(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
30 June 2016 (30.06.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/102737 A1

(51) International Patent Classification:
G10L 15/14 (2006.01) *G06K 9/62* (2006.01)

(21) International Application Number:
PCT/FI2014/051036

(22) International Filing Date:
22 December 2014 (22.12.2014)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **NOKIA TECHNOLOGIES OY** [FI/FI]; Karaportti 3, FI-02610 Espoo (FI).

(72) Inventors: **ERONEN, Antti**; Rintamäenkatu 13 A 6, FI-33820 Tampere (FI). **LEPPÄNEN, Jussi**; Kossinkatu 1 B, FI-33580 Tampere (FI). **SAARI, Pasi**; Mesikämmen 10 C 17, FI-40400 Jyväskylä (FI). **LEHTINIEMI, Arto**; Lomarantatie 17, FI-33880 Lempäälä (FI).

(74) Agents: **NOKIA TECHNOLOGIES OY** et al.; Ari Aarnio, IPR Department, Karakaari 7, FI-02610 Espoo (FI).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *of inventorship (Rule 4.17(iv))*

Published:

— *with international search report (Art. 21(3))*

WO 2016/102737 A1

(54) Title: TAGGING AUDIO DATA

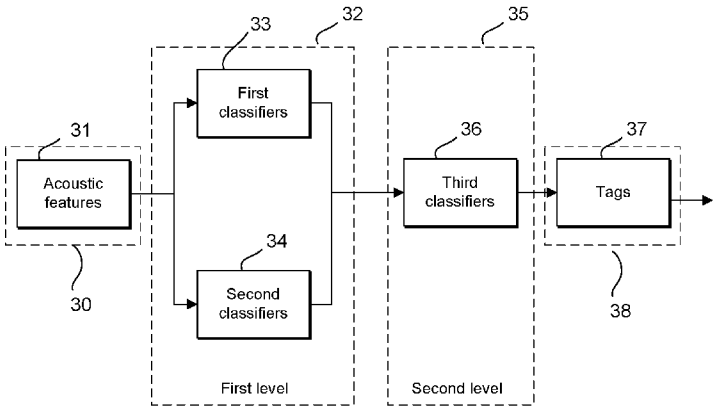

Figure 3

(57) Abstract: A method comprises determining acoustic feature(s) of audio data, generating first and second classifications based on the feature(s) using first and second classifiers respectively, generating at least one third classification based on said first and second classifications using a third classifier and storing tag(s)for said audio data based on said third classification. The first and/or third classifiers may be non-probabilistic, e.g. a support vector machine (SVM) classifier. The second classifier may be probabilistic, e.g. based on a Gaussian Mixture Model (GMM). Another method determines whether audio data matches an audio track in a catalogue, based on audio-fingerprints and/or metadata. If so, information for the audio data is obtained from the matching track. If not, then one or more acoustic features of the audio data are extracted and used to continue the search. If no match is found, then information based on the extracted features is uploaded to the catalogue.

Tagging audio data

Field

This disclosure relates to analysing audio data. In particular, this disclosure relates to determining and storing classification and/or tagging information for audio tracks, such as autotagging a piece of music.

Background

Audio content databases, streaming services, online stores and media player software applications often include genre classifications, to allow a user to search for tracks to play stream and/or download.

Some databases, services, stores and applications also include a facility for recommending music tracks to a user based on a history of music that they have accessed in conjunction with other data, such as rankings of tracks or artists from the user, history data from other users who have accessed the same or similar tracks in the user's history or otherwise have similar user profiles, metadata assigned to the tracks by experts and/or users, and so on.

Summary

According to one aspect, an apparatus includes a controller and a memory in which is stored computer-readable instructions which, when executed by the controller, cause the controller to determine one or more acoustic features of audio data, generate at least one first classification based on the one or more determined acoustic features using a respective first classifier, generate at least one second classification based on the one or more determined acoustic features using at least one respective second classifier, the second classifier being different from the first classifier, generate a third classification based on said first and second classifications using a third classifier, and store at least one tag for said audio data based on said third classification.

At least one of said first classifier and said third classifier may be a non-probabilistic classifier, such as a support vector machine (SVM) classifier.

The second classifier may be a probabilistic classifier. For example, the second classifier may be based on one or more Gaussian Mixture Models.

In one example embodiment, the first and third classifiers are non-probabilistic classifiers, while the second classifier is a probabilistic classifier.

The metadata may indicates at least one of the following characteristics of the audio data: a musical instrument included in the audio data, the presence or absence of vocals and/or a vocalist gender, presence or absence of music and a musical genre.

The one or more acoustic features may include at least one feature based on a mel-frequency cepstral coefficient (MFCC). For example, the one or more acoustic features may include a mean of a plurality of MFCCs for the audio data, a variance of such MFCCs, time derivatives of such MFCCs and so on. Other examples of acoustic features that may be extracted include a fluctuation pattern, a danceability feature, a feature relating to tempo, such as beats per minute, a chorus-related feature, a duration of a musical track, a feature relating to sound pressure level, a brightness-related feature; and a low frequency ratio related feature.

The computer-readable instructions, when executed by the controller, may cause the controller to select one or more tracks from a catalogue having one or more tags matching the at least one tag for audio data and to output information identifying said one or more selected tracks. For example, information regarding the selected tracks may be presented as a recommendation of tracks based on a similarity to the audio data.

The computer-readable instructions, when executed by the controller, may cause the controller to store one or more of at least one of said first and second classifications and at least one of said one or more acoustic features of the audio data. Such data may be stored in a database in the apparatus, or in a storage that is accessible by the apparatus via a network.

According to a second aspect, a method includes determining one or more acoustic features of audio data, generating at least one first classification based on the one or more determined acoustic features using a respective first classifier, generating at least one second classification based on the one or more determined acoustic features using at least one respective second classifier, the second classifier being different from the first classifier, generating a third classification based on said

first and second classifications using a third classifier, and storing at least one tag for said audio data based on said third classifier.

This second aspect may also provide a computer program including computer readable instructions which, when executed by a processor, causes the processor to perform such a method.

The method may also include selecting one or more tracks from a catalogue having similar metadata to said metadata of the audio data and outputting information identifying said one or more selected tracks.

According to a third aspect, a non-transitory tangible computer program product includes computer readable instructions which, when executed by a processor, causes the processor to determine one or more acoustic features of audio data, generate at least one first classification based on the one or more determined acoustic features using a respective first classifier, generate at least one second classification based on the one or more determined acoustic features using at least one respective second classifier, the second classifier being different from the first classifier, generate a third classification based on said first and second classifications using a third classifier, and store at least one tag for said audio data based on said third classifier.

According to a fourth aspect, an apparatus includes a feature extractor to determine one or more acoustic features of audio data, at least one first classifier to generate at least one first classification based on the one or more determined acoustic features, at least one second classifier to determine at least one second classification based on the one or more determined acoustic features, the second classifier being different from the first classifier, a third classifier configured to generate a third classification based on said first and second classifications using a third classifier and a tagging module to store at least one tag for said audio data based on said third classification.

In one example, the apparatus includes a controller configured to provide the feature extractor, the first classifier, the second classifier, the third classifier and the tagging module. For example, the apparatus may be a server and the controller may be a processing arrangement configured to execute a computer program to

provide the feature extractor, the first, second and third classifiers and the tagging module.

According to a fifth aspect, an apparatus includes a controller and a memory in which is stored computer-readable instructions which, when executed by the controller, cause the controller to determine whether audio data matches an audio track in a catalogue of audio tracks, based on at least one of an audio-fingerprint and metadata of the audio data, if a match is determined, retrieve at least one tag for the matching audio track from the catalogue and store at least one tag for the audio data corresponding to the retrieved at least one tag and, if a match is not determined, then extract one or more acoustic features of the audio data, determine whether the audio data matches an audio track in the catalogue of audio tracks, based on said one or more acoustic features and if a match is determined then retrieve at least one tag for the matching audio track from the catalogue and store at least one tag for the audio track corresponding to the retrieved at least one tag and if a match is not determined then upload to the catalogue at least one tag based on the extracted features of the audio track.

In some embodiments, the audio-fingerprint is based on an audio waveform of at least part of the audio data. For example, the audio-fingerprint may be a feature vector containing information sufficient for identifying such an audio waveform.

The extraction of the one or more acoustic features may include extracting a first subset of one or more acoustic features, where if a match is not determined based on said one or more acoustic features, the extraction of one or more acoustic features and determining whether the audio data matches an audio track in the catalogue based on the one or more acoustic features is repeated for at least one further subset of one or more acoustic features. For example, the first subset may include one or more acoustic features that are computationally lighter than the one or more acoustic features of the at least one further subset. In another example, the at least one further subset may include one or more acoustic features computed based on one or more acoustic features of the first subset.

The audio data may include a music track, in which case the at least one tag may indicate an instrument included in said music track and/or a genre of said music track.

The computer-readable instructions, when executed by the controller, may cause the controller to select one or more tracks from a catalogue having one or more tags matching the at least one tag for audio data and to output information identifying said one or more selected tracks.

The one of more acoustic features may include one or more of mel-frequency cepstral coefficients, a fluctuation pattern feature, beat tracking features, an accent feature, an energy feature, second phase, non-causal beat tracking features, danceability and club-likeness features, a chorus-related feature, a classification of the audio data as being an instrumental or vocal track, a vocalist gender classification, a tag or classification indicating a musical instrument and a tag or classification indicating a musical genre.

The computer-readable instructions, when executed by the controller, cause the controller to generate, for the audio data, at least one of at least one mel-frequency cepstral coefficient, a tag or classification indicating a musical instrument and a tag or classification indicating a musical genre.

The computer-readable instructions, when executed by the controller, may cause the controller to generate a first classification indicating a musical instrument or genre based on the one or more extracted acoustic features using a respective first classifier;
generate at least one second classification based on the one or more extracted acoustic features using at least one respective second classifier, the second classifier being different from the first classifier; and generate a third classification based on said first and second classifications using a third classifier, where the at least one tag for said audio data is based on said third classification.

