}$ from the reference pattern R_2. Indeed, $4 = |-4 - (-3)| + |-1| + |1| + |-1|$. The first two terms are contributed from a deviation of type d) and the second two terms from a deviation of type b). It is important to notice

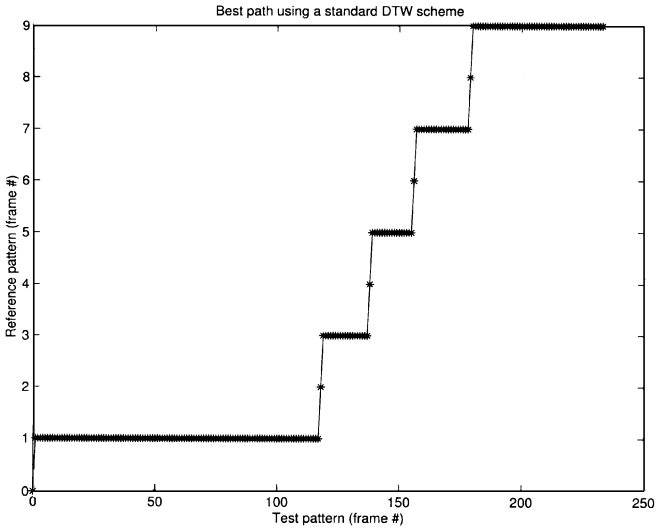


Fig. 8. Best path resulting from the application of a standard DTW scheme. Long horizontal sub-paths have no contribution to the cost.

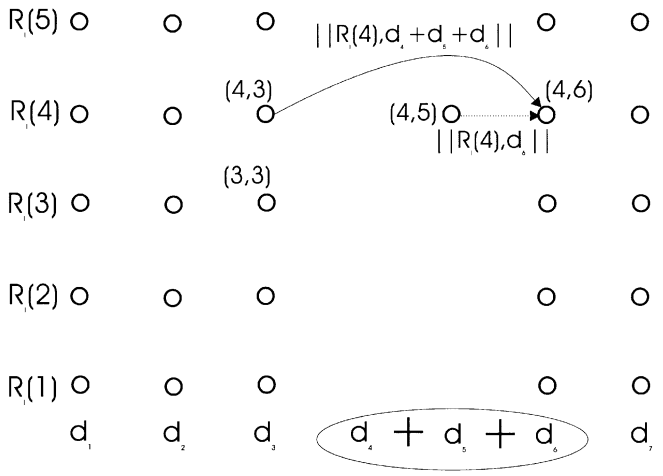


Fig. 9. Euclidean distance below the dotted line is the cost assigned to transition $(4,\,5) \rightarrow (4,\,6)$ via a standard DTW scheme, whereas the distance above the long solid line is the cost assigned to the long transition $(4,\,3) \rightarrow (4,\,6)$.

that the matching cost can be quite high, even though the human ear will classify the pattern with a great degree of confidence. This was indeed the case with informal psychoacoustic experiments conducted in the laboratory for the musical patterns that we studied.

These high costs are due to the fact that standard DTW treats, in general, each symbol of the feature sequence as being uncorrelated with its preceding symbols. However, variations usually appear as groups of symbols. For example, transients of type b) appear in pairs and transients of type c) form even longer groups. Although some variations may be eliminated by means of heuristic post processing rules, such rules are hard to extract, apply and trust. Before we proceed, it is important to keep in mind that, in the case of the standard DTW scheme, the cost $D_{\min}(j,\,i)$ of the best path reaching node $(j,\,i)$ is determined according to the equation

$$D_{\min}(j,\,i) = \min\{D_{\min}(j,\,i-1),\,D_{\min}(j-1,\,i-1), D_{\min}(j-1,\,i)\} + c(j,\,i) \quad (1)$$

where $D_{\min}(j,\,i-1), D_{\min}(j-1,\,i-1), D_{\min}(j-1,\,i)$ is the cost of the best path reaching $(j,\,i-1), (j-1,\,i-1), (j-1,\,i)$ respectively, and $c(j,\,i)$ is the value of the cost grid on $(j,\,i)$. This means that the allowed predecessors of $(j,\,i)$, as shown in Fig. 7, are $(j,\,i-1)$, $(j-1,\,i-1)$ and $(j-1,\,i)$.

