On the Automated Recognition of Seriously Distorted Musical Recordings

Dimitrios Fragoulis, George Rousopoulos, Thanasis Panagopoulos, Constantin Alexiou, and Constantin Papaodysseus

***Abstract*—In this paper, a new methodology is presented for the automated recognition-identification of musical recordings that have suffered from a high degree of playing speed and frequency band distortion. The procedure of recognition is essentially based on the comparison between an unknown musical recording and a set of model ones, according to some predefined specific characteristics of the signals. In order to extract these characteristics from a musical recording, novel feature extraction algorithms are employed. This procedure is applied to the whole set of model musical recordings, thus creating a model characteristic database. Each time we want an unknown musical recording to be identified, the same procedure is applied to it, and subsequently, the derived characteristics are compared with the database contents via an introduced set of criteria. The proposed methodology led to the development of a system whose performance was extensively tested with various types of broadcasted musical recordings. The system performed successful recognition for the 94% of the tested recordings. It should be noted that the presented system is parallelizable and can operate in real time.**

***Index Terms*—Automatic music recognition, distorted in frequency recordings, fuzzy logic and music, musical recording automated recognition, music pattern recognition, music processing.**

I. Introduction

CURRENT research in the field of music pattern recognition and processing, among others, deals with classical pattern recognition methods used to correlate small-duration parts of music [7], [8] with automatic music transcription [4]–[6]. Moreover, a considerable effort has been made to apply the techniques of connectionism and parallel distributed processing (PDP) in a wide range of topics in music [10]–[12], [16]–[20]. The so-called "connectionist" or neural network computer models allow investigation of processes, such as learning, generalization, and forms of representation, that are difficult or impossible to study in earlier physiological models. Hence, they can probably help us to learn more about the processes and representations involved in music perception.

Music transcription systems usually work by deriving information about the tempo, the scale, the sound length, and the key of a musical signal (for definitions see the Glossary in Appendix C). However, songs are not stable signals in time; they contain abrupt sounds, and they have many fluctuations in pitch, which are factors that reduce the accuracy of sound segmentation, thus reducing the accuracy of the developed musical score data. Since the existing music transcription systems generate musical score data with low accuracy, they are not of widespread practical use.

It seems that analysis of music into notes is unnecessary for classification of music. Thus, more effort should be spent attempting to build systems that operate directly on music. An interesting speech/music discrimination system based on features that were thought to be useful discriminators was presented recently [21]. This system does not recognize musical recordings but instead classifies a signal as speech or music, assuming that there are no regions of overlap.

In any case, the realization of a system that automatically recognizes musical recordings remains one of the major issues in the field of one-dimensional (1-D) digital signal processing. Such a system could find extended application to the automatic broadcast counting and would be a very useful tool for companies in the field of intellectual property rights or companies that compile musical data for statistical purposes (e.g., charts).

The methodology introduced in this paper provides the ability to develop such a system that accomplishes automatic recognition of an unknown musical recording, among a set of others considered to be the model ones. The system works successfully for signals that have suffered a frequency-speed distortion up to a high degree, which is the case for most of the musical recordings received by radio. The term "frequency-speed distortion up to a high degree" or simply "up to a high frequency-speed distortion" is used to describe the following.

a) There may be a non-audible noise present at the radio received signal.

b) The CD and radio obtained musical recordings may have been played at arbitrarily different speeds. Most of the tested recordings have shown a playing speed difference of up to 5% but have been observed differing the playing speeds up to 15%. A change in playing speed essentially causes a "stretch" to the spectral shape of the recording (see Section V). Therefore, the frequency components of the recording are shifted from their initial positions, and as a result, the sound quality changes to a point. Experiments have shown that the "just noticeable difference" (jnd) of two sinusoidal tones of different frequency varies between 0.1%–0.2% [22], [23]. However, for complex signals such as musical recordings that include a variety of spectral components with different duration and intensities, it seems quite difficult to define a "just noticeable playing speed difference" threshold. In spite of the objective difficulties, we have observed, without claiming that this constitutes a founded psychoacoustic experimental, that a playing speed difference smaller than

Manuscript received October 12, 1999; revised December 6, 2000. The associate editor coordinating the review of this paper and approving it for publication was Prof. Bjorn Ottersten.

The authors are with the Department of Electrical and Computer Engineering, Division of Computer Science, National Technical University of Athens, Athens, Greece (e-mail: frag@mlab.ece.ntua.gr; cpapaod@cs.ntua.gr).

Publisher Item Identifier S 1053-587X(01)02251-6.

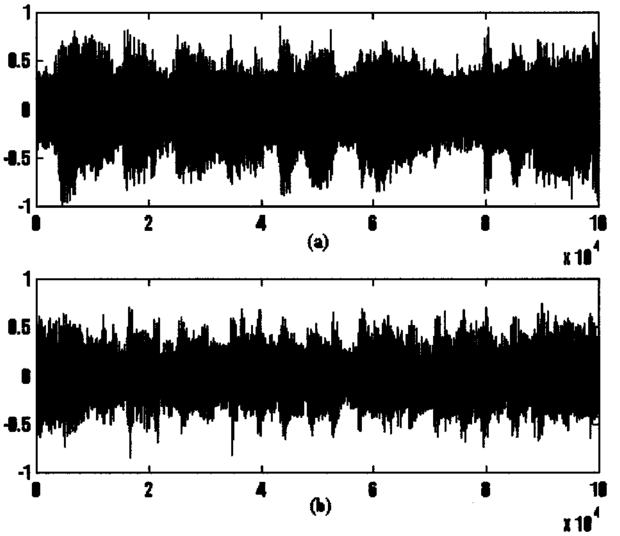


Fig. 1. Depiction of the differences in the time domain between a part of a model musical composition and a part of a radio sampled one that correspond to exactly the same piece of music (from “Born to be Wild” by Steppenwolf).

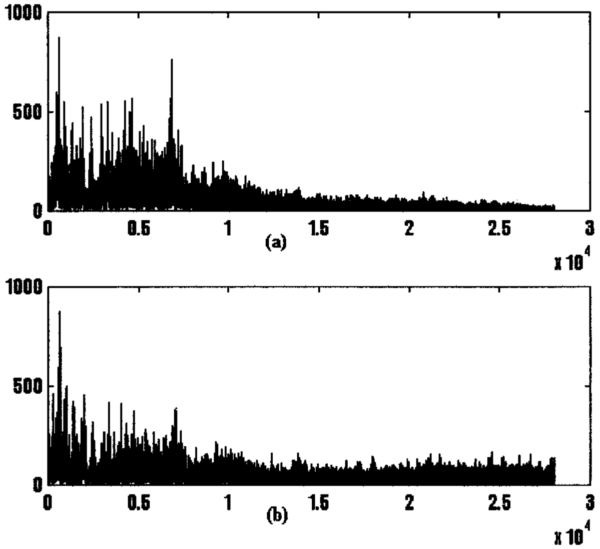


Fig. 2. Depiction of the differences in the frequency domain between a part of a model musical composition and a part of a radio sampled one that correspond to exactly the same piece of music (from “Baker Street” by Gerry Rafferty).

approximately 2% does not cause a noticeable change in the sound quality of the musical recording.

c) Radio stations that transmit a considered musical recording may amplify frequency bands.