According to a sixth aspect, a method includes determining whether audio data matches an audio track in a catalogue of audio tracks, based on at least one of an audio-fingerprint and metadata of the audio data, if a match is determined, retrieving at least one tag for the matching audio track from the catalogue and storing at least one tag for the audio data corresponding to the retrieved at least one tag, and, if a match is not determined, then extracting one or more acoustic features of the audio data, determining whether the audio data matches an audio track in the catalogue of audio tracks, based on said one or more acoustic features, and if a match is determined then retrieving at least one tag for the matching audio

track from the catalogue and storing at least one tag for the audio data corresponding to the retrieved at least one tag and if a match is not determined then uploading to the catalogue at least one tag based on the extracted features of the audio track.

The sixth aspect may also provide a computer program comprising computer readable instructions which, when executed by a processing arrangement, causes the processing arrangement to perform a method according to such a method.

According to a seventh aspect, a non-transitory tangible computer program product includes computer readable instructions which, when executed by a processing arrangement, causes the processing arrangement to determine whether audio data matches an audio track in a catalogue of audio tracks, based on at least one of an audio-fingerprint and metadata of the audio data, if a match is determined, retrieve at least one tag for the matching audio track from the catalogue and store at least one tag for the audio data corresponding to the retrieved at least one tag, and if a match is not determined, then extract one or more acoustic features of the audio data, determine whether the audio data matches an audio track in the catalogue of audio tracks, based on said one or more acoustic features and if a match is determined then retrieve metadata for the matching audio track from the catalogue and store at least one tag for the audio data corresponding to the retrieved at least one tag and if a match is not determined, then upload to the catalogue at least one tag based on the extracted features of the audio track.

The computer readable instructions, when executed by the processing arrangement, may further cause the processor to generate a first classification indicating a musical instrument or genre based on the one or more extracted acoustic features using a respective first classifier, generate at least one second classification based on the one or more extracted acoustic features using at least one respective second classifier, the second classifier being different from the first classifier, and generate a third classification based on said first and second classifications using a third classifier, where the at least one tag for said audio data is based on said third classification.

According to an eighth aspect, an apparatus includes a track matcher to determine whether audio data matches an audio track in a catalogue of audio tracks based on at least one of an audio-fingerprint and metadata of the audio data, a feature

extractor to extract acoustic features from the audio data, a data retriever to retrieve at least one tag for the matching audio track from the catalogue and store at least one tag for the audio data corresponding to the retrieved at least one tag if a match is determined, and a tagging module, wherein the track matcher is configured to, if a matching audio track is not found based on the at least one of the audio-fingerprint and the metadata of the audio data, determine whether audio data matches an audio track in a catalogue of audio tracks based on said extracted acoustic features and the tagging module is configured to, if a match is not found based on the extracted acoustic features, upload to the catalogue at least one tag based on the extracted features of the audio track.

In one example, the apparatus includes a controller configured to provide the track matcher, feature extractor, data retriever and tagging module. For example, the apparatus may be a server and the controller may be a processing arrangement configured to execute a computer program to provide the track matcher, feature extractor, data retriever and tagging module.

Brief description of the drawings

Embodiments will now be described by way of non-limiting examples with reference to the accompanying drawings, of which:

Figure 1 is a schematic diagram of a system in which an embodiment may be included;

Figure 2 is a schematic diagram of components of an analysis server according to an embodiment, in the system of Figure 1;

Figure 3 is an overview of a method of determining tag information for an audio track according to an embodiment;

Figure 4 is a flowchart of a method according to Figure 3, which may be performed by the analysis server of Figure 2;

Figure 5 is a flowchart of a method of calculating mel-frequency cepstral coefficients in part of the method of Figure 4;

Figure 6 depicts an example of frame blocking and windowing in the method of Figure 5;

Figure 7 is an example of a spectrum generated by transforming a portion of a frame in the method of Figure 5;

Figure 8 depicts a bank of weighted mel-frequency filters used in the method of Figure 5;

Figure 9 depicts a spectrum of log mel-band energies in the method of Figure 5;

Figure 10 is an overview of a process for obtaining multiple types of acoustic features in the method of Figure 4;

Figure 11 shows example probability distributions for a number of first classifications;

Figure 12 shows the example probability distributions of Figure 11 after logarithmic transformation;

Figure 13 is an overview of a method according to another embodiment, which may be performed by the analysis server of Figure 2; and

Figure 14 is a flowchart of a method corresponding to the overview shown in Figure 13.

Detailed description

Embodiments described herein concern determining and storing classification information, or tags, for audio data. Embodiments of the present invention are described in the context of music, namely classifying and tagging a music track. However, other embodiments may concern determining and storing classification of other types of audio tracks, such as determining whether an audio track contains spoken word elements, or combinations of music and spoken word elements, or other sounds such as birdsong or sound effects.

Referring to Figure 1, an analysis server 100 is shown connected to a network 102, which can be any data network such as a Local Area Network (LAN), Wide Area Network (WAN) or the Internet. The analysis server 100 is configured to receive and process requests for audio content from one or more terminals 104 via the network 102.

In the present example, three terminals 104 are shown, each incorporating media playback hardware and software, such as a speaker (not shown) and/or audio output jack (not shown) and a processor (not shown) that executes a media player software application to stream and/or download audio content over the network 102 and to play audio content through the speaker. As well as audio content, the terminals 104 may be capable of streaming or downloading video content over the network 102 and presenting the video content using the speaker and a display 106. Suitable terminals 104 will be familiar to persons skilled in the art. For instance a smart phone could serve as a terminal 104 in the context of this application

although a laptop, tablet or desktop computer may be used instead. Such devices include music and video playback and data storage functionality and can be connected to the music analysis sever 100 via a cellular network, Wi-fi, Bluetooth® or any other suitable connection such as a cable or wire.

As shown in Figure 2, the analysis server 100 includes a controller 202, an input and output interface 204 configured to transmit and receive data via the network 102, a memory 206 and a mass storage device 208 for storing video and audio data.

The controller 202 is connected to each of the other components in order to control operation thereof. The controller 202 may take any suitable form. For instance, it may be a processing arrangement that includes a microcontroller, plural microcontrollers, a processor, or plural processors.

The memory 206 and mass storage device 208 may be in the form of a non-volatile memory such as read only memory (ROM) a hard disk drive (HDD) or a solid state drive (SSD). The memory 206 stores, amongst other things, an operating system 210 and at least one software application 212 to be executed by the controller 202. Random Access Memory (RAM) 214 is used by the controller 202 for the temporary storage of data.

The operating system 210 may contain code which, when executed by the controller 202 in conjunction with the RAM 214, controls operation of analysis server 100 and provides an environment in which the or each software application 212 can run.

Software application 212 is configured to control and perform audio and video information processing by the controller 202 of the analysis server 100. The operation of this software application 212 according to a first embodiment will now be described in detail, with reference to Figures 3 and 4. In the following, the accessed audio track is referred to as the input signal.

Figure 3 is an overview of a determination of tag information for the audio track by the controller 202 of the analysis server 100, in which the controller 202 acts as a feature extractor 30, first level classifiers 32, second level classifiers 33, and a tagging module 38. Acoustic features 31 of the audio are extracted and input to first level classifiers 32 to generate first level classifications for the audio track. In this example, first classifiers 33 and second classifiers 34 are used to generate first

and second classifications respectively. In the embodiments to be described below, the first classifiers 33 are non-probabilistic classifiers, while the second classifiers 34 are probabilistic classifiers.

The first and second classifications generated by the first level classifiers 32 are provided as inputs to a second level classifier 35. One or more second level classifications are generated by the second level classifier 35, based at least in part on the first and second classifications. In the embodiments to be described below, the second level classifiers 35 include a third classifier 36, which outputs a third classification.

One or more tags 37 are generated, based on the second level classifications. Such tags 37 may be stored by the tagging module 38 to characterise the audio track in a database, organise or search a database of audio tracks and/or determine a similarity between the audio track and other audio tracks, for example, to select other audio tracks for playback or purchase by a user.

The method will now be described in more detail, with reference to Figures 4 to 12.

Beginning at step s4.0 of Figure 4, if the received input signal is in a compressed format, such as MPEG-1 Audio Layer 3 (MP3), Advanced Audio Coding (AAC) and so on, the input signal is decoded into pulse code modulation (PCM) data (step s4.1). In this particular example, the samples for decoding are taken at a rate of 44.1 kHz and have a resolution of 16 bits.

Next, the software application 212 causes the controller 202 to extract acoustic features 31 or descriptors which indicate characteristics of the audio track (step s4.2). In this particular embodiment, the features 31 are based on mel-frequency cepstral coefficients (MFCCs). In other embodiments, other features such as fluctuation pattern and danceability features, beats per minute (BPM) and related features, chorus features and other features may be used instead of, or as well as MFCCs.

An example method for extracting acoustic features 31 from the input signal at step s4.2 will now be described, with reference to Figure 5.

Starting at step s5.0, the controller 202 may, optionally, resample the decoded input signal at a lower rate, such as 22050 kHz (step s5.1).

An optional "pre-emphasis" process is shown as step s5.2. Since audio signals conveying music tend to have a large proportion of their energy at low frequencies, the pre-emphasis process filters the decoded input signal to flatten the spectrum of the decoded input signal. The relatively low sensitivity of the human ear to low frequency sounds may be modelled by such flattening. One example of a suitable filter for this purpose is a first-order Finite Impulse Response (FIR) filter with a transfer function of $1-0.98z^{-1}$.

At step s5.3, the controller 202 blocks the input signal into frames. The frames may include, for example, 1024 or 2048 samples of the input signal, and the subsequent frames may be overlapping or they may be adjacent to each other according to a hop-size of, for example, 50% and 0%, respectively. In other examples, the frames may be non-adjacent so that only part of the input signal is formed into frames.

Figure 6 depicts an example in which an input signal 50 is divided into blocks to produce adjacent frames of about 30 ms in length which overlap one another by 25%. However, frames of other lengths and/or overlaps may be used.

A Hamming window, such as windows 52a to 52d, is applied to the frames at step s5.4, to reduce windowing artifacts. An enlarged portion in Figure 6 depicts a frame 54 following the application of the window 52d to the input signal 50.

At step s5.5, a Fast Fourier Transform (FFT) is applied to the windowed signal to produce a magnitude spectrum of the input signal. An example FFT spectrum is shown in Figure 7. Optionally, the FFT magnitudes may be squared to obtain a power spectrum of the signal for use in place of the magnitude spectrum in the following steps.