2) Description of the CDDTW Algorithm: Let us now introduce CDDTW. At a first step, we define the "*context of length N of a symbol d_i in the feature sequence*" to be the set of symbols $\{d_{i-N+1},\,d_{i-N+2},\,\ldots,\,d_i\}$. This set includes the $N-1$ symbols preceding d_i plus d_i itself. At a second step, we assume that node $(j,\,i)$ can be reached from nodes

$$\{(j,\,i-1),\,(j-1,\,i-1),\,(j,\,i-2), (j-1,\,i-2),\,\ldots,\,(j,\,i-N),\,(j-1,i-N)\}.$$

In other words, the set of allowed predecessors of $(j,\,i)$ is extended to include nodes ranging up to N columns on the left of $(j,\,i)$ in the cost grid, excluding vertical paths, i.e., excluding node $(j-1,\,i)$. In order to define the transition costs, let us first start with an example. Fig. 9 shows a possible transition $(4,\,3) \rightarrow (4,\,6)$. The cost depends on the symbols d_4, d_5 and d_6 of the feature sequence, which form the context of length 3 of d_6. In order to calculate this cost, one has, at first, to sum these symbols, thus generating a new symbol $S = d_4 + d_5 + d_6$. In the sequel, the Euclidean distance between S and $R_l(4)$ is computed and this is defined as the cost associated with the specific transition.

Summing symbols is an attempt to cancel out (from sequence **D**), the deviations described in Section III-B. The longer the transition, the more complicated the deviations that are canceled out. For simple deviations, such as those of type a), b), and d), short transitions are sufficient. However, when the complex version of deviations of type c) is encountered, or when deviations are combined to generate complex phenomena, long transitions, involving up to nine symbols, are necessary. The transition marked with a solid line in Fig. 9 is a relatively short one, involving the symbols d_4, d_5 and d_6. Transitions of this type are expected to cancel out simple deviations of type b).

In the general case, the cost of a transition $(j,\,i-k) \rightarrow (j,\,i)$ or $(j-1,\,i-k) \rightarrow (j,\,i)$ is equal to $\|R_l(j),\,\sum_{m=i-k+1}^{i} d_m\|$. The cost $D_{\min}(j,\,i)$ of the best path reaching node $(j,\,i)$ is therefore equal to the minimum cost generated by the paths reaching $(j,\,i)$ and is computed according to the following equation:

$$\begin{aligned} D_{\min}(j,\,i) = \min \Bigg\{ & D_{\min}(j,\,i-1) + \|R_l(j),\,d_i\|, \\ & D_{\min}(j-1,\,i-1) + \|R_l(j), d_i\|, \\ & D_{\min}(j,\,i-2) + \|R_l(j),\,d_i + d_{i-1}\|, \\ & D_{\min}(j-1,\,i-2) + \|R_l(j),\,d_i + d_{i-1}\|,\,\ldots \\ & \ldots, D_{\min}(j,i-N) + \left\| R_l(j), \sum_{m=i-N+1}^{i} d_m \right\|, \\ & D_{\min}(j-1,i-N) + \left\| R_l(j), \sum_{m=i-N+1}^{i} d_m \right\| \Bigg\}. \end{aligned} \quad (2)$$

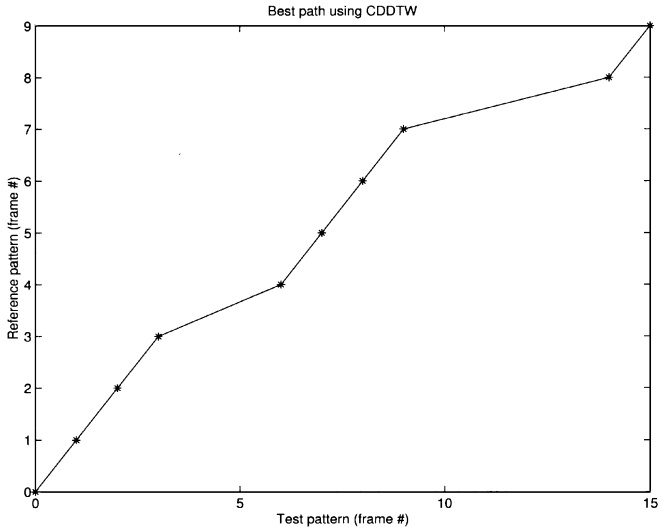


Fig. 10. Best path resulting from the application of the CDDTW scheme, corresponding to Fig. 8.

In the example used before with DTW, employing CDDTW to match sequence $\mathbf{D} = \{0, -2, 0, -3, 0, -1, 0, -4, 0, 4, 0, 1, 0, -1, 0\}$ against R_2 yields a zero cost. The resulting best path is shown in Fig. 10. It can be observed that CDDTW treated frequency jumps $\{d_4 \cdots d_6\} \equiv \{-3, 0, -1\}$ as one jump equal to $-3 + 0 + (-1) = -4$ and frequency jumps $\{d_{10}, \ldots, d_{14}\} \equiv \{4, 0, 1, 0, -1\}$ as one jump equal to $4 + 0 + 1 + 0 + (-1) = 4$. In other words, the first of the two summations canceled out a deviation of type d) and the second one a deviation of type b). This is the key that reduces the overall cost to zero.