The experiments have been performed on approximately 920 recordings obtained from 18 different FM radio stations and cover a very extended range of signal strengths. Five of them use a compressed form for recordings they broadcast (e.g., transmitting MPEG–Layer 3 compressed music). These experiments show that at least 96% of the obtained recordings satisfy the aforementioned conditions. Notice that a radio-received signal having suffered even a small distortion of the type described in a)–c) can manifest an obvious discrepancy from its CD counterpart both in time and frequency domain: a fact that creates serious difficulties in the automatic recognition procedure (see Figs. 1 and 2).

The introduced methodology, as it will be shown in the following, offers the ability to distinguish musical parts that correspond to the same melodic pattern (sequence of notes that constitute a melody), such as different performances of the same musical composition. This characteristic of the method and system introduced here is related to the specific feature extraction procedure applied to the input musical part. As an example, we can consider the case of two recordings of the same melody performed by two different singers, accompanied by either the same or by different instruments. The proposed model, with proper adjustment of its parameter values, is able to distinguish these two recordings despite the fact that they correspond to the same melodic pattern. Therefore, we can say that speaker recognition and voice verification are two potential applications of the introduced methodology. According to the above discussion, if we consider recordings of a group of speakers articulating a defined set of phrases, then it is expected that the model, after proper changes, might be able to recognize either the speaker articulating a specific phrase or the phrase articulated by a specific speaker.

II. Problem Description

If one attempts to recognize a musical recording automatically, one, among others, faces the following difficulties.

a) Usually, a musical recording is a fast varying signal comprising a variety of different signals such as the voice of the singer and the sounds produced by various musical instruments. Therefore, a musical recording is a mixture of many frequencies: a fact that makes the identification of single instruments in it and the note transcription an extremely difficult task.

b) In order for such a system to have a serious applicability, it must be able to recognize a musical recording among many tenths of hundreds or thousands of others. Considering that a 3-min-long musical recording in “.wav” form occupies approximately 10 MB and that, in order to obtain CD quality sound, one has to sample the musical recording at a sampling frequency of 44 100 samples/s, it is obvious that one must manipulate a huge amount of information.

c) The transmission and reception procedure can distort the time and frequency domain information of each musical recording.

d) The fact that radio and TV station personnel (e.g., DJ) frequently acts on a transmitted musical recording at will either by amplifying selected frequency bands and/or by changing the speed the source (CD, tape, etc.) is played seems to constitute perhaps the most important problem since it is not possible to model such a disturbance.

III. Presentation of the Proposed Methodology

Consider two signals corresponding to the same musical recording: one received by a radio or a TV set and, therefore, having suffered an arbitrary distortion due to the reasons referred to in the previous section, and the other obtained from a CD. First of all, the aforementioned distortion drastically changes the quantitative information obtained from a time domain signal analysis. For example, if one considers quantities such as number of zero-crossings per frames of N samples, relative position of peaks, average slope or curvature per frame, relative amplitude of peaks etc., for the model song obtained from CD and the very same musical recording received by a radio or a TV set, then it is clear that huge differences occur between values of the same quantities for the two signals. Clearly, the greater the distortion, the greater the discrepancy between these values.

Similarly, in the frequency domain, serious discrepancies appear between many (if not drastically most) quantitative characteristics-parameters of the Discrete Fourier Transform (DFT) performed on the CD musical recording and the DFT of the signal received by a radio or a TV set. For example, the number of peaks per frame of N samples, the actual DFT peaks amplitude, the energy per various bands, the order of the higher peaks etc., manifest serious discrepancies (see Fig. 2).

However, we have spotted some critical similarities between the spectrum of the CD-obtained signal and the radio-obtained signal of the very same musical recording, and we exploited them in order to achieve automatic recognition of musical recordings. After an extended number of experiments, we have reached the conclusion that the musical information existing in a time frame of a musical recording is intimately connected to the position of spectral peaks of this frame. Therefore, if the spectral peak position information is kept for sufficiently many frames starting at various time instances of a musical recording, then the recording identification is achieved. However, since it is impossible to store such a large amount of information for just one musical recording, a reduction of the necessary storage capacity is attempted by a division of the frequency domain in bands. The width of the bands is chosen to be almost exponentially augmented in order to imitate the frequency selectivity of the human ear, i.e., the experimentally verified shape of the auditory filter [14], [15].

IV. Dividing the Audibility Domain into Bands

The whole audibility domain is divided into 57 bands of almost exponential width, as shown in Table I. It is possible that other divisions in bands also work well. In any case, the final criterion for the correctness of a choice is the efficiency of recognition validated by the experiment. In addition, the proper division in bands is strongly correlated with the degree of distortion that the whole set of unknown musical recordings has suffered. If it can be ensured that the considered unknown musical recordings have suffered a smaller distortion than the one described in the previous section, then another choice of band division may be optimal. The usefulness of dividing the audibility domain into bands will be made clear in the subsequent sections.

V. Building a Set of "Band Representative Vectors" for the Unknown Musical Recording

Suppose that a part of an unknown musical recording is given in order to be recognized automatically. Then, at first, we do the following.

TABLE I
Division of the Spectrum into 57 Bands

BAND INDEX	BAND RANGE (IN Hz)	BAND INDEX	BAND RANGE (IN Hz)	BAND INDEX	BAND RANGE (IN Hz)
0	0 - 50	20	330 - 355	40	1467 - 1579
1	51- 80	21	356 - 382	41	1580 - 1702
2	81- 90	22	383 - 412	42	1703 - 1834
3	91- 100	23	413 - 444	43	1835 - 1976
4	100 - 107	24	445 - 478	44	1977 - 2129
5	108 - 116	25	479 - 516	45	2130 - 2293
6	117 - 125	26	517 - 556	46	2294 - 2471
7	126 - 134	27	557 - 599	47	2472 - 2663
8	135 - 145	28	600 - 645	48	2664 - 2869
9	146 - 156	29	646 - 695	49	2870 - 3090
10	157 - 168	30	696 - 749	50	3091 - 3326
11	169 - 181	31	750 - 807	51	3327 - 3577
12	182 - 195	32	808 - 869	52	3578 - 3835
13	196 - 210	33	870 - 937	53	3836 - 4103
14	211 - 227	34	938 - 1009	54	4104 - 4380
15	228 - 244	35	1010 - 1088	55	4381 - 4700
16	245 - 263	36	1089 - 1172	56	4701 - 11025
17	264 - 284	37	1173 - 1263		
18	285- 306	38	1264 - 1360		
19	307 - 329	39	1361 - 1466		

A1) We take at random a part of the unknown musical recording, of length BL samples, and we transform it into a form suitable for processing by a computer, preferably in ".wav" format. In the subsequent analysis, we will refer to it by the name "the radio signal part," although the unknown musical recording can be obtained from CD, television, tape, or any other related source for which the introduced automatic recognition methodology works perfectly well too.

A2) At the beginning of this signal, we pick a "first frame" of N samples (say, $2^{13} = 8192$), and we apply the DFT transform on them.

A3) We calculate the absolute value of this DFT transform, and then, we apply a masking-like procedure described in Appendix A, on each peak of it. This procedure causes the elimination of some acoustically less important peaks of the DFT. Next, the positions of the remaining peaks are successively multiplied by the stretch or shift factors: $f_i, i = 0, 1, \cdots, S$. In this way, $(S + 1)$ shifted copies of the peaks are derived and stored temporarily in $(S + 1)$ arrays. On every one of these arrays, the procedure described in **A4**) and **A5**) is applied.