The spectrum produced by the FFT at step s5.5 may have a greater frequency resolution at high frequencies than is necessary, since the human auditory system is capable of better frequency resolution at lower frequencies but is capable of lower frequency resolution at higher frequencies. So, at step s5.6, the spectrum is filtered to simulate non-linear frequency resolution of the human ear.

In this example, the filtering at step s5.6 is performed using a filter bank having channels of equal bandwidths on the mel-frequency scale. The mel-frequency scaling may be achieved by setting the channel centre frequencies equidistantly on a mel-frequency scale, given by the Equation (1),

$$Mel(f) = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \quad (1)$$

where f is the frequency in Hertz.

The output of each filtered channel is a sum of the FFT frequency bins belonging to that channel, weighted by a mel-scale frequency response. The weights for filters in an example filter bank are shown in Figure 8. In Figure 8, 40 triangular-shaped bandpass filters are depicted whose center frequencies are evenly spaced on a perceptually motivated mel-frequency scale. The filters may span frequencies from 30 hz to 11025 Hz, in the case of the input signal having a sampling rate of 22050 Hz. For the sake of example, the filter heights in Figure 8 have been scaled to unity.

Variations may be made in the filter bank in other embodiments, such as spanning the band centre frequencies linearly below 1000 Hz, scaling the filters such that they have unit area instead of unity height, varying the number of frequency bands, or changing the range of frequencies spanned by the filters.

The weighted sum of the magnitudes from each of the filter bank channels may be referred to as mel-band energies $\tilde{m}_j$,where $j=1...N$, N being the number of filters.

In step s5.7, a logarithm, such as a logarithm of base 10, may be taken from the mel-band energies $\tilde{m}_j$, producing log mel-band energies m_j. An example of a log mel-band energy spectrum is shown in Figure 9.

Next, at step s5.8, a Discrete Cosine Transform is applied to a vector of the log mel-band energies m_j to obtain the MFCCs according to Equation (2),

$$c_{mel}(i) = \sum_{j=1}^{N} m_j \cos\left(\frac{\pi \cdot i}{N}\left(j - \frac{1}{2}\right)\right) \quad (2)$$

where N is the number of filters, $i=0,\ldots, I$ and I is the number of MFCCs. In an exemplary embodiment, $I=20$.

At step s5.9, further mathematical operations may be performed on the MFCCs produced at step s5.8, such as calculating a mean of the MFCCs and/or time derivatives of the MFCCs to produce the required audio features 31 on which the calculation of the first and second classifications by the first and second classifiers 33, 34 will be based.

In this particular embodiment, the audio features 31 produced at step s5.9 include one or more of:

- a MFCC matrix for the audio track;
- first and, optionally, second time derivatives of the MFCCs, also referred to as "delta MFCCs";
- a mean of the MFCCs of the audio track;
- a covariance matrix of the MFCCs of the audio track;
- an average of mel-band energies over the audio track, based on output from the channels of the filter bank obtained in step s5.6;
- a standard deviation of the mel-band energies over the audio track;
- an average logarithmic energy over the frames of the audio track, obtained as an average of $\underline{c_{mel}(0)}$ over a period of time obtained, for example, using Equation (2) at step s4.8; and
- a standard deviation of the logarithmic energy.

The extracted features 31 are then output (step s5.10) and the feature extraction method ends (step s5.11).

As noted above, the features 31 extracted at step s4.2 may also include a fluctuation pattern and danceability features for the track, such as:

- a median fluctuation pattern over the song;
- a fluctuation pattern bass feature;
- a fluctuation pattern gravity feature;
- a fluctuation pattern focus feature;

- a fluctuation pattern maximum feature;
- a fluctuation pattern sum feature;
- a fluctuation pattern aggressiveness feature;
- a fluctuation pattern low-frequency domination feature;
- a danceability feature (detrended fluctuation analysis exponent for at least one predetermined time scale);and
- a club-likeness value.

The mel-band energies calculated in step s5.8 may be used to calculate one or more of the fluctuation pattern features listed above. In an example method of fluctuation pattern analysis, a sequence of logarithmic domain mel-band magnitude frames are arranged into segments of a desired temporal duration and the number of frequency bands is reduced. A FFT is applied over coefficients of each of the frequency bands across the frames of a segment to compute amplitude modulation frequencies of loudness in a described range, for example, in a range of 1 to 10 Hz. The amplitude modulation frequencies may be weighted and smoothing filters applied. The results of the fluctuation pattern analysis for each segment may take the form of a matrix with rows corresponding to modulation frequencies and columns corresponding to the reduced frequency bands and/or a vector based on those parameters for the segment. The vectors for multiple segments may be averaged to generate a fluctuation pattern vector to describe the audio track.

Danceability features and club-likeness values are related to beat strength, which may be loosely defined as a rhythmic characteristic that allows discrimination between pieces of music, or segments thereof, having the same tempo. Briefly, a piece of music characterised by a higher beat strength would be assumed to exhibit perceptually stronger and more pronounced beats than another piece of music having a lower beat strength. As noted above, a danceability feature may be obtained by detrended fluctuation analysis, which indicates correlations across different time scales, based on the mel-band energies obtained at step s5.8.

Examples of techniques of club-likeness analysis, fluctuation pattern analysis and detrended fluctuation analysis are disclosed in British patent application no. 1401626.5, as well as example methods for extracting MFCCs. The disclosure of GB 1401626.5 is incorporated herein by reference in its entirety.

The features 31 extracted at step s4.2 may include features relating to tempo in beats per minute (BPM), such as:

- an average of an accent signal in a low, or lowest, frequency band;
- a standard deviation of said accent signal;
- a maximum value of a median or mean of periodicity vectors;
- a sum of values of the median or mean of the periodicity vectors;
- tempo indicator for indicating whether a tempo identified for the input signal is considered constant, or essentially constant, or is considered non-constant, or ambiguous;
- a first BPM estimate and its confidence;
- a second BPM estimate and its confidence;
- a tracked BPM estimate over the audio track and its variation;
- a BPM estimate from a lightweight tempo estimator.

Example techniques for beat tracking, using accent information, are disclosed in US published patent application no. 2007/240558 A1, US patent application no. 14/302,057, International (PCT) published patent application nos. WO2013/164661 A1 and WO2014/001849 A1, the disclosures of which are hereby incorporated by reference in their entireties.

In one example beat tracking method, described in GB 1401626.5, one or more accent signals are derived from the input signal 50, for detection of events and/or changes in the audio track. A first one of the accent signals may be a chroma accent signal based on fundamental frequency F_O salience estimation, while a second one of the accent signals may be based on a multi-rate filter bank decomposition of the input signal 50.

A BPM estimate may be obtained based on a periodicity analysis for extraction of a sequence of periodicity vectors on the basis of the accent signals, where each periodicity vector includes a plurality of periodicity values, each periodicity value describing the strength of periodicity for a respective period length, or "lag". A point-wise mean or median of the periodicity vectors over time may be used to indicate a single representative periodicity vector over a time period of the audio track. For example, the time period may be over the whole duration of the audio track. Then, an analysis can be performed on the periodicity vector to determine a

most likely tempo for the audio track. One example approach comprises performing k-nearest neighbours regression to determine the tempo. In this case, the system is trained with representative music tracks with known tempo. The k-nearest neighours regression is then used to predict the tempo value of the audio track based on the tempi of k-nearest representative tracks. More details of such an approach have been described in Eronen, Klapuri, "Music Tempo Estimation With k -NN Regression", IEEE Transactions on Audio, Speech, and Language Processing, Vol. 18, Issue 1, pages 50-57, the disclosure of which is incorporated herein by reference.

Chorus related features that may be extracted at step s3.2 include:

- a chorus start time; and
- a chorus end time.

Example systems and methods that can be used to detect chorus related features are disclosed in US 2008/236371 A1, the disclosure of which is hereby incorporated by reference in its entirety.

Other features that may be used as additional input include:

- a duration of the audio track in seconds,
- an A-weighted sound pressure level (SPL);
- a standard deviation of the SPL;
- an average brightness, or spectral centroid (SC), of the audio track, calculated as a spectral balancing point of a windowed FFT signal magnitude in frames of, for example, 40 ms in length;
- a standard deviation of the brightness;
- an average low frequency ratio (LFR), calculated as a ratio of energy of the input signal below 100Hz to total energy of the input signal, using a windowed FFT signal magnitude in 40 ms frames; and
- a standard deviation of the low frequency ratio.

Figure 10 is an overview of a process of extracting multiple acoustic features 31, some or all of which may be obtained in step s4.2. Figure 10 shows how some input features are derived, at least in part, from computations of other input features. The features 31 shown in Figure 10 include the MFCCs, delta MFCCs and mel-band

energies discussed above in relation to Figure 5, indicated using bold text and solid lines.

The dashed lines and standard text in Figure 10 indicate other features that may be extracted and made available alongside, or instead of, the MFCCs, delta MFCCs and mel-band energies, for use in calculating the first level classifications. For example, as discussed above, the mel-band energies may be used to calculate fluctuation pattern features and/or danceability features through detrended fluctuation analysis. Results of fluctuation pattern analysis and detrended fluctuation analysis may then be used to obtain a club-likeness value. Also as noted above, beat tracking features, labeled as "beat tracking 2" in Figure 10, may be calculated based, in part, on a chroma accent signal from a F_0 salience estimation.

Returning to Figure 4, in steps s4.3 to s4.10, the software application 212 causes the controller 202 to produce the first level classifications, that is the first classifications and the second classifications, based on the features 31 extracted in step s4.2. Although Figure 4 shows steps s4.3 to s4.10 being performed sequentially, in other embodiments, steps s4.3 to s4.7 may be performed after, or in parallel with, steps s4.8 to s4.10.

The first and second classifications are generated using the first classifiers 33 and the second classifiers 34 respectively, where the first and second classifiers 33, 34 are different from one another. For instance, the first classifiers 33 may be non-probabilistic and the second classifiers 34 may be probabilistic classifiers, or vice versa. In this particular embodiment, the first classifiers 33 are support vector machine (SVM) classifiers, which are non-probabilistic. Meanwhile, the second classifiers 34 are based on one or more Gaussian Mixture Models (GMMs).