It is possible to reduce further the computational complexity of CDDTW if the following observation is taken into account: in the feature sequence

$$\mathbf{D} = \{0, -2, 0, -3, 0, -1, 0, -4, 0, 4, 0, 1, 0, -1, 0\}$$

each pair of nonzero symbols is separated by a single zero. Due to the way CDDTW works a zero does not alter the cumulative jumps that are calculated. Therefore, it is possible to omit zeros entirely, both from the features sequence of the unknown pattern and the reference pattern. In our example, $\mathbf{D}$ becomes $\{-2, -3, -1, -4, 4, 1, -1\}$ and R_2 becomes $\{-2, -4, -4, 4\}$. This suggests that it suffices to keep the nonzero d_is from the original feature sequence.

3) Determination of the Endpoints of the Unknown Feature Sequence: A final issue that has to be dealt with is the determination of the endpoints of the feature sequence to be recognized. This consideration stems from the fact that, although the endpoints of the reference pattern are precisely known, this is not always the case with the unknown musical pattern due to the segmentation procedure that has isolated it from its context. In this paper, we have so far assumed that the best path always begins with node (1, 1) and ends at node (J, I). In standard DTW schemes, the Bridle algorithm [22], [23] is usually employed in order to let the DTW scheme itself to automatically detect the endpoints of the feature sequence to be recognized. This is the technique that we also adopted and implies that the best path is allowed to begin with any one of the nodes $(1, 1), \ldots, (1, e_1)$ and end at any one of the nodes $(J, I-e_2), \ldots, (J, I)$. For the signals that we studied, we let $e_1 = 3$ and $e_2 = 3$.

4) Pseudo-Language Description of CDDTW: To summarize, we present below, in pseudo-language, the resulting CDDTW algorithm, which consists of three stages: initialization, recursion and termination. J is the length of the reference sequence R_l and I is the length of the feature sequence to be recognized. Each element (j, i) of the two-dimensional array $D_{\min}$ is used to hold the cost of the best path reaching node (j, i) of the matching grid. The two-dimensional array P is used to store the indexes of the predecessor of every node (j, i) in the best path reaching (j, i).

Initialization: $D_{\min}(1, i) = \|R_{M_i}(1), d_i\|, \ i = 1, \ldots, e_1$
$P(1, i) = (0, 0), \ i = 1, \ldots, e_1$
$D_{\min}(1, i) = \infty, \ i = e_1 + 1, \ldots, I$
$D_{\min}(j, 1) = \infty, \ j = 2, \ldots, J$
Set context length N (*e.g.* $N = 9$)

Recursion: *For* $i = 2, \ldots I$
For $j = 2, \ldots J$
Compute $D_{\min}(j, i)$ *as in* (2)
Store $P(j, i)$
Next j
Next i

Termination: *Total Cost* $= \min\{D_{\min}(J, I), \ldots, D_{\min}(J, I-e_2)\}$

The best path is extracted from matrix P, starting from the node that corresponds to the minimum total cost and going backward until we reach the fictitious node (0, 0), which has been defined as the predecessor of (1, 1), while initializing the algorithm.

The major computational burden of the CDDTW technique is contributed by the cost in equation (2). For each node of the cost grid a "min" operation is required. Furthermore, for a context length of N symbols, $N^2 + 4N$ additions and $2N$ multiplications are required. Thus, for a grid of I, J dimensions the overall computational complexity amounts to $IJ(N^2 + 4N)$ additions and $2IJN$ multiplications. A typical value for N is 9 and in our experiments worst case values for I, J were $I = 20$ and $J = 30$.

IV. Application of the Method in the Context of Greek Traditional Music

The Greek Traditional clarinet is an instrument that closely resembles the western-type clarinet. Its spectral properties change as the frequency increases and three spectral regions can be observed:

- A low frequency region (between $D3 = 146.8$ Hz and $D4 = 294$ Hz) with a relatively weak fundamental and a strong second harmonic.
- A middle frequency region (between $D\#4 = 311$ Hz and $G\#4 = 415$ Hz) with a strong fundamental.

- A high frequency region (between $A4 = 440$ Hz and $C\#6 = 1109$ Hz), with a strong fundamental and several even and odd harmonics present.