A4) We assign each peak of the current array to the proper band (see Table I). If more than one peak belong to the same band, we choose, among them, the maximum amplitude peak, and we consider it to be the amplitude of the specific band corresponding to the shift factor f_i. Otherwise, if no peaks belong to a band, we assign a zero value to this band amplitude in connection with the shift factor f_i.

A5) We find the L bands of greater amplitude, and we store their corresponding index numbers. In this way, we obtain a vector we call "*the first band representative vector corresponding to the shift factor* f_i," which corresponds to the first frame with values the aforementioned index numbers.

The above procedure results in the creation of a group of $(S+1)$ "*first band representative vectors*."

B1) We choose a "second frame" of N samples [N exactly the same as in step **A2**) above] at a fixed distance of l time samples from the first sample of the radio signal part, and we apply the DFT transform to them.

Next, we repeat steps **A2**)–**A5**) for this "second frame" in order to obtain a group of $(S+1)$ "*second band representative vectors*."

•

•

•

$\mathrm{B_M}$) We choose an "Mth frame" of N samples at a fixed distance of l time samples from the first sample of the $(M-1)$th frame, and we apply the DFT transform to them.

Next, we repeat steps **A2**)–**A5**) for this "Mth frame" in order to obtain a group of $(S+1)$ "M*th band representative vectors*."

In this way, we obtain M groups of band representative vectors [each group consisting of $(S+1)$ vectors], corresponding to the above M chosen frames.

Notice that the experiments we have performed show that the number L of band representatives must satisfy the inequality $17 \leq L \leq 25$.

VI. Building a Set of "Band Representative Vectors" for a Model Musical Recording

We apply an analogous procedure to each signal obtained from a CD (we will call it ”the CD signal"). In fact, we take N samples starting at the first sample of the CD signal, we repeat steps **A2**)–**A5**), and in this way, we create a vector of L elements. We do the same for every sample of the CD signal, thus finally obtaining a set of vectors where each consists of L elements. We will use for this set the name "*model set of band representative vectors*." Notice that it is very usual that two or more consecutive time samples correspond to identical band representative vectors. Therefore, we attach to each such vector the number of time samples for which it remains identical, which we will call "repetitions number of the vector."

The creation of those vectors requires a considerable amount of computational complexity since it involves a fast Fourier transform (FFT) computation of N samples for each sample of the CD signal in hand. Considering that a musical recording of 3–5 min in duration, sampled at a rate of 22 050 samples/s, consists of approximately $3,\ 5 * 60 * 22\,050 = 4\,630\,500$ samples, it is clear that the creation of these vectors may require many hours of computations, even if a 500-MHz Pentium III processor is used. Clearly, for a longer recording, say a classical one of 30-min duration, the creation of these vectors may require several days if the classical FFT method is used. For this reason, we have applied an adaptive FFT computation algorithm, which is presented in Appendix B, that achieves a considerable reduction of the overall computation time.

VII. Coding of the Band Representative Vectors of the Model Set

The band representative vectors derived from each model musical recording require a great amount of storage capacity if they are stored directly in a file with the ASCII format or even with the standard binary format. Therefore, we have developed an efficient binary-encoding scheme that drastically reduces the required storage amount without any loss of information, thus also decreasing the access time to the band representative vectors. This scheme will be presented below.

As mentioned in the previous subsection, each band representative vector consists of L index numbers, where each index characterizes a frequency band. The order of these band indices in each vector is not of importance for the automatic recognition method we employ; therefore, we store them in descending value. In addition, it is quite common that consecutive band representative vectors, i.e., vectors that correspond to windows that differ only in one sample, are exactly the same. Therefore, in order to store a sequence of identical band representative vectors, it is sufficient to store the corresponding vector once, together with the number of consecutive identical band representative vectors, which we will call "number of repetitions." In the following, when we refer to a band representative vector, we consider that a corresponding number of repetitions is attached to it.

In order to obtain a more efficient coding, we exploit the fact that even when two consecutive band representative vectors are different, the number of different entries in them is typically very small, usually one or two. Therefore, we have developed the following differential coding algorithm.

A) The band representative vector corresponding to the first sample of the model musical recording in hand is stored as follows: We assign to each band representative vector of the CD signal a 57-element binary array. Each element of this binary array represents one of the 54 bands into which we have decided to divide the whole audibility domain. A value of "1" is assigned to an array element when the corresponding band is one of the L greater in amplitude bands of the window in hand with a nonzero value, whereas a value of "0" is assigned otherwise. Notice that at most L bits can be set to "1."

For example, if $L = 18$, then a possible band representative vector is

$$[56\ 51\ 48\ 45\ 40\ 34\ 31\ 28\ 27\ 23\ 20\ 19\ 15\ 11\ 9\ 6\ 2\ 1].$$

We create a binary array of 58 binary digits, all elements of which are zero, except those with a position corresponding to a band index of the above vector. Therefore, we obtain the array

$$[0110001001010001000110010001\\ 10010010000010000100100100001].$$

B) In order to store the information of subsequent band representative vectors, we consider the number of different entries between the vector in hand and the previous vector.

When the number of different values is less than or equal to 3, we simply store the frequency band index numbers that belong to the present vector but not in the previous one (named "incoming indices") and the index numbers of the frequency bands that were entries of the previous vector but are not present in the vector in hand (named "outgoing indices"). We let the outgoing indices first, and the incoming indices next, form an array called "the information array." If the number of outgoing indices in not the same as the number of incoming indices, then the index "-1" is inserted in the proper place.

When this number of different values is greater than 3, we store the whole vector with the compressed scheme described in **A)**. Notice that with this method, the storage of the model band representative vectors requires 35 times less space than the original index-numbers array and approximately nine times less space than the one used by a simple binary coding method.

In both cases, regardless of the number of different entries, an additional number must be stored to represent the number of repetitions of each band representative vector. To survive a further storage reduction, we use a variable number of bytes for the storage of this number, according to its size, as shown in Table II.

VIII. Pattern Matching Algorighm for the Band Representative Vectors

Consider two sets of band representative vectors that correspond to exactly the same piece of music: one to the "radio/TV received" musical recording and the other to the model one. All entries of these vectors cannot, in practice, be identical due to the existent distortion. Therefore, in order to achieve musical recording recognition, it is absolutely necessary to employ a pattern matching algorithm that allows for a successful matching between two sets of vectors, even if they have a considerable number of different elements.

Thus, a pattern matching algorithm has been developed, where each band representative vector of the radio/TV received recording part is considered as an independent state. Transition to the mth state is allowed if and only if all imposed restrictions in the previous $m-1$ states are satisfied. To set ideas, we compare the first band representative vector $\mathbf{V_1}$ corresponding to the shift factor f_i of the unknown part with the first band representative vector $\mathbf{U_{1,\,n}}$ of a model musical recording (where index $\mathbf{n}$ expresses the starting time sample of the window that has generated the representative vector in hand), and then, we have the following.