In step s4.3, one, some or all of the features 31 or descriptors extracted in step s4.2, to be used to produce the first classifications 33, are selected and, optionally, normalised. For example, a look up table 216 or database may be stored in the memory 206 of the for each of the first classifications to be produced by the analysis server 100, that provides a list of features to be used to generate each first classifier and statistics, such as mean and variance of the selected features, that can be used in normalisation of the extracted features 31. In such an example, the controller 202 retrieves the list of features from the look up table 216, and accordingly selects and normalises the listed features for each of the first

classifications to be generated. The normalisation statistics for each first classification in the database may be determined during training of the first classifiers 33.

As noted above, in this example, the first classifiers 33 are SVM classifiers. The SVM classifiers 33 are trained using a database of audio tracks for which information regarding musical instruments and genre is already available. The database may include tens of thousands of tracks for each particular musical instrument that might be tagged.

Examples of musical instruments for which information may be provided in the database include:

- Accordion;
- Acoustic guitar;
- Backing vocals;
- Banjo;
- Bass synthesizer;
- Brass instruments;
- Glockenspiel;
- Drums;
- Eggs;
- Electric guitar;
- Electric piano;
- Guitar synthesizer;
- Keyboards;
- Lead vocals;
- Organ;
- Percussion;
- Piano;
- Saxophone;
- Stringed instruments;
- Synthesizer; and

- Woodwind instruments.

The training database includes indications of genres that the audio tracks belong to, as well as indications of genres that the audio tracks do not belong to. Examples of musical genres that may be indicated in the database include:

- Ambient and new age;
- Blues;
- Classical;
- Country and western;
- Dance;
- Easy listening;
- Electronica;
- Folk and roots;
- Indie and alternative;
- Jazz;
- Latin;
- Metal;
- Pop;
- Rap and hip hop;
- Reggae;
- Rock;
- Soul, R&B and funk; and
- World music.

By analysing features 31 extracted from the audio tracks in the training database, for which instruments and/or genre are known, a SVM classifier 33 can be trained to determine whether or not an audio track includes a particular instrument, for example, an electric guitar. Similarly, another SVM classifier 33 can be trained to determine whether or not the audio track belongs to a particular genre, such as Metal.

In this embodiment, the training database provides a highly imbalanced selection of audio tracks, in that a set of tracks for training a given SVM classifier 33 includes many more positive examples than negative ones. In other words, for training a SVM classifier 33 to detect the presence of a particular instrument, a set of audio tracks for training in which the number of tracks that include that instrument is significantly greater than the number of tracks that do not include that instrument will be used. Similarly, in an example where a SVM classifier 33 is being trained to determine whether an audio track belongs to a particular genre, the set of audio tracks for training might be selected so that the number of tracks that belong to that genre is significantly greater than the number of tracks that do not belong to that genre.

An error cost may be assigned to the different first classifications 33 to take account of the imbalances in the training sets. For example, if a minority class of the training set for a particular first classification includes 400 songs and an associated majority class contains 10,000 tracks, an error cost of 1 may be assigned to the minority set and an error cost of 400/10,000 may be assigned to the majority class. This allows all of the training data to be retained, instead of downsampling data of the negative examples.

New SVM classifiers can be added by collecting new training data and training the new classifiers. Since the SVM classifiers 33 are binary, new classifiers can be added alongside existing classifiers.

As mentioned above, the training process can include determining a selection of one or more features 31 to be used as a basis for particular first classifications and statistics for normalising those features 31. The number of features available for selection, M, may be much greater than the number of features selected for determining a particular first classification, N; that is, $M >> N$. The selection of features 31 to be used is determined iteratively, based on a development set of audio tracks for which the relevant instrument or genre information is available, as follows.

Firstly, to reduce redundancy, a check is made as to whether two or more of the features are so highly correlated that the inclusion of more than one of those features would not be beneficial. For example, if two features have a correlation

coefficient that is larger than 0.9, then only one of those features is considered available for selection.

The feature selection training starts using an initial selection of features, such as the average MFCCs for audio tracks in the development set or a single "best" feature for a given first classification. For instance, a feature that yields the largest classification accuracy when used individually may be selected as the "best" feature and used as the sole feature in an initial feature selection.

An accuracy of the first classification based on the initial feature selection is determined. Further features are then added to the feature selection to determine whether or not the accuracy of the first classification is improved by their inclusion.

Features to be tested for addition to the selection of features may be chosen using a method that combines forward feature selection and backward feature selection in a sequential floating feature selection. Such feature selection may be performed during the training stage, by evaluating the classification accuracy on a portion of the training set.

In each iteration, each of the features available for selection is added to the existing feature selection in turn, and the accuracy of the first classification with each additional feature is determined. The feature selection is then updated to include the feature that, when added to the feature selection, provided the largest increase in the classification accuracy for the development set.

After a feature is added to the feature selection, the accuracy of the first classification is reassessed, by generating first classifications based on edited features selections in which each of the features in the feature selection is omitted in turn. If it is found that the omission of one or more features provides an improvement in classification accuracy, then the feature that, when omitted, leads to the biggest improvement in classification accuracy is removed from the feature selection. If no improvements are found when any of the existing features are left out, but the classification accuracy does not change when a particular feature is omitted, that feature may also be removed from the feature selection in order to reduce redundancy.

The iterative process of adding and removing features to and from the feature selection continues until the addition of a further feature no longer provides a significant improvement in the accuracy of the first classification. For example, if the improvement in accuracy falls below a given percentage, the iterative process may be considered complete, and the current selection of features is stored in the lookup table 216, for use in selecting features in step s4.2.

The normalisation of the selected features 31 at step s4.3 is optional. Where provided, the normalization of the selected features 31 in step s4.3 may potentially improve the accuracy of the first classifications.

In another embodiment, at step s4.3, a linear feature transform may be applied to the available features 31 extracted in step s4.2, instead of performing the feature selection procedure described above. For example, a Partial Least Squares Discriminant Analysis (PLS-DA) may be used to obtain a linear combination of features for calculating a corresponding first classification. Instead of using the above iterative process to select N features from the set of M features, a linear feature transform is applied to an initial high-dimensional set of features to arrive at a smaller set of features which provides a good discrimination between classes. The initial set of features may include some or all of the available features, such as those shown in Figure 10, from which a reduced set of features can be selected based on the result of the transform.

The PLS-DA transform parameters may be optimized and stored in a training stage. During the training stage, the transform parameters and its dimensionality may be optimized for each tag or output classification, such as an indication of an instrument or a genre. More specifically, the training of the system parameters can be done in a cross-validation manner, for example, as five-fold cross-validation, where all the available data is divided into five non-overlapping sets. At each fold, one of the sets is held out for evaluation and the four remaining sets are used for training. Furthermore, the division of folds may be specific for each tag or classification.

For each fold and each tag or classification, the training set is split into 50%-50% inner training-test folds. Then, the PLS-DA transform may be trained on the inner training-test folds and the SVM classifier 33 may be trained on the obtained dimensions. The accuracy of the SVM classifier 33 using the transformed features

transformed may be evaluated on the inner test fold. It is noted that, when a feature vector (track) is tested, it is subjected to the same PLS-DA transform, the parameters of which were obtained during training. This manner, an optimal dimensionality for the PLS-DA transform may be selected. For example, the dimensionality may be selected such that the area under the receiver operating characteristic (ROC) curve is maximized. In one example embodiment, an optimal dimensionality is selected among candidates between 5 to 40 dimensions. Hence, the PLS-DA transform is trained on the whole of the training set, using the optimal number of dimensions, and then used in training the SVM classifier 33.

In the following, an example is discussed in which the selected features 31 on which the first classifications are based are the mean of the MFCCs of the audio track and the covariance matrix of the MFCCs of the audio track, although in other examples alternative and/or additional features, such as the other features shown in Figure 10, may be used.

At step s4.4, the controller 202 defines a single "feature vector" for each set of selected features 31 or selected combination of features 31.

The feature vectors may then be normalized to have a zero mean and a variance of 1, based on statistics determined and stored during the training process.

At step s4.5, the controller 202 generates one or more first probabilities that the audio track has a certain characteristic, corresponding to a potential tag 37, based on the normalized transformed feature vector or vectors. A first classifier 33 is used to calculate a respective probability for each feature vector defined in step s4.4. In this manner, the number of SVM classifiers 33 corresponds to the number of characteristics or tags 37 to be predicted for the audio track.

In this particular example, a probability is generated for each instrument tag and for each musical genre tag to be predicted for the audio track, based on the mean MFCCs and the MFCC covariance matrix. In addition, a probability may be generated based on whether the audio track is likely to be an instrumental track or a vocal track. Also, for vocal tracks, another first classification may be generated based on whether the vocals are provided by a male or female vocalist. In other embodiments, the controller may generate only one or some of these probabilities and/or calculate additional probabilities at step 4.5. The different classifications

may be based on respective selections of features from the available features 31 extracted in step s4.2.

The SVM classifiers 33 may use a radial basis function (RBF) kernel *K*, defined as:

$$K(\vec{u},\vec{v}) = e^{-\gamma\left(\|\vec{u}-\vec{v}\|^2\right)} \quad (3)$$

where the default γ parameter is the reciprocal of the number of features in the feature vector, $\vec{u}$ is the input feature vector and $\vec{v}$ is a support vector.

The first classifications may be based on an optimal predicted probability threshold that separates a positive prediction from a negative prediction for a particular tag, based on the probabilities output by the SVM classifiers 33. The setting of an optimal predicted probability threshold may be learned in the training procedure to be described later below. Where there is no imbalance in data used to train the first classifiers 33, the optimal predicted probability threshold may be 0.5. However, where there is an imbalance between the number of tracks providing positive examples and the number of tracks provided negative examples in the training sets used to train the first classifiers 33, the threshold p_{thr} may be set to a prior probability of a minority class P_{min} in the first classification, using Equation (4) as follows:

$$p_{thr} = P_{\min} = \frac{n_{\min}}{n_{maj}} \quad (4)$$

where, in the set of *n* tracks used to train the SVM classifiers, n_{min} is the number of tracks in the minority class and n_{maj} is the number of tracks in a majority class.

The prior probability P_{min} may be learned as part of the training of the SVM classifier 33.

Probability distributions for examples of possible first classifications, based on an evaluation of a number *n* of tracks, are shown in Figure 11. The nine examples in Figure 11 suggest a correspondence between a prior probability for a given first classification and its probability distribution based on the *n* tracks. Such a correspondence is particularly marked where the SVM classifier 33 was trained

with an imbalanced training set of tracks. Consequently, the predicted probability threshold for the different examples vary over a considerable range.