We have adopted the notation that the fundamental coincides with the first harmonic, the second harmonic is twice the fundamental, etc. The lowest possible fundamental that the instrument can produce depends on its tuning. For the purpose of our study, this was measured to be equal to $D3 = 146.8$ Hz.

A set of 1200 musical patterns were generated by four professional Greek Traditional Clarinet players in a monophonic environment, involving all the aforementioned twelve types of musical patterns. The sampling rate, F_s, was set equal to 22 050 Hz. For the feature generation stage, the new fundamental frequency tracking algorithm along with the narrowed autocorrelation method and Tolonen's multipitch analysis model were extensively tested. For the new algorithm, the frame length was chosen equal to 4096 samples. The moving window was shifted 10 ms at each step and was multiplied with a Hamming function prior to the calculation of the Fourier coefficients. The above frame length indicated that the smallest distinguishable frequency jump was equal to $F_s/4096 = 5.38$ Hz and ensures that a quarter-tone resolution was possible for frequencies above 185 Hz. In the frequency region below 185 Hz, although only half-tone resolution was achieved, this was not crucial, because quarter-tone phenomena hardly ever appear in this very low spectral region of the instrument. In addition, with a window length equal to a power of two, it was possible to use the well known Fast Fourier Transform for the calculation of the Fourier coefficients. A Hamming window was chosen as a window multiplier because it provides a less abrupt transition at the boundaries of the frame (compared to a rectangular window) and exhibits preferable sidelobe characteristics (an attenuation of -30 dB in the sidelobes).

The values of T_p, T_h and K (see Section II-A) were determined after extensive experimentation. Specifically, T_p was chosen equal to 1/8, $T_h = (1/15)(H_a - L_a)$ and $K = 5$, where H_a denotes the amplitude of the highest Fourier peak of the spectrogram of the musical pattern and L_a the amplitude of the smallest Fourier peak of the spectrogram of the musical pattern.

For the quantization step we used an alphabet of 121 discrete symbols, with each symbol being equal to a frequency jump in the range of $-60 \cdots + 60$ quarter-tones, i.e., $G = 60$ (Section II-B).

Two sets of experiments were carried out. One using the standard DTW technique employing Itakura and Sakoe-Chiba constraints. The latter proved to be more robust for the signals of our interest since these constraints allow for long horizontal and vertical paths in the cost grid. The success rate obtained was of the order 93%, with little variations depending on the pitch extraction algorithm used. The other set of experiments employed the new CDDTW scheme and the success rate was significantly improved to above 95%. It must be stated that this method was basically immune to variations of the pitch tracking method used. The context length for the CDDTW scheme was set equal to 9 symbols. All experiments were carried out using the MATLAB environment.

V. Conclusions and Future Research

In this paper, an efficient scheme for the recognition of isolated musical patterns was presented. The scheme is based on CDDTW, a novel extension of standard DTW schemes. The feature generation stage of the scheme employs a new fundamental frequency tracking algorithm. The methodology was applied with success in the context of Greek Traditional Music. A reason for this choice is that it provides a musically homogeneous material, generated by the traditional mode of playing the instrument, and at the same time presents many constraints (like the unequal musical intervals and the change of the spectral content of the sound depending on the playing mode). Future research will focus on applying this new recognition scheme in the context of Classic Western Music, with other instruments besides clarinet and with multi-dimensional feature vectors.

References

[1] J. A. Moorer, "Signal processing aspects of computer music—A survey," *Proceedings of the IEEE*, vol. 65, no. 8.

[2] M. R. Schroeder, "Period histogram and product spectrum: New methods for fundamental-frequency measurement," *Journal of the Acoustical Society of America*, vol. 43, no. 4, 1968.

[3] M. Piszczalski and B. Galler, "Predicting musical pitch from component frequency ratios," *Journal of the Acoustical Society of America*, vol. 66, no. 3, 1979.

[4] J. C. Brown, "Musical fundamental frequency tracking using a pattern recognition method," *Journal of the Acoustical Society of America*, vol. 92, no. 3, 1992.

[5] J. C. Brown and B. Zhang, "Musical frequency tracking using the methods of conventional and narrowed autocorrelation," *Journal of the Acoustical Society of America*, vol. 89, no. 5, 1991.

[6] J. C. Brown, "A high resolution fundamental frequency determination based on phase changes of the Fourier transform," *Journal of the Acoustical Society of America*, vol. 94, no. 2, 1993.

[7] D. Cooper and K. C. Ng, "A monophonic pitch-tracking algorithm based on waveform periodicity determinations using landmark points," *Computer Music Journal*, vol. 20, no. 3, Fall 1996.