TABLE II
(a) Number of Bytes Used for the Storage of the Information Array and the Number of Repetitions, According to Its Size. (b) Code Character Value According to the Number of Changes and Number of Repetitions

Code Character	Information Array	Repetitions
1 byte	2,4,6 or 8 bytes	1,2 or 4 bytes

(a)

		Code character value according to number of changes and number of repetitions		
		Number of Repetitions		
		0<Number<256	255<Number<64768	64767<Number
Number of changes	1	**4**	**7**	**10**
	2	**5**	**8**	**11**
	3	**6**	**9**	**12**
	>3	**1**	**2**	**3**

(b)

$\mathbf{A_1}$) If the number of common elements is less than $0.53 * L$, then we stop the comparison procedure in hand, and we restart to compare vector $\mathbf{V_1}$ with the next band representative vector of the model set $\mathbf{U_{1,\,(n+1)}}$.

$\mathbf{B_1}$) If the number of common elements is greater than or equal than $0.53 * L$, then, and only then, we proceed to the comparison between the second band representative vector $\mathbf{V_2}$ that corresponds to the same shift factor f_i of the unknown recording and the vector $\mathbf{U_{2,\,(n+[\mathit{l}*\mathit{f_i}])}}$ of the model set of band representatives corresponding to the time sample $[l * f_i]$, where $[x]$ stands for the integer part of the real number x.

Then, we have the following.

$\mathbf{A_2}$) If the number of common elements between $\mathbf{V}_2$ and $\mathbf{U_{2,\,(n+[\mathit{l}*\mathit{f_i}])}}$ is less than $0.53 * L$, then we do not continue the comparison, and we restart the comparison of vector $\mathbf{V}_1$ with the next $\mathbf{U_{1,\,n}}$ band representative vector of the model set $\mathbf{U_{1,\,(n+1)}}$, just as in $\mathbf{A_1}$.

$\mathbf{B_2}$) If the number of common elements between $\mathbf{V}_2$ and $\mathbf{U_{2,\,(n+[\mathit{l}*\mathit{f_i}])}}$ is greater than or equal to $0.53 * L$, then, and only then, do we proceed to the comparison between the third band representative vector $\mathbf{V}_3$ that corresponds to the shift factor f_i of the unknown recording and a vector $\mathbf{U_{3,\,(n+[2*\mathit{l}*\mathit{f_i}])}}$ of the model set of band representatives corresponding to the time sample $[2 * l * f_i]$.

•

•

•

$\mathbf{A_M}$) If the number of common elements between $\mathbf{V}_M$ and $\mathbf{U_{M,\,(n+[(M-1)*\mathit{l}*\mathit{f_i}])}}$ is less than $0.53 * L$, then we do not continue the comparison. We ignore all previous comparisons, and we restart the comparison of vector $\mathbf{V}_1$ with the next band representative vector of the model set $\mathbf{U}_{\mathbf{1},\,(\mathbf{n+1})}$, just as in $\mathbf{A_1}$.

$\mathbf{B_M}$) If the number of common elements between $\mathbf{V}_M$ and $\mathbf{U}_{\mathbf{M},\,(\mathbf{n}+[(\mathbf{M-1})*l*f_i])}$ is greater than or equal to $0.53*L$, then, and only then, do we proceed to the comparison between the mean values of common elements between all the previous pairs of band representative vectors, namely

$$(\mathbf{V_1},\, \mathbf{U_{1,\,n}}),\ \left(\mathbf{V_2},\, \mathbf{U}_{\mathbf{2},\,[\mathbf{n}+l*f_i]}\right),\ \cdots$$
$$\left(\mathbf{V_M},\ \mathbf{U}_{\mathbf{M},\,[\mathbf{n}+(\mathbf{M-1})*l*f_i]}\right).$$

$\mathbf{C}$) If this mean value is greater than or equal to $0.72*L$, then the matching criterion is satisfied. Otherwise, if the mean value is smaller than $0.72*L$, then the matching criterion is not satisfied. We consider that no matching exists, and we restart the comparison of vector $\mathbf{V}_1$ with the next band representative vector of the model set $\mathbf{U}_{\mathbf{1},\,(\mathbf{n+1})}$, just as in $\mathbf{A_1}$.

Notice that $\mathbf{U_{i,\,n}}$ can be identical to $\mathbf{U}_{\mathbf{i},\,(\mathbf{n+1})}$, $i = 1,\, 2,\, \cdots,\, M$, in which case, no comparison is performed at the specific stage, but instead, the value of the previous comparison is used.

If all comparisons prove to be successful, then the current value of the shift factor, say f_e, is stored, and the system proceeds to the final criterion that will be described in the following subsection. The value of f_e constitutes an estimation of the difference in playing speed between the unknown musical recording and the current model musical recording.

Otherwise, if all the model band representative sets have been compared with the band representative vectors of the unknown musical recording corresponding to the shift factor f_i without success, then the comparisons restart in order to examine the band representative vectors of the unknown recording corresponding to the next shift factor f_{i+1}.

The whole procedure stops when all band representative sets of the unknown musical recording corresponding to all the factors f_i, $i = 0,\, 1,\, \cdots,\, S$ have been compared with all model representative vectors of the database unsuccessfully, in which case, the system decides that the part of the unknown musical recording does not correspond to any of the model musical recordings of the database. The state diagram shown in Fig. 3 represents the above algorithm.

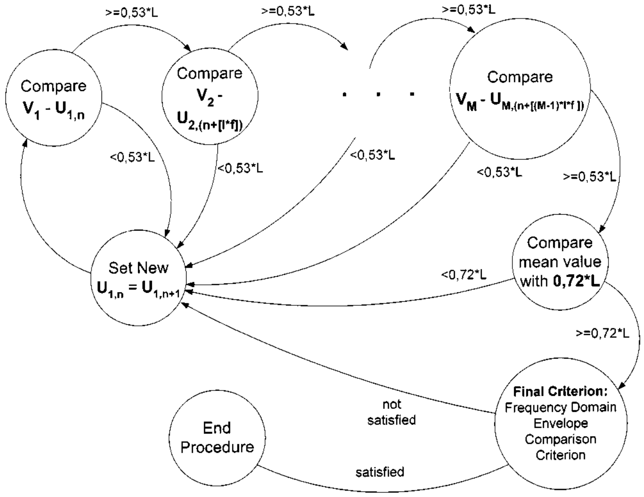


Fig. 3. State diagram for the pattern matching algorithm for a single shift factor f.

IX. Final Stage Pattern Matching Algorithm—Envelope Matching in the Frequency Domain

An additional final criterion for the identification of a musical recording suffering from an up to high-frequency-speed distortion is the similarity of its frequency domain stretched envelope with the corresponding envelope of its counterpart model musical recording. To be specific, suppose that we pick a part of such an unknown signal, e.g., received by radio or TV, of duration, say, T_R s. Suppose, moreover, that we have applied the first criterion to this signal part and that we have obtained a positive outcome for a part of a model musical recording. Thus, we want to confirm if these two parts correspond to the same musical recording. In order to accomplish this, we do the following.