Optionally, a logarithmic transformation may be applied to the probabilities output by the SVM classifiers 33 (step s4.6), so that the probabilities of all the first classifications are on the same scale and the optimal predicated probability threshold may correspond to a predetermined value, such as 0.5.

Equations (5) to (7) below provide an example normalization which adjusts the optimal predicted probability threshold to 0.5. Where the probability output by a SVM classifier 33 is p and the prior probability P of a particular tag being applicable to a track is greater than 0.5, then the normalized probability p_{norm} is given by:

$$p_{norm} = 1 - (1 - p)^L \tag{5}$$

$$\text{where } L = \frac{\log(0.5)}{\log(1 - P)} \tag{6}$$

Meanwhile, where the prior probability P is less than or equal to 0.5, then the normalised probability p_{norm} is given by:

$$p_{norm} = p^{L'} \tag{7}$$

$$\text{where } L' = \frac{\log(0.5)}{\log(P)} \tag{8}$$

Figure 12 depicts the example probability distributions of Figure 11 after a logarithmic transformation has been applied, on which optimal predicated probability thresholds of 0.5 are marked.

The first classifications are then output (step s4.7). The first classifications correspond to the normalised probability p_{norm} that a respective one of the tags 37 to be considered applies to the audio track. The first classifications may include probabilities p_{inst1} that a particular instrument is included in the audio track and probabilities p_{gen1} that the audio track belongs to a particular genre.

Returning to Figure 4, in steps s4.8 to s4.10, second classifications for the input signal are based on the MFCCs and other parameters produced in step s4.2, using the second classifiers 34. In this particular example, the features 31 on which the second classifications are based are the MFCC matrix for the audio track and the first time derivatives of the MFCCs.

In steps s4.8 to s4.10, the probabilities of the audio track including a particular instrument or belonging to a particular genre are assessed using probabilistic models that have been trained to represent the distribution of features extracted from audio signals captured from each instrument or genre. As noted above, in this example the probabilistic models are GMMs. Such models can be trained using an expectation maximisation algorithm that iteratively adjusts the model parameters to maximise the likelihood of the model for a particular instrument or genre generating features matching one or more input features in the captured audio signals for that instrument or genre. The parameters of the trained probabilistic models may be stored in a database, for example, in the database 208 if the analysis server 100, or in remote storage that is accessible to the analysis server 100 via a network, such as the network 102.

For each instrument or genre, at least one likelihood is evaluated that the respective probabilistic model could have generated the selected or transformed features from the input signal. The second classifications correspond to the models which have the largest likelihood of having generated the features of the input signal.

In this example, probabilities are generated for each instrument tag at step s4.8 and for each musical genre tag at step s4.9, as well as a probability whether the audio track is likely to be an instrumental track or a vocal track may also be generated. Also, for vocal tracks, another probability may be generated based on whether the vocals are provided by a male or female vocalist. In other embodiments, the controller 202 may generate only one or some of these second classifications and/or calculate additional second classifications at steps s4.8 and s4.9.

In this embodiment, in steps s4.8 and s4.9, probabilities p_{inst2} that the instrument tags will apply, or not apply, are produced by the second classifiers 34 using first

and second Gaussian Mixture Models (GMMs), based on the MFCCs and their first time derivatives calculated in step s4.2. Meanwhile, probabilities p_{gen2} that the audio track belongs to a particular musical genre are produced by the second classifiers 34 using third GMMs. However, the first and second GMMs used to compute the instrument-based probabilities p_{inst2} may be trained and used slightly differently from third GMMs used to compute the genre-based probabilities $_{gen2}$, as will now be explained.

In the following, step s4.8 precedes step s4.9. However, in other embodiments, step s4.9 may be performed before, or in parallel with, step s4.8.

In this particular example, first and second GMMs are used to generate the instrument-based probabilities p_{inst2} (step s4.8),based on MFCC features 31 obtained in step s4.2.

The first and second GMMs used in step s4.8 may have been trained with an Expectation-Maximisation (EM) algorithm, using a training set of examples which are known either to include the instrument and examples which are known to not include the instrument. For each track in the training set, MFCC feature vectors and their corresponding first time derivatives are computed. The MFCC feature vectors for the examples in the training set that contain the instrument are used to train a first GMM for that instrument, while the MFCC feature vectors for the examples that do not contain the instrument are used to train a second GMM for that instrument. In this manner, for each instrument to be tagged, two GMMs are produced. The first GMM is for a track that includes the instrument, while the second GMM is for a track that does not include the instrument. In this example, the first and second GMMs each contain 64 component Gaussians.

The first and second GMMs may then be refined by discriminative training using a maximum mutual information (MMI) criterion on a balanced training set where, for each instrument to be tagged, the number of example tracks that contain the instrument is equal to the number of example tracks that do not contain the instrument.

Returning to the determination of the second classifications, two likelihoods are computed based on the first and second GMMs and the MFCCs for the audio track. The first is a likelihood that the corresponding instrument tag applies to the track,

referred to as L_{yes}, while the second is a likelihood that the instrument tag does not apply to the track, referred to as L_{no}. The first and second likelihoods may be computed in a log-domain, and then converted to a linear domain.

The first and second likelihoods L_{yes}, L_{no} are then mapped to a probability p_{inst2} of the tag applying as follows:

$$p_{inst2} = \frac{L_{yes}}{\left(L_{yes} + L_{no}\right)} \tag{9}$$

As noted above, the third GMMs, used for genre-based classification, are trained differently to the first and second GMMs. For each genre to be considered, a third GMM is trained based on MFCCs for a training set of tracks known to belong to that genre. One third GMM is produced for each genre to be considered. In this example, the third GMM includes 64 component Gaussians.

In step s4.9, for each of the genres that may be tagged, a likelihood L is computed for the audio track belonging to that genre, based on the likelihood of each of the third GMMs being capable of outputting the MFCC feature vector of the audio track. For example, to determine which of the eighteen genres in the list hereinabove might apply to the audio track, eighteen likelihoods would be produced.

The genre likelihoods are then mapped to probabilities p_{gen2}, as follows:

$$p_{gen2}(i) = \frac{L(i)}{\sum_{j=1}^{m} L(j)} \tag{10}$$

where m is the number of genre tags to be considered.

The second classifications, which correspond to the probabilities p_{inst2} and p_{gen2}, are then output (step s4.10).

In another embodiment, the first and second GMMs for analysing the instruments included in the audio track may be trained and used in the manner described above for the third GMMs. In yet further embodiments, the GMMs used for analysing

genre may be trained and used in the same manner, using either of techniques described in relation to the first, second and third GMMs above.

The first classifications p_{inst1} and p_{gen1} and the second classifications p_{inst2} and p_{gen2} for the audio track are normalized to have a mean of zero and a variance of 1 (step s4.11) and collected to form a feature vector for input to one or more second level classifiers 35 (step s4.12). In this particular example, the second level classifiers 35 include third classifiers 36. The third classifiers 36 may be non-probabilistic classifiers, such as SVM classifiers.

The third classifiers 36 may be trained in a similar manner to that described above in relation to the first classifiers 33. At the training stage, the first classifiers 33 and the second classifiers 34 may be used to output probabilities for the training sets of example audio tracks from the database. The outputs from the first and second classifiers 33, 34 are then used as input data to train the third classifier 35.

The third classifier 36 generates determine probabilities p_{inst3} for whether the audio track contains a particular instrument and/or probabilities p_{gen3} for whether the audio track belongs to a particular genre (step s4.13).

The probabilities p_{inst3} , p_{gen3} are then log normalised (step s4.14), as described above in relation to the first classifications, so that a threshold of 0.5 may be applied to generate the third classifications, which are output at step s4.15.

The controller 202 then determines whether each instrument tag and each genre tag 37 applies to the audio track based on the third classifications (step s4.16).

Where it is determined that an instrument or genre tag 37 applies to the audio track (step s4.16), the tag 37 is associated with the track (step s4.17), for example, by storing an indication that the tag 37 applies as part of metadata for the audio track.

Alternatively, or additionally, the probabilities themselves and/or the features 31 extracted at step s4.2 may be output for further analysis and/or storage.

Optionally, if a recommendation of other audio tracks based on the analysed audio track is required, for example if a user wishes to find music that has a similarity to the analysed audio track, the controller 202 may then search for one or more tracks

having matching tags in an existing catalogue (step s4.18). The catalogue may be a database stored in the memory of the analysis server 100 or accessible via the network 102 or other network. Information identifying one or more matching tracks may then be output (step s4.19), for example, by being transmitted to the device 104 for presentation on display 106.

The process ends at step s4.20.

In this manner, the method of Figure 4 can provide a multi-level analysis of the audio track and corresponding auto-tagging.

Figure 13 is an overview of a method according to another embodiment, which may be performed by the analysis server 100 of Figure 2. In the method, the controller 202 of the analysis server 100 acts as a track matcher 131 and a data retriever 135, as well as a feature extractor 30 and a tagging module 38.

The controller 202 can access a catalogue of tracks that already have one or more of tags, features, metadata or other information available. The catalogue may be stored in the database 208 in the analysis server 100, or in a remote storage accessible via the network 102 or other network.

The track matcher 131 searches the database for a matching track. The search for a matching track may be based on an audio-fingerprint 133 for some or all of the audio data, where the audio-fingerprint 133 has been accessed or otherwise obtained by the controller 202, and/or available metadata 134.

If a matching catalogue track is found, then the data retriever 135 collects information 133, such as tags, metadata and so on, for the matching catalogue track from the database 208. The information 135 can then be associated with the audio data, for example by storing tag information for the audio data. Therefore, if a match can be found based on the audio finger-print 133 or metadata 134, the information 135 for the audio data can be obtained without having to extract or analyse acoustic features 31 of the audio data, reducing the computation load required to obtain that information 135.

If a matching catalogue track is not found, then one or more acoustic features 31 are extracted from the audio data and a search for a catalogue track having

matching feature(s) is made. The controller 202 may repeatedly extract and search for subsets of features from the audio data, stopping the search and retrieving the information 135 when a matching track is found. Therefore, the method of Figure 13 may decrease the computing resources required to analyse and tag an audio track, by reducing potentially unnecessary computation as, once a match has been found, the extraction of further acoustic features 31 and searching based on those features is not performed.