[8] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 6, November 2000.

[9] H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, February 1978.

[10] H. Sakoe, "Two-level DP matching: A dynamic programming based pattern recognition algorithm for connected word recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 27, December 1979.

[11] F. Itakura, "Minimum prediction residual principle applied to speech recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 23, February 1975.

[12] H. F. Silverman and D. P. Morgan, "The application of dynamic programming to connected speech recognition," *IEEE ASSP Magazine*, July 1990.

[13] J. Deller, J. Proakis, and J. Hansen, *Discrete-Time Processing of Speech Signals*: McMillan, 1993.

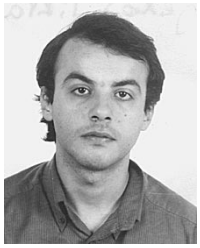
[14] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*: Academic Press, 1998.

[15] A. Pikrakis, S. Theodoridis, and D. Kamarotos, "Recognition of isolated musical patterns in the context of greek traditional music using dynamic time warping techniques," in *Proceedings of the International Computer Music Conference (ICMC)*, 1997.

[16] ——, "Recognition of isolated musical patterns using discrete observation hidden Markov models," in *Proceedings of EUSIPCO*, 1998.

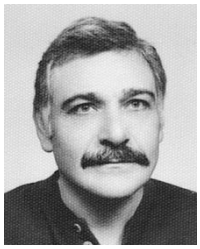
[17] D. R. Stammen and B. Pennycook, "Real-time recognition of melodic fragments using the dynamic time warping algorithm," in *Proceedings of the International Computer Music Conference (ICMC)*, 1993.

[18] B. Kostek, "Computer-based recognition of musical phrases using the rough-set approach," in *Information Sciences 104*, 1998.
[19] B. Kostek and M. Szczerba, "Parametric representation of musical phrases," in *101st Convention of the AES*, November 1996.
[20] P. E. Papamichalis, *Practical Approaches to Speech Coding*: Prentice-Hall, 1987.
[21] S. Karas, *Theoritikon—Methodos, on Greek Traditional Music* (in Greek) Athens, 1982.
[22] J. S. Bridle and M. D. Brown, "Connected word recognition using whole word templates," in *Proceedings of the Institute for Acoustics, Autumn Conference*, Nov. 1979.
[23] J. S. Bridle, R. M. Chamberlain, and M. D. Brown, "An algorithm for connected word recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 2, 1982.



Aggelos Pikrakis (A'97) received the diploma in computer engineering and informatics from the University of Patras, Greece, and the Ph.D. degree in signal processing from the University of Athens, Greece.

He is currently a Research Fellow at the University of Athens. His research interests are in the areas of signal processing for music with emphasis on Greek traditional music, content based music retrieval, and intelligent agents for data mining applications.



Sergios Theodoridis (M'87–SM'00) received an honors degree in physics from the University of Athens, Greece, and the M.Sc. and Ph.D. degrees from the Department of Electronics and Electrical Engineering of Birmingham University, U.K.

He is currently a Professor of signal processing and communications in the Department of Informatics and Telecommunications of Athens University. His research interests lie in the areas of adaptive algorithms, channel equalization, pattern recognition, signal processing for music and OCR systems. He has published more than 100 papers in prestigious international journals and refereed conferences. He is the co-editor of the book *Efficient Algorithms for Signal Processing and System Identification* (Englewood Cliffs, NJ: Prentice-Hall 1993), the co-author of the book *Pattern Recognition* (New York: Academic, 1998), and three books in Greek, two of them for the Greek Open University. He is member of the editorial boards of *Signal Processing* and *Applied Signal Processing*.

Dr. Theodoridis is currently an associate editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He is a member of the adcom of EURASIP and a Fellow of IEE.

Dimitris Kamarotos studied music and computers in Athens, Grfeece, and continued musicology, composition, clarinet, and electronic music in Paris, France. His postgraduate research focused on the creation of tools for semi-automated music composition (IR-CAM 1984).

Since 1986, he has worked as Research Manager in many projects in collaboration with Athens Polytechnic School, University of Thessaloniki, and the Center for Contemporary Music Research. He was Research Manager of the H.X.E. Project on creation of tools for the automated comparison of audio patterns. The same idea of automated pattern recognition extended into a monophonic musical environment was the field of research in collaboration with Prof. S. Theodoridis and Dr. A. Pikrakis. He also collaborated with B. Garton and T. Rikakis (Columbia University, NY) and P. Cook (Princeton University, Princeton, NJ) in the foundation of the computer music studio in CCMR and the IPSA.