1) We apply the WL sample DFT transform on the corresponding parts of the two signals. Proper WL values will be given below.
2) We create *a first frequency domain envelope* of the halves of the two DFTs by finding all maxima of the DFT for both signal parts and by interpolating them linearly.
3) If WL is greater than 32 000 samples, then experiments show that these two envelopes follow the two signals very closely: a fact that is not satisfactory for the present application. Therefore, we repeat step 2) of the above procedure in the sense that we find all maxima of the first frequency domain envelope of both signal parts, and we interpolate them linearly, thus obtaining a "*second frequency domain envelope*" of each signal part.
4) If WL is greater than 64 000 samples, then we repeat step 2) of this procedure once more, and we get a "*third frequency domain envelope*" for both halves of the WL sample DFTs. In the subsequent analysis, we will refer to them by the name "*frequency domain envelopes*."
5) We compute the integral of these two frequency domain envelopes, and we normalize them by dividing all their values by the corresponding integral. Afterwards, we multiply the position of each point of the envelope that corresponds to the unknown musical recording with the shift factor f_e, which has been estimated in the first matching criterion. In this way, we virtually equalize the playing speed of the unknown musical recording with the one of the current model musical recording.
6) Finally, we calculate the difference between the absolute values of the two normalized frequency domain envelopes. It has been observed that if the two WL sample DFTs correspond to the same part of a musical recording, then the difference between the absolute values of the their half frequency domain envelopes has a value smaller than a specific corresponding threshold. For example, if $WL = 2^{17}$ samples, then the threshold value is 1.05.

Concluding, we can say that the value of the difference between the absolute values of the half frequency domain envelopes of two signal parts (one model and one suffered an up to high frequency-speed distortion) is another criterion for determining whether or not these two parts correspond to the same musical recording. An example of the aforementioned similarities between the frequency domain envelopes of two signal parts, corresponding to exactly the same piece of music, is given in Fig. 3.

In addition, if the two signal parts correspond to the same interval of a musical recording, then the difference between the absolute values of their half frequency domain envelopes is a measure of the degree of distortion of the unknown signal.

X. Application of the Methodology—The Developed System

On the basis of the above methodology, a system has been developed that performs the following.

1) It picks a part of an unknown signal (e.g., radio or TV signal) with length $BL = 265\,000$ samples or, equivalently, of a duration of approximately 12 s, at random. Next, by applying to that part the procedure described in the previous section, a set of S groups of 11 vectors is obtained, where each vector consists of $L = 18$ band representatives. Each vector of the unstretched set, corresponding to the stretch factor $f_0 = 1$, has been calculated at a sample of the time domain having a distance of 22 000 samples from its subsequent vector or/and its previous one. Although the system works perfectly well for great many values of S and f_i, where the only limitation is the necessary processing time, we have chosen $S = 7$, and

$$f_i = \begin{cases} 1 + \left(\dfrac{i+1}{2}\right) * 0.075, & \text{if } i \text{ odd} \\ 1, & \text{if } i = 0 \\ 1 - (i/2) * 0.075, & \text{if } i \text{ even.} \end{cases}$$

2) For each stretch factor f_i, we check to see if there is a model set of band representative vectors having a distance in the time domain of $[22\,000 * f_i]$ samples that match with the corresponding 11-band representative vectors of the unknown. If the musical recording that corresponds to the unknown signal exists in our database, then some band representative vectors of this model musical recording satisfy this first matching criterion for a specific value of the shift factor f_i. In the case that there are more than one musical recordings that satisfy the above criterion, the final stage criterion is applied to define the one that corresponds to the unknown signal on a FFT window of length $WL = 2^{17}$ samples. If the least squares difference between the absolute value of the specific FFT and of the model musical recording in hand is less than 1.05, then the system decides that the particular unknown recording has been identified. When both criteria are satisfied, the system offers the exact matching samples in both the model musical recording and the unknown one.

XI. Experimental Results

On the basis of the methodology introduced in this paper, we have developed a system for automatic recognition of radio obtained musical recordings.

We have tested this system in connection with 458 CD musical recordings (whose type and composers are referred to in Table III) and 920 musical recordings received from a variety of radio stations, where most have suffered from an up to high-frequency-speed distortion.

In each one of the unknown signals, we have applied the following "pseudosampling" process, thousands of times.

We select a $BL = 265\,000$ sample part of this signal, corresponding to approximately 12 s time duration. The beginning of this frame is randomly chosen by a random numbers generator in order to imitate an actual random sampling process. To each such BL sample signal part, we apply the aforementioned procedure and then the first matching criterion, and if this is satisfied, we apply the final stage criterion of the FFT envelopes.

For each unknown signal, we repeat this pseudosampling process thousands of times until the following inequality is satisfied:

$$\frac{\text{(Number of samples of unknown composition)}}{\text{(Number of random pseudosamplings)}} < 1000.$$

For example, in an unknown musical recording that is 3 min long, the pseudosampling and the related automatic recognition procedure are performed more than 3000 times.

From the 920 musical recordings received from a variety of radio stations, 813 of them have their model counterpart recordings included in the system. The remaining 107 unknown musical recordings were correctly rejected by the system for 100% of the thousands of pseudosamplings performed on them.

The developed system recognized 740 musical recordings with a 100% success rate for all the thousands random pseudosamplings performed in each recording.

The other 73 radio received musical recordings had suffered from even more serious distortion than the one described in the introduction. However, the developed system still succeeds in recognizing them with a rate varying from 25–98% of the performed pseudosamplings with an average approximately 35%, according to the degree and type of distortion that each of these musical recordings suffered.

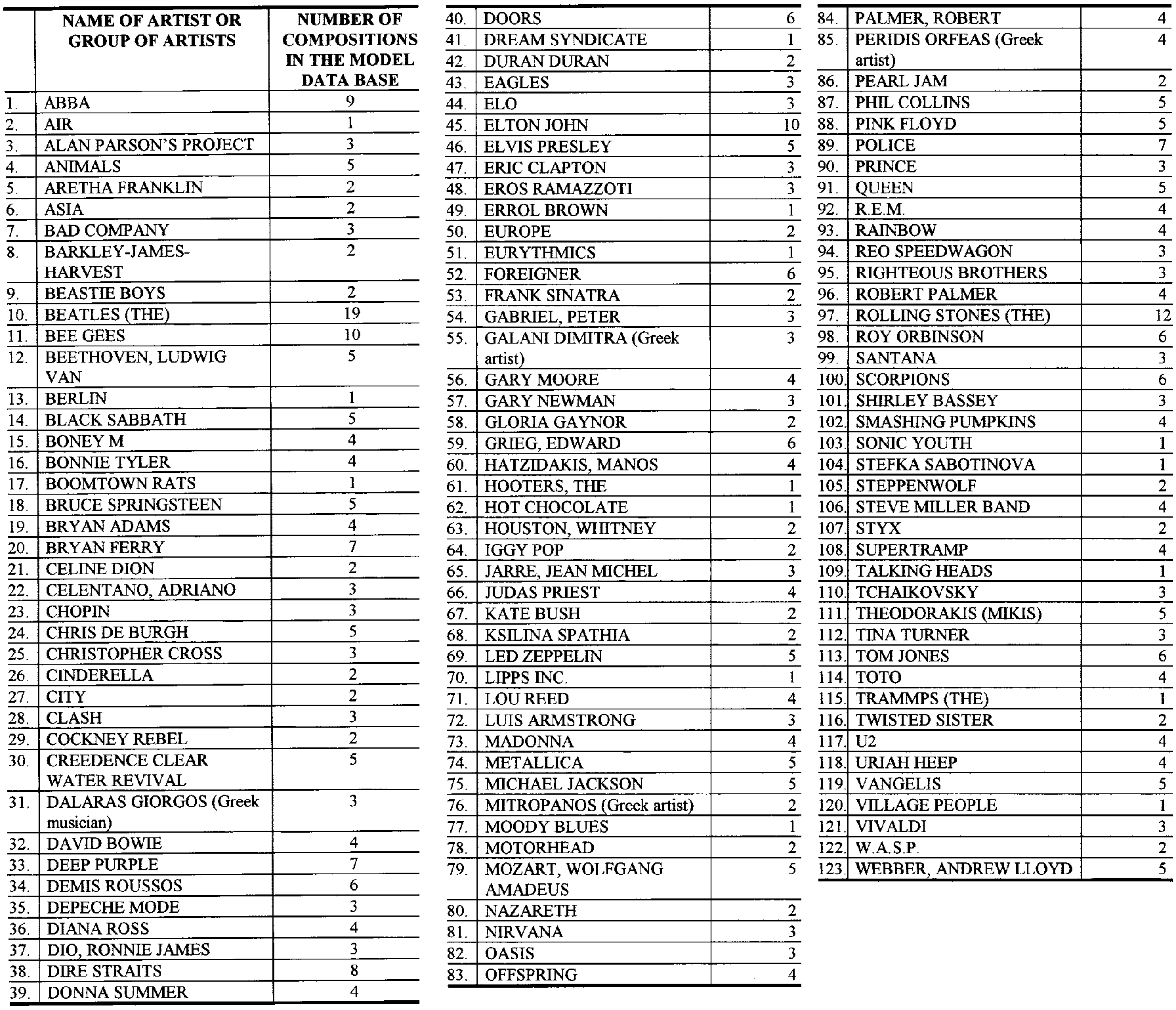
Therefore, the system manifests an overall successful recognition percentage of 94% as follows:

$$P_{ov} = \frac{(813 - 73) * 1 + 73 * 0,\ 35}{813} \approx 0,\ 94.$$

Thus, one can safely state that the introduced methodology and the related system offer automatic recognition of musical recordings with a total success rate of more than 94%. Notice that this percentage is independent of the fact of whether or not the musical recording has been dynamically compressed.

Extensive research for the improvement of recognition rate is currently being carried out, and the results will be presented in forthcoming publications.

TABLE III
TOTAL NUMBER OF 458 MUSICAL COMPOSITIONS INCLUDED IN THE DATABASE PER ARTIST/GROUP



	NAME OF ARTIST OR GROUP OF ARTISTS	NUMBER OF COMPOSITIONS IN THE MODEL DATA BASE
1.	ABBA	9
2.	AIR	1
3.	ALAN PARSON'S PROJECT	3
4.	ANIMALS	5
5.	ARETHA FRANKLIN	2
6.	ASIA	2
7.	BAD COMPANY	3
8.	BARKLEY-JAMES-HARVEST	2
9.	BEASTIE BOYS	2
10.	BEATLES (THE)	19
11.	BEE GEES	10
12.	BEETHOVEN, LUDWIG VAN	5
13.	BERLIN	1
14.	BLACK SABBATH	5
15.	BONEY M	4
16.	BONNIE TYLER	4
17.	BOOMTOWN RATS	1
18.	BRUCE SPRINGSTEEN	5
19.	BRYAN ADAMS	4
20.	BRYAN FERRY	7
21.	CELINE DION	2
22.	CELENTANO, ADRIANO	3
23.	CHOPIN	3
24.	CHRIS DE BURGH	5
25.	CHRISTOPHER CROSS	3
26.	CINDERELLA	2
27.	CITY	2
28.	CLASH	3
29.	COCKNEY REBEL	2
30.	CREEDENCE CLEAR WATER REVIVAL	5
31.	DALARAS GIORGOS (Greek musician)	3
32.	DAVID BOWIE	4
33.	DEEP PURPLE	7
34.	DEMIS ROUSSOS	6
35.	DEPECHE MODE	3
36.	DIANA ROSS	4
37.	DIO, RONNIE JAMES	3
38.	DIRE STRAITS	8
39.	DONNA SUMMER	4
40.	DOORS	6
41.	DREAM SYNDICATE	1
42.	DURAN DURAN	2
43.	EAGLES	3
44.	ELO	3
45.	ELTON JOHN	10
46.	ELVIS PRESLEY	5
47.	ERIC CLAPTON	3
48.	EROS RAMAZZOTI	3
49.	ERROL BROWN	1
50.	EUROPE	2
51.	EURYTHMICS	1
52.	FOREIGNER	6
53.	FRANK SINATRA	2
54.	GABRIEL, PETER	3
55.	GALANI DIMITRA (Greek artist)	3
56.	GARY MOORE	4
57.	GARY NEWMAN	3
58.	GLORIA GAYNOR	2
59.	GRIEG, EDWARD	6
60.	HATZIDAKIS, MANOS	4
61.	HOOTERS, THE	1
62.	HOT CHOCOLATE	1
63.	HOUSTON, WHITNEY	2
64.	IGGY POP	2
65.	JARRE, JEAN MICHEL	3
66.	JUDAS PRIEST	4
67.	KATE BUSH	2
68.	KSILINA SPATHIA	2
69.	LED ZEPPELIN	5
70.	LIPPS INC.	1
71.	LOU REED	4
72.	LUIS ARMSTRONG	3
73.	MADONNA	4
74.	METALLICA	5
75.	MICHAEL JACKSON	5
76.	MITROPANOS (Greek artist)	2
77.	MOODY BLUES	1
78.	MOTORHEAD	2
79.	MOZART, WOLFGANG AMADEUS	5
80.	NAZARETH	2
81.	NIRVANA	3
82.	OASIS	3
83.	OFFSPRING	4
84.	PALMER, ROBERT	4
85.	PERIDIS ORFEAS (Greek artist)	4
86.	PEARL JAM	2
87.	PHIL COLLINS	5
88.	PINK FLOYD	5
89.	POLICE	7
90.	PRINCE	3
91.	QUEEN	5
92.	R.E.M.	4
93.	RAINBOW	4
94.	REO SPEEDWAGON	3
95.	RIGHTEOUS BROTHERS	3
96.	ROBERT PALMER	4
97.	ROLLING STONES (THE)	12
98.	ROY ORBINSON	6
99.	SANTANA	3
100.	SCORPIONS	6
101.	SHIRLEY BASSEY	3
102.	SMASHING PUMPKINS	4
103.	SONIC YOUTH	1
104.	STEFKA SABOTINOVA	1
105.	STEPPENWOLF	2
106.	STEVE MILLER BAND	4
107.	STYX	2
108.	SUPERTRAMP	4
109.	TALKING HEADS	1
110.	TCHAIKOVSKY	3
111.	THEODORAKIS (MIKIS)	5
112.	TINA TURNER	3
113.	TOM JONES	6
114.	TOTO	4
115.	TRAMMPS (THE)	1
116.	TWISTED SISTER	2
117.	U2	4
118.	URIAH HEEP	4
119.	VANGELIS	5
120.	VILLAGE PEOPLE	1
121.	VIVALDI	3
122.	W.A.S.P.	2
123.	WEBBER, ANDREW LLOYD	5

XII. TIME REQUIRED FOR AUTOMATIC RECOGNITION

Suppose that a part of a musical recording has been obtained from the radio and stored in a ".wav" format file. It is difficult to give an exact estimate of the time required for the automatic recognition of the radio signal since this time depends on the exact size of the database of musical characteristics, as well as on the peculiarities of the radio signal in hand. However, extended experiments show that when the underlying hardware is a Pentium III at 500 MHz, with 256 MB RAM and the operating system is Red Hat LINUX, then a typical maximum time required for deciding if the radio signal in hand corresponds to a specific CD musical recording whose musical characteristics are stored in the system database, or not, is a bit less than 7 min for a database consisting of 458 model musical recordings.