If no match is found based on the audio-fingerprint 133 or metadata 134 no further features remain to be extracted are analysed, then the tagging module 38 updates the catalogue by uploading information 136 for the unmatched audio data. The information 136 may include one or more of the audio-fingerprint 133, metadata 134 and some or all of the extracted features 31. Optionally, the controller 202 may proceed to obtain first, second and third classifications, as described above with reference to Figures 3 and 4, and/or tags 37, and upload those to the catalogue.

Figure 14 is a flowchart showing the method of Figure 13 in more detail. Starting at step s14.0, if the received input signal is in a compressed format, it is decoded into pulse code modulation (PCM) data (step s14.1), in a similar manner to that described above in relation to step s4.1.

The controller 202 then determines whether the received input signal matches an audio track that already has tag information available (steps s14.2 to s14.9). In this particular example, a number of different techniques are used to search for a match in an existing catalogue of audio tracks, as follows.

One suitable technique is audio-fingerprinting (steps s14.2 to s14.4). A compact acoustic feature set is extracted from at least a part of an audio waveform obtained from the input signal 50 to form a feature vector. The extracted feature set is then used as the audio-fingerprint 133 (step s14.2) and the catalogue of previously analysed tracks is searched for a track having an exactly, or nearly exactly, matching feature vector (step s14.3).

If a track with a matching audio-fingerprint is found (step s14.4), the tags and/or probabilities for the matching track are retrieved from the catalogue (step s14.5), output and/or stored as the tags and/or probabilities for the audio track (step s14.6), without the controller 202 having to classify the input signal.

In this particular embodiment, decoding of the input signal (step s14.1) is performed before audio-fingerprinting (steps s14.2 to s14.4). However, in other embodiments, the audio-fingerprint may be calculated (step s14.2) directly from the compressed input signal before decoding (step s14.1) is performed. In such other embodiments, if a match is found (step s14.4), the data for the matching track can be retrieved (step s14.5) and output or stored (step s14.6) without having to decode the input signal.

In another technique for finding a match, metadata, such as artist, track name, album name or other textual metadata for the audio track is obtained (step s14.7), for example by extracting metadata from the input signal, such as metadata from metadata fields of a container format storing the encoded audio waveform of the audio track or from receiving manual input from a user of a device 104. The controller 202 then searches for tracks in the catalogue with matching metadata (step s14.8). In some embodiments, fuzzy string matching, for example using the Levenshtein distance, can be used to allow for minor differences in the textual metadata.

If a match is found (step s14.9), then the tags and/or probabilities for the matching track are retrieved (step s14.5), output and/or stored as the tags and/or probabilities for the audio track in the input signal (step s14.6), without performing a full classification of the input signal.

Figure 14 shows metadata matching (steps s14.7 to s14.9) being performed only in the event of no matching tracks being found using audio-fingerprinting (steps s14.2 to s14.4). However, other embodiments might utilise only one of these matching techniques. Further embodiments may attempt to find a match using metadata matching and perform audio-fingerprinting in the event of no such match being identified, or perform both techniques in parallel. In yet further embodiments, other techniques for finding matching audio tracks may be used instead of, or as well as, audio-fingerprinting and metadata matching.

If no match has been found in the audio tracks in the existing catalogue (step s14.9), then the controller 202 begins to extract features 31 from the input signal (step s14.10) and searches the existing catalogue for a previously analysed track having matching feature values (step s14.11).

The extraction of features is described above in relation to Figures 5 and 10. However, in this embodiment, the controller 202 extracts subsets of the one or more features 31 discussed above. If a match is not found based on the extracted features 31 (step s14.12), then the further features 31 are extracted (step s14.10) and a search is made based on the further features 31 (step s14.11) until a match is found (step s14.12) or no further features are to be analysed (step s14.13).

Since the controller 202 stops extracting features 31 (step s14.10) from the audio data if a match has been found (step s14.12), the amount of computation needed to classify the audio track may be reduced because, in at least some instances, not all of the features 31 will need to be utilised in order to locate a match.

In some embodiments, the controller 202 may initially extract subsets of the features 31 that are computationally light compared with other features at step s14.10, to try to minimise the computation required to locate a matching track.

In another example, the controller may extract features according to dependency, so that features that provide input for other features are prioritised. For example, as shown in Figure 10, extract of chorus detection features is based, at least in part, on beat tracking, shown as beat tracking 1 and beat tracking 2 in Figure 10, and fundamental frequency F_0 salience based chroma, also known as pitch class. Therefore, in one example, beat tracking 1 features may be extracted at step s14.10 in a first iteration and used to search for a matching track in the catalogue at step s14.11. If a match is not found in that initial iteration (step s14.2), then beat tracking 2 features and F_0 salience chroma features may be determined in subsequent iterations (step s14.10), and chorus detection based features determined and used to search (step s14.11) in even later iterations.

An example order for feature extraction is:

- MFCCs, fluctuation pattern features, computationally light beat tracking features;
- Accent feature analysis results;
- Energy feature analysis results;
- Second phase, non-causal beat tracking features;
- Danceability and club-likeness analysis;
- Chorus analysis results;
- Classification of a track as being an instrumental or vocal track;

- Vocalist gender classification;
- First and second classifications for instruments;
- First and second classifications for genres;
- Third classification for instruments; and
- Third classification for genres.

If a match is found (step s14.12) between the extracted feature, or features, from the input signal and the feature(s) of a track in the catalogue with a high enough confidence, the tags and/or probabilities for the matching track are retrieved from the catalogue (step s14.5), output and/or stored as the tags and/or probabilities for the audio track (step s14.6). A high enough confidence may be determined at step s14.12 if only a single match is found in the catalogue.

If a match is not found at step s14.12, and no further features are to be extracted (step s14.13), then first, second and third classifications are computed for the audio track, as described above in relation to steps s4.3 to s4.16 of Figure 4 (step s14.14). Tags for the audio track are then determined based on the third classifications (step s14.15) and the audio track is tagged accordingly (step s14.16).

The controller 202 then transmits update information to the catalogue to include the newly analysed audio track and one or more of its tags, probabilities and features (step s14.17).

Optionally, if a recommendation of other audio tracks based on the analysed audio track is required, for example if a user wishes to find music that has a similarity to the analysed audio track, the controller 202 may then search for one or more audio tracks having matching tags in an existing catalogue (step s14.18). The catalogue may be a database stored in the memory of the analysis server 100 or accessible via the network 102 or other network. Information identifying one or more matching tracks may then be output (step s14.19), for example, for display to a user.

The process then ends (step s14.20).

The method of Figure 14 may be particularly useful where a large catalogue of audio tracks is available, either stored in a database in memory or on disk key value storage or the like at the analysis server 100 or accessible by the analysis server 100 via a network, such as the network 102. By providing a cache of previous analysis

results in the catalogue, it may be possible to respond to a request for analysis of an audio track using existing information for that track. In such a system, it might only be necessary to perform a full analysis for an audio track that is not already in the catalogue, potentially reducing computation and processing requirements.

The information in the catalogue may be based on one or more of results of automated analysis, such as the method of Figure 4, information obtained from web-crawling, human curation and social tags. For instance, human input may be used to complement data obtained automatically, by providing information that cannot be obtained through automated analysis, or to add extra information, or to verify the tags applied automatically.

The information in the catalogue may be used for searching the catalogue for audio tracks and/or recommending a track to a user based on similarity of features of audio tracks already accessed or ranked by the user.

It will be appreciated that the above-described embodiments are not limiting on the scope of the invention, which is defined by the appended claims and their alternatives. Various alternative implementations will be envisaged by the skilled person, and all such alternatives are intended to be within the scope of the claims.

It is noted that the disclosure of the present application should be understood to include any novel features or any novel combination of features either explicitly or implicitly disclosed herein or any generalization thereof and during the prosecution of the present application or of any application derived therefrom, new claims may be formulated to cover any such features and/or combination of such features.

Embodiments of the present invention may be implemented in software, hardware, application logic or a combination of software, hardware and application logic. The software, application logic and/or hardware may reside on memory, or any computer media. In an example embodiment, the application logic, software or an instruction set is maintained on any one of various conventional computer-readable media. In the context of this document, a "computer-readable medium" may be any media or means that can contain, store, communicate, propagate or transport the instructions for use by or in connection with an instruction execution system, apparatus, or device, such as a computer.

A computer-readable medium may comprise a computer-readable storage medium that may be any tangible media or means that can contain or store the instructions for use by or in connection with an instruction execution system, apparatus, or device, such as a computer as defined previously. The computer-readable medium may be a volatile medium or non-volatile medium.

According to various embodiments of the previous aspect of the present invention, the computer program according to any of the above aspects, may be implemented in a computer program product comprising a tangible computer-readable medium bearing computer program code embodied therein which can be used with the processor for the implementation of the functions described above.

Reference to "computer-readable storage medium", "computer program product", "tangibly embodied computer program" etc, or a "processor" or "processing circuit" etc. should be understood to encompass not only computers having differing architectures such as single/multi processor architectures and sequencers/parallel architectures, but also specialised circuits such as field programmable gate arrays FPGA, application specify circuits ASIC, signal processing devices and other devices. References to computer program, instructions, code etc. should be understood to express software for a programmable processor firmware such as the programmable content of a hardware device as instructions for a processor or configured or configuration settings for a fixed function device, gate array, programmable logic device, etc.

If desired, the different functions discussed herein may be performed in a different order and/or concurrently with each other. Furthermore, if desired, one or more of the above-described functions may be optional or may be combined.

Although various aspects of the invention are set out in the independent claims, other aspects of the invention comprise other combinations of features from the described embodiments and/or the dependent claims with the features of the independent claims, and not solely the combinations explicitly set out in the claims.

Claims

1. Apparatus comprising:
a controller; and
a memory in which is stored computer-readable instructions which, when executed by the controller, cause the controller to:
determine one or more acoustic features of audio data;
generate at least one first classification based on the one or more determined acoustic features using a respective first classifier;
generate at least one second classification based on the one or more determined acoustic features using at least one respective second classifier, the second classifier being different from the first classifier;
generate a third classification based on said first and second classifications using a third classifier; and
store at least one tag for said audio data based on said third classification.

2. Apparatus according to claim 1, wherein at least one of said first classifier and said third classifier is a non-probabilistic classifier.

3. Apparatus according to claim 2, wherein said non-probabilistic classifier is a support vector machine classifier.

4. Apparatus according to claim 1, 2 or 3, wherein said second classifier is a probabilistic classifier.

5. Apparatus according to claim 4, wherein said probabilistic classifier is based on one or more Gaussian Mixture Models.

6. Apparatus according to any of the preceding claims, wherein said metadata indicates at least one of:
a musical instrument;
a vocalist gender;
presence of vocals in the audio data;
presence of music; and
a musical genre.

7. Apparatus according to any of the preceding claims, wherein said one or more acoustic features include at least one feature based on a mel-frequency cepstral coefficient.

8. Apparatus according to claim 7, wherein said one or more acoustic features include at least one of:

a fluctuation pattern;

a danceability feature;

a feature relating to beats per minute;

a chorus-related feature;

a duration;

a feature relating to sound pressure level;

a brightness-related feature; and

a low frequency ratio related feature.

9. Apparatus according to any of the preceding claims, wherein the computer-readable instructions, when executed by the controller, cause the controller to select one or more tracks from a catalogue having one or more tags matching the at least one tag for audio data and to output information identifying said one or more selected tracks.

10. Apparatus according to any of the preceding claims, wherein the computer-readable instructions, when executed by the controller, cause the controller to store one or more of:

at least one of said first and second classifications; and

at least one of said one or more acoustic features of the audio data.

11. A method comprising:

determining one or more acoustic features of audio data;

generating at least one first classification based on the one or more determined acoustic features using a respective first classifier;

generating at least one second classification based on the one or more determined acoustic features using at least one respective second classifier, the second classifier being different from the first classifier;

generating a third classification based on said first and second classifications using a third classifier; and

storing at least one tag for said audio data based on said third classifier.

12. A method according to claim 11, wherein at least one of said first classifier and said third classifier is a non-probabilistic classifier.

13. A method according to claim 11 or 12, wherein said second classifier is a probabilistic classifier.

14. A method according to claim 11, 12 or 13, wherein said at least one tag indicates at least one of:

a musical instrument; and

a musical genre.

15. A method according to any of claims 11 to 14, wherein said one or more acoustic features include at least one feature based on a mel-frequency cepstral coefficient.

16. A method according to any of claims 11 to 15, further comprising selecting one or more tracks from a catalogue having similar metadata to said metadata of the audio data and outputting information identifying said one or more selected tracks.

17. A computer program comprising computer readable instructions which, when executed by a processing arrangement, causes the processing arrangement to perform a method according to any of claims 11 to 16.

18. A non-transitory tangible computer program product comprising computer readable instructions which, when executed by a processing arrangement, causes the processing arrangement to:

determine one or more acoustic features of audio data;

generate at least one first classification based on the one or more determined acoustic features using a respective first classifier;

generate at least one second classification based on the one or more determined acoustic features using at least one respective second classifier, the second classifier being different from the first classifier;

generate a third classification based on said first and second classifications using a third classifier; and

store at least one tag for said audio data based on said third classifier.

19. Apparatus comprising:

a controller; and

a memory in which is stored computer-readable instructions which, when executed by the controller, cause the controller to:

determine whether audio data matches an audio track in a catalogue of audio tracks, based on at least one of an audio-fingerprint and metadata of the audio data;

if a match is determined, retrieve at least one tag for the matching audio track from the catalogue and store at least one tag for the audio data corresponding to the retrieved at least one tag; and

if a match is not determined, then:

extract one or more acoustic features of the audio data;

determine whether the audio data matches an audio track in the catalogue of audio tracks, based on said one or more acoustic features;

if a match is determined, then retrieve at least one tag for the matching audio track from the catalogue and store at least one tag for the audio track corresponding to the retrieved at least one tag;

if a match is not determined, then upload to the catalogue at least one tag based on the extracted features of the audio track.

20. Apparatus according to claim 19, wherein said audio-fingerprint is based on an audio waveform of at least part of the audio data.

21. Apparatus according to claim 19 or 20, wherein extracting said one or more acoustic features comprises extracting a first subset of one or more acoustic features, wherein if a match is not determined based on said one or more acoustic features, said extracting one or more acoustic features and determining whether the audio data matches an audio track in the catalogue based on the one or more acoustic features is repeated for at least one further subset of one or more acoustic features.

22. Apparatus according to claim 21, wherein said first subset comprises one or more acoustic features that are computationally lighter than the one or more acoustic features of the at least one further subset.

23. Apparatus according to claim 21 or 22, wherein said at least one further subset comprises one or more acoustic features computed based on one or more acoustic features of the first subset.

24. Apparatus according to any of claims 19 to 23, wherein the audio data comprises a music track and said at least one tag indicates one of:

an instrument included in said music track; and

a genre of said music track.

25. Apparatus according to any of claims 19 to 24, wherein the computer-readable instructions, when executed by the controller, cause the controller to select one or more tracks from a catalogue having one or more tags matching the at least one tag for audio data and to output information identifying said one or more selected tracks.

26. Apparatus according to any of claims 19 to 25, wherein said one of more acoustic features includes at least one of:

mel-frequency cepstral coefficients;

a fluctuation pattern feature;

beat tracking;

an accent feature;

an energy feature;

second phase, non-causal beat tracking;

danceability and club-likeness analysis;

a chorus feature;

a classification of the audio data as being an instrumental or vocal track;

a vocalist gender classification;

a tag or classification indicating a musical instrument; and

a tag or classification indicating a musical genre.

27. Apparatus according to claim 26, wherein the computer-readable instructions, when executed by the controller, cause the controller to generate, for the audio data, one of:

at least one mel-frequency cepstral coefficient;

a tag or classification indicating a musical instrument; and

a tag or classification indicating a musical genre.

28. Apparatus according to claim 27, wherein the computer-readable instructions, when executed by the controller, cause the controller to:

generate a first classification indicating a musical instrument or genre based on the one or more extracted acoustic features using a respective first classifier;

generate at least one second classification based on the one or more extracted acoustic features using at least one respective second classifier, the second classifier being different from the first classifier; and

generate a third classification based on said first and second classifications using a third classifier;

wherein the at least one tag for said audio data based on said third classification.

29. A method comprising:

determining whether audio data matches an audio track in a catalogue of audio tracks, based on at least one of an audio-fingerprint and metadata of the audio data;

if a match is determined, retrieving at least one tag for the matching audio track from the catalogue and storing at least one tag for the audio data corresponding to the retrieved at least one tag; and

if a match is not determined, then:

extracting one or more acoustic features of the audio data;

determining whether the audio data matches an audio track in the catalogue of audio tracks, based on said one or more acoustic features;

if a match is determined, then retrieving at least one tag for the matching audio track from the catalogue and storing at least one tag for the audio data corresponding to the retrieved at least one tag;

if a match is not determined, then uploading to the catalogue at least one tag based on the extracted features of the audio track.

30. A method according to claim 29, wherein extracting said one or more acoustic features comprises extracting a first subset of one or more acoustic features, wherein if a match is not determined based on said one or more acoustic features, said extracting one or more acoustic features and determining whether the audio data matches an audio track in the catalogue based on the one or more acoustic features is repeated for at least one further subset of one or more acoustic features.

31. A method according to claim 30, wherein said first subset comprises one or more acoustic features that are computationally lighter than the one or more acoustic features of the at least one further subset.

32. A method according to claim 30 or 31, wherein said at least one further subset comprises one or more acoustic features computed based on one or more acoustic features of the first subset.

33. A method according to any of claims 29 to 31, wherein the audio data comprises a music track and said at least one tag indicates one of:

an instrument included in said music track; and

a genre of said music track.

34. A method according to claim 33, comprising generating, for the audio data, one of:

at least one mel-frequency cepstral coefficient;

a tag or classification indicating a musical instrument; and

a tag or classification indicating a musical genre.

35. A method according to claim 34, comprising:

generating a first classification indicating a musical instrument or genre based on the one or more extracted acoustic features using a respective first classifier;

generating at least one second classification based on the one or more extracted acoustic features using at least one respective second classifier, the second classifier being different from the first classifier; and

generating a third classification based on said first and second classifications using a third classifier;

wherein the at least one tag for said audio data is based on said third classification.

36. A computer program comprising computer readable instructions which, when executed by a processing arrangement, causes the processing arrangement to perform a method according to any of claims 29 to 35.

37. A non-transitory tangible computer program product, comprising computer readable instructions which, when executed by a processing arrangement, causes the processing arrangement to:

determine whether audio data matches an audio track in a catalogue of audio tracks, based on at least one of an audio-fingerprint and metadata of the audio data;

if a match is determined, retrieve at least one tag for the matching audio track from the catalogue and store at least one tag for the audio data corresponding to the retrieved at least one tag; and

if a match is not determined, then:

extract one or more acoustic features of the audio data;

determine whether the audio data matches an audio track in the catalogue of audio tracks, based on said one or more acoustic features;

if a match is determined, then retrieve metadata for the matching audio track from the catalogue and store at least one tag for the audio data corresponding to the retrieved at least one tag;

if a match is not determined, then upload to the catalogue at least one tag based on the extracted features of the audio track.

38. A non-transitory tangible computer program product according to claim 37, comprising computer readable instructions which, when executed by the processing arrangement, causes the processing arrangement to:

generate a first classification indicating a musical instrument or genre based on the one or more extracted acoustic features using a respective first classifier;

generate at least one second classification based on the one or more extracted acoustic features using at least one respective second classifier, the second classifier being different from the first classifier; and

generate a third classification based on said first and second classifications using a third classifier;

wherein the at least one tag for said audio data is based on said third classification.

39. An apparatus comprising:

a feature extractor to determine one or more acoustic features of audio data;

at least one first classifier to generate at least one first classification based on the one or more determined acoustic features;

at least one second classifier to determine at least one second classification based on the one or more determined acoustic features, the second classifier being different from the first classifier;

a third classifier configured to generate a third classification based on said first and second classifications using a third classifier; and

a tagging module to store at least one tag for said audio data based on said third classification.