The average time required for deciding if the radio signal in hand corresponds to a specific CD musical recording of the system data base or not is approximately 3.5 min for the same database consisting of 458 model musical recordings.

Notice that the whole system has been developed in such a way that it is parallelizable. In this way, if one uses K processors, then the time required for the automatic recognition of a radio signal when the model database consists of NS songs is approximately divided by the number of processors K. Therefore, one can achieve real-time automatic recognition, even when the database contains many thousands of musical recordings, where the only restriction is the hardware availability.

XIII. TIME REQUIRED FOR FEATURE EXTRACTION FROM CD OBTAINED MUSICAL RECORDINGS

The necessary processing time for extracting the set of musical characteristics from a single model musical recording depends highly on the exact size of the specific recording. Experiments show that when the aforementioned hardware and operating system are used, then for a model musical recording of 3.5 min in duration, the necessary time is approximately 2.5 h. However, we must stress that this procedure takes place only once for each model musical recording so that the obtained file of its musical characteristics is inserted into the database. Notice that this procedure is also parallelizable.

XIV. Conclusion

In this paper, a methodology and a related system for the automatic recognition of musical recordings is presented. The system comprises a database of characteristics extracted from each model musical recording by means of novel feature extraction algorithms. Criteria for the comparison of an unknown musical recording with the model ones are introduced. The system was tested with extended experiments, which have demonstrated that it offers more than 94% recognition rate, even for musical recordings that have suffered from up to high-frequency-speed distortion. It is parallelizable, and it can operate essentially in real time for many thousands of recordings, according to the underlying hardware.

Appendix A

On each selected signal, after calculating the absolute values of the DFT, spotting the peaks, and sorting them according to their amplitudes, we employ the spreading function of masking [1]–[3], [9], [13] as follows.

We use the formula

$$F(z) = 15.81 + 7.5 * (z + 0.474) - 17.5 * \sqrt{1 + (z + 0.474)^2}$$

where

$$z = 13 * \arctan(0.000\,76(\acute{\imath} - \acute{\imath}_0)) + 3.5 * \arctan\left(\left(\frac{\acute{\imath} - \acute{\imath}_0}{7500}\right)^2\right)$$

$\acute{\imath}_0$ is the masker frequency, and $\acute{\imath}$ is the variable frequency (both with values in Hertz). We select the highest amplitude peak $\acute{\imath}$. If the amplitude of a peak is smaller than the value of $F(\acute{\imath})$, then we remove this peak from the "list" of the sorted peaks. We continue by applying the same procedure to the next remaining peak until all peaks are exhausted.

It is clear that the spreading function is limited between the frequencies $\acute{\imath}_0 - W$ and $\acute{\imath}_0 + W$, where $\acute{\imath}_0$ is the masker frequency. Our extended experiments indicate that the value of W must lie in the range of 27 to 67 Hz, depending on the quality of the radio/TV obtained signal. The choice of an appropriate W value depends on the degree of distortion of the radio/TV obtained signal. The smaller the value of W, the greater the discrimination capability, but, in addition, the greater the difficulty in recognizing highly distorted signals.

Appendix B

Let us suppose that we have computed the FFT of a signal $x[n]$ of N samples, starting at sample α of the time domain and ending at sample $(\alpha + N - 1)$. Next, suppose that we want to calculate the N-sample FFT of the same signal $x[n]$ starting at sample $(\alpha+1)$ of the time domain and ending at sample $(\alpha+N)$. This second FFT calculation can be performed adaptively, i.e., by taking into account the information of the first FFT, as it is described below [24]–[27].

Notice, at first, that the N-sample DFT of the $x[n]$, starting at sample α and ending at $(\alpha + N - 1)$, is given by

$$X[k] = \sum_{n=0}^{N-1} x[n + \alpha] W^{kn}, \qquad \text{where } W = e^{-j(2\pi/N)}$$

whereas the N-sample DFT of the $x[n]$, starting at sample $(\alpha + 1)$ and ending at $(\alpha + N)$, is given by

$$X_s[k] = \sum_{n=0}^{N-1} x[n + \alpha + 1] W^{kn}.$$

Thus

$$\begin{aligned} X_s[k] = &(-x[\alpha] + x[\alpha]) * W^{-k} \\ &+ \left(\sum_{n=0}^{N-2} x[n + \alpha + 1] W^{k(n+1)}\right) * W^{-k} \\ &+ x_s[\alpha + N] W^{k(N-1)}. \end{aligned}$$

In the last summation, we make the substitution of the dummy variable

$$i = n + 1$$

which implies that

$$\begin{aligned} X_s[k] = &-x[\alpha] * W^{-k} + x[\alpha] * W^{-k} \\ &+ \left(\sum_{i=1}^{N-1} x[i + \alpha] W^{ki}\right) * W^{-k} \\ &+ x_s[\alpha + N] W^{k(N-1)} \Leftrightarrow \\ X_s[k] = &-x[\alpha] * W^{-k} + \left(\sum_{i=0}^{N-1} x[i + \alpha] W^{ki}\right) * W^{-k} \\ &+ x_s[\alpha + N] W^{k(N-1)}. \end{aligned}$$

Therefore, we finally obtain that

$$X_s[k] = -x[\alpha] * W^{-k} + X[k] * W^{-k} + x_s[\alpha + N] W^{k(N-1)}$$

namely, the sought-for recursive-adaptive FFT computation.

Notice that the computational complexity of a standard FFT is $(N/2) * \log_2 N$ complex multiplications, i.e., $2N * \log_2 N$ simple real number multiplications, whereas the adaptive FFT computation algorithm, presented here, requires $8N$ real multiplications. Therefore, the latter algorithm is many times faster than the standard FFT, according to the window length N, for N greater than 16 samples. In our case, it is desirable to have as high a resolution and accuracy as possible, where the only restriction is the limited processing time. Therefore, we found that the FFT window length of values $N = 8*1024$, $N = 16*1024$, and $N = 32 * 1024$ offers very good results. It is clear that for these values of N, the adaptive FFT algorithm presented here requires a number of multiplications from 3.25 to 3.75 times smaller than the one of the standard FFT. Considering the memory allocations as well as the additions involved, one can safely state that the presented adaptive FFT method calculates the set of L-element vectors of the whole musical recording at least four and a half times faster than the classical FFT method. To set ideas, in order to obtain this set of vectors for a musical recording that is 3.5 min long sampled at a rate of 22 050 samples/s, the time required is at least 7 h if the standard FFT algorithm is used, while it is less than 90 min when the presented adaptive FFT computation is applied.

APPENDIX C

Key in Music: A closed system of functionally related chords generated by certain tonal conventions associated with the western concept of diatronic major and minor scales. The broader term tonality is sometimes used as a synonym for key.

Scale: A pattern of pitch relationships.

Tempo or Rhythm: A concept that embraces all durational aspects of music. The specific occurrence of notes in musical time is determined by rhythm.

Monophonic Music: Usually describes music for a single voice or part, for example, playing song and unaccompanied solo song. However, some tend to consider monophonic music to be a sequence of simple notes produced by a single instrument, and we adopt this notation throughout the paper.

Polyphonic Music: Music in more than one part, music in many parts, and the style in which all or several of the musical parts move to some extent independently. However, some tend to consider polyphonic music to be an arbitrarily complicated melody produced by a variety of instruments and/or voices, and we adopt this notation throughout the paper.

Instrumental Music: Music performed using one or more musical instruments without the presence of voice.

Song with Lyrics: A piece of music performed by a voice, with or without instrumental accompaniment.

Recognition of an Unknown Musical Recording: Essentially a matching procedure between an unknown recording and a known one, based on common characteristics and leading to the identification of the unknown musical recording.

REFERENCES

[1] M. Bosi *et al.*, "ISO/IEC MPEG-2 Advanced audio coding," *J. Audio Eng. Soc.*, vol. 10, pp. 789–813, 1997.

[2] K. Brandenburg and G. Stoll, "ISO-MPEG-1 Audio: A generic standard for coding of high quality digital audio," *J. Audio Eng. Soc.*, vol. 42, no. 10, pp. 780–792, 1994.

[3] P. Davis, "A tutorial on MPEG/Audio compression," *IEEE Multimedia J.*, Summer 1995.

[4] R. J. McNab, L. A. Smith, I. H. Witten, C. L. Henderson, and S. J. Cunningham, "Toward the digital music library: Tune retrieval from acoustic input," *Proc. ACM Digital Libraries*, pp. 11–18, 1996.

[5] R. J. McNab, L. A. Smith, D. Bainbridge, and I. H. Witten, "The New Zealand digital library MELody inDEX," *D-Lib Mag.*, 1997.

[6] M. Mongeau and D. Sankoff, "Comparison of musical sequences," *Comput. Humanities*, vol. 24, pp. 161–175, 1990.

[7] A. Pikrakis, S. Theodoridis, and D. Kamarotos, "Recognition of isolated musical patterns in the context of Greek traditional music," in *Proc. Int. Conf. Electron. Circuits Syst. (ICECS)*, 1997.

[8] ——, "Recognition of isolated musical patterns using discrete observation hidden Markov models," in *Proc. Eur. Signal Process. Conf.*, 1998.

[9] J. H. Rothweiler, "Polyphase quadrature filters—A new subband coding technique," in *Proc. Int. IEEE Acoust., Speech, Signal Process. Conf.*, vol. 27.2, Boston, MA, 1983, pp. 1280–1283.

[10] C. J. Stevens and C. R. Latimer, "A comparison of connectionist models of music recognition and human performance," *Minds Mach.*, vol. 2, pp. 279–400, 1992.

[11] ——, "Recognition of short tonal compositions by connectionist models and listeners: Effects of feature manipulation and training," *Musikometrika*, vol. 5, pp. 197–224, 1993.

[12] ——, "Music recognition: An illustrative application of a connectionist model," *Psychol. Music*, vol. 25, pp. 161–185, 1997.

[13] E. Zwicker and H. Fastl, *Psychoacoust.*. Berlin, Germany: Springer-Verlag, 1990.

[14] S. Handel, *Listening: An Introduction to the Perception of Auditory Events*. Cambridge, MA: MIT Press, 1989.

[15] B. C. J. Moore, *An Introduction to the Psychology of Hearing*. New York: Academic, 1997.

[16] P. M. Todd and D. G. Loy, *Music and Connectionism*. Cambridge, MA: MIT Press, 1991.

[17] M. C. Mozer and T. Soukup, "Connectionist music composition based on melodic and stylistic constraints," in *Advances in Neural Information Processing Systems 3*. San Francisco, CA: Morgan Kaufmann, 1991, pp. 789–796.

[18] A. Weigend, "Connectionism for music and audition," in *Advances in Neural Information Processing Systems 6*. San Francisco, CA: Morgan Kaufmann, 1994, pp. 1163–1164.

[19] A. Robel, "Neural network modeling of speech and music signals," in *Advances in Neural Information Processing Systems 9*. Cambridge, MA: MIT Press, 1997, p. 779.

[20] R. O. Duda, "Connectionist models for auditory scene analysis," in *Advances in Neural Information Processing Systems 6*. San Francisco, CA: Morgan Kaufmann, 1994, pp. 1069–1076.

[21] E. D. Scheirer and M. Slaney, "Construction and evaluation of a robust multifeature speech/music discriminator," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Munich, Germany, 1997.

[22] B. C. J. Moore, "Frequency difference limens for short-duration tones," *J. Acoust. Soc. Amer.*, vol. 54, pp. 610–619, 1973.

[23] C. C. Wier, W. Jesteadt, and D. M. Green, "Frequency discrimination as a function of frequency and sensation level," *J. Acoust. Soc. Amer.*, vol. 61, pp. 178–184, 1977.

[24] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[25] M. E. Frerking, *Digital Signal Processing in Communication Systems*. Boston, MA: Kluwer, 1994.

[26] M. R. Portnoff, "Implementation of the digital phase vocoder using the fast Fourier transform," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 243–248, 1976.

[27] J. B. Allen and L. R. Rabiner, "A unified approach to short-time Fourier analysis and synthesis," *Proc. IEEE*, vol. 65, pp. 1558–1564, 1977.

Dimitrios Fragoulis was born in Athens, Greece, in 1973. He received the Diploma and M.Sc. degrees in electrical and computer engineering from National Technical University of Athens in 1996. He is currently pursuing the Ph.D. degree in computer engineering at the same university.

His research interests and recent work are in music and speech processing and automatic recognition, study of psychological and perceptual aspects of sound, etc. He has four publications in international journals and eight publications in international conferences on these subjects.

George Rousopoulos was born in Athens, Greece, in 1971. He received the diploma in computer and software engineering from the Technical University of Patras, Patras, Greece, in 1994. He received the Ph.D. degree in computer engineering from the National Technical University of Athens in 2000.

His research interests and recent work are in music and speech processing and automatic recognition, image processing, pattern recognition, algorithm robustness, algorithms for echo cancellation, etc. He has six publications in international journals and ten publications in international conferences on these subjects.

Thanasis Panagopoulos was born in Athens, Greece, in 1973. He received the diploma and M.Sc. degrees in electrical and computer engineering from National Technical University of Athens in 1996. He is currently pursuing the Ph.D. degree in computer engineering at the same university.

His research interests and recent work are in music and speech processing and automatic recognition, image processing, pattern recognition, algorithms for echo cancellation, etc. He has three publications in international journals and five publications in international conferences on these subjects.

Constantin Alexiou was born in Igoumenitsa, Greece, in 1973. He received the diploma and M.Sc. degrees in electrical and computer engineering from National Technical University of Athens, Athens, Greece, in 1996. He is currently pursuing the Ph.D. degree in computer engineering at the same university.

His research interests and recent work are in music and speech processing and automatic recognition, algorithm robustness, algorithms for echo cancellation, biomedical engineering, etc. He has three publications in international journals and seven publications in international conferences on these subjects.

Constantin Papaodysseus was born in Athens, Greece. He received the diploma in electrical and computer engineering from National Technical University of Athens (NTUA) and the M.Sc. degree from Manchester University, Manchester, U.K. He received the Ph.D. degree in computer engineering from NTUA.

Since 1996, he has been an Associate Professor with Department of Electrical and Computer Engineering, NTUA. His research interests include music and speech processing and automatic recognition, image processing, applied mathematics, algorithm robustness and quantization error analysis, adaptive algorithms, biomedical engineering, etc. He has more than 25 publications in international journals and many publications in international conferences on these subjects.