40. An apparatus comprising:

a track matcher to determine whether audio data matches an audio track in a catalogue of audio tracks based on at least one of an audio-fingerprint and metadata of the audio data;

a feature extractor to extract acoustic features from the audio data;

a data retriever to retrieve at least one tag for the matching audio track from the catalogue and store at least one tag for the audio data corresponding to the retrieved at least one tag if a match is determined; and

a tagging module;

wherein the track matcher is configured to, if a matching audio track is not found based on the at least one of the audio-fingerprint and the metadata of the audio data, determine whether audio data matches an audio track in a catalogue of audio tracks based on said extracted acoustic features; and

the tagging module is configured to, if a match is not found based on the extracted acoustic features, upload to the catalogue at least one tag based on the extracted features of the audio track.

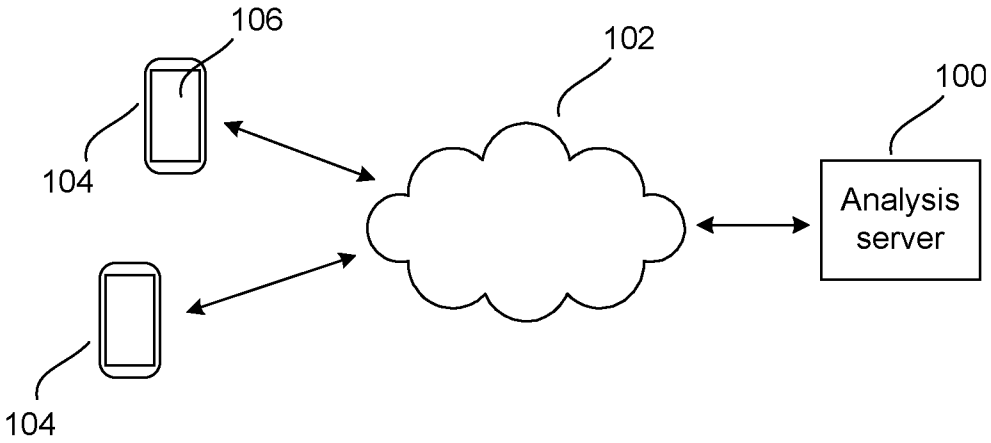

Figure 1

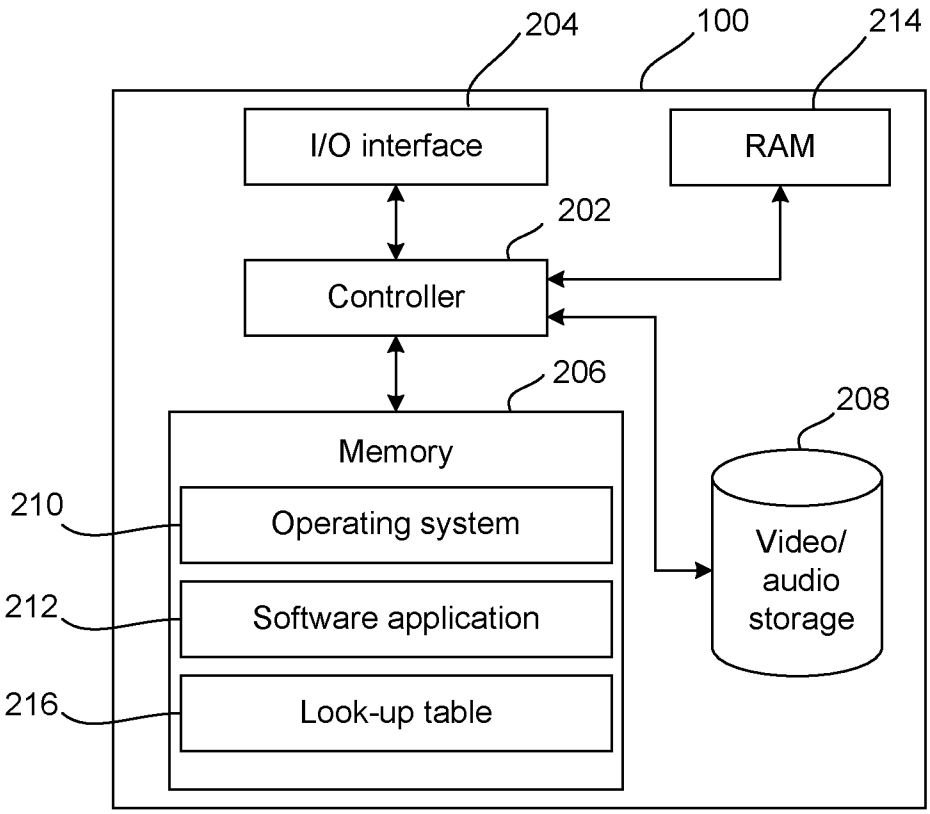

Figure 2

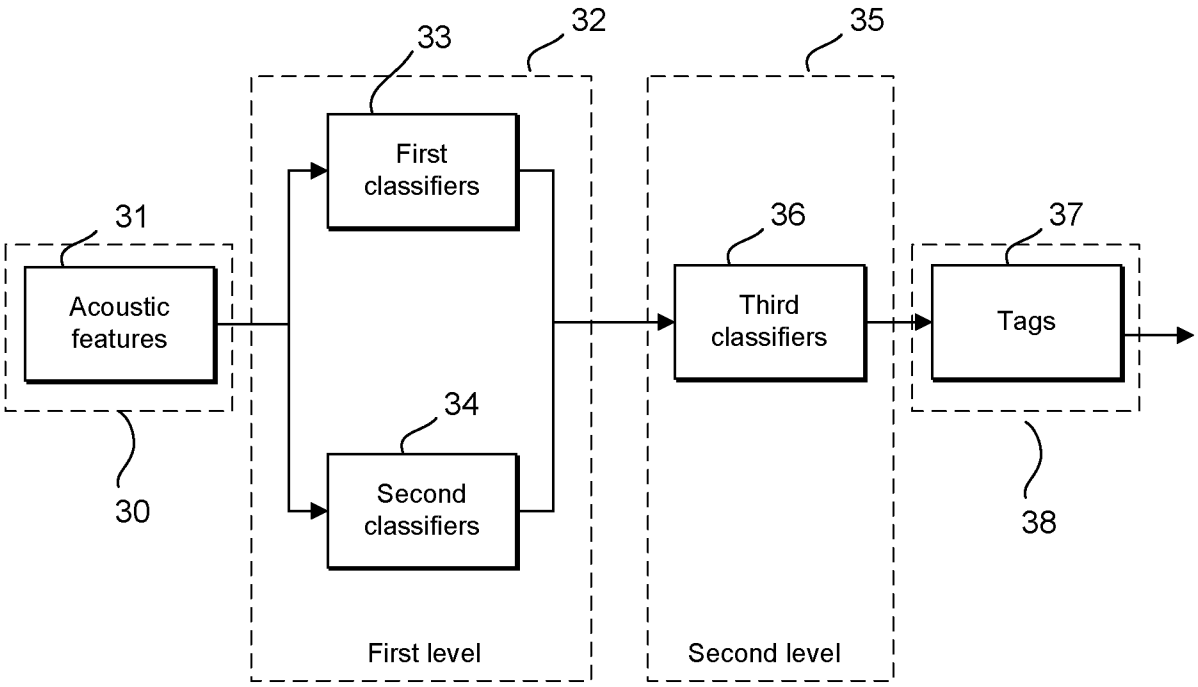

Figure 3

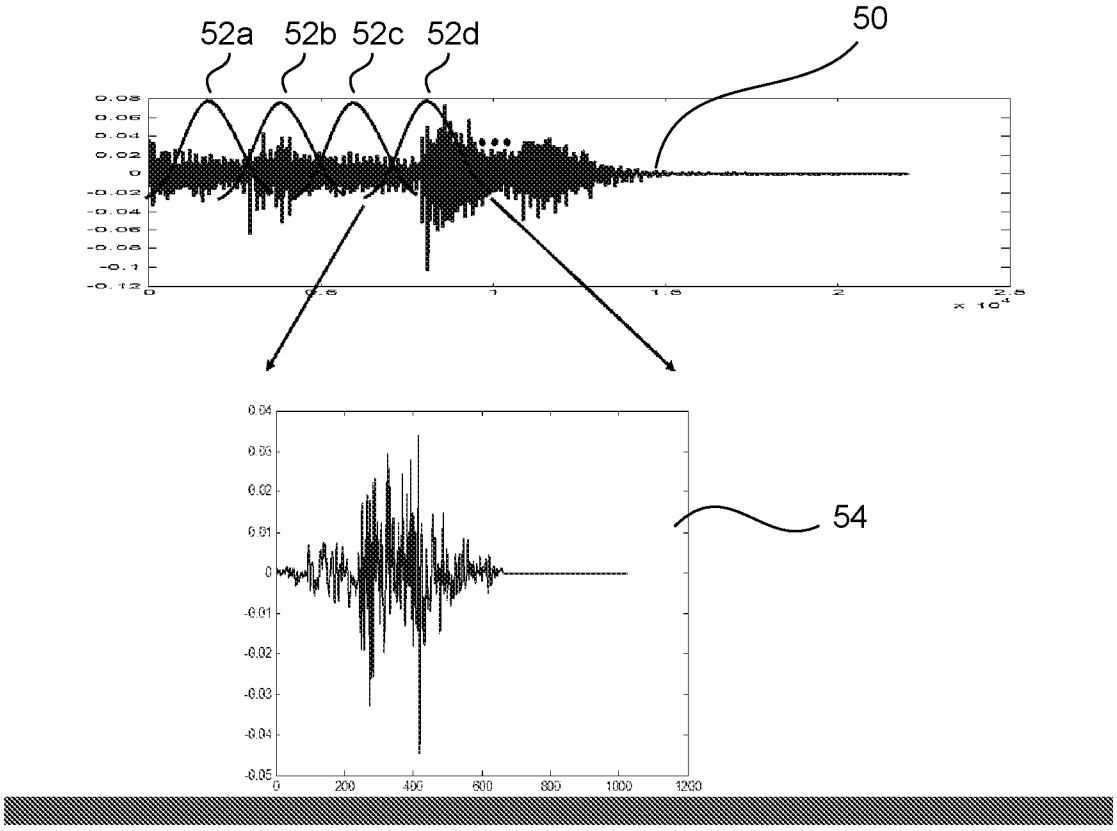

Figure 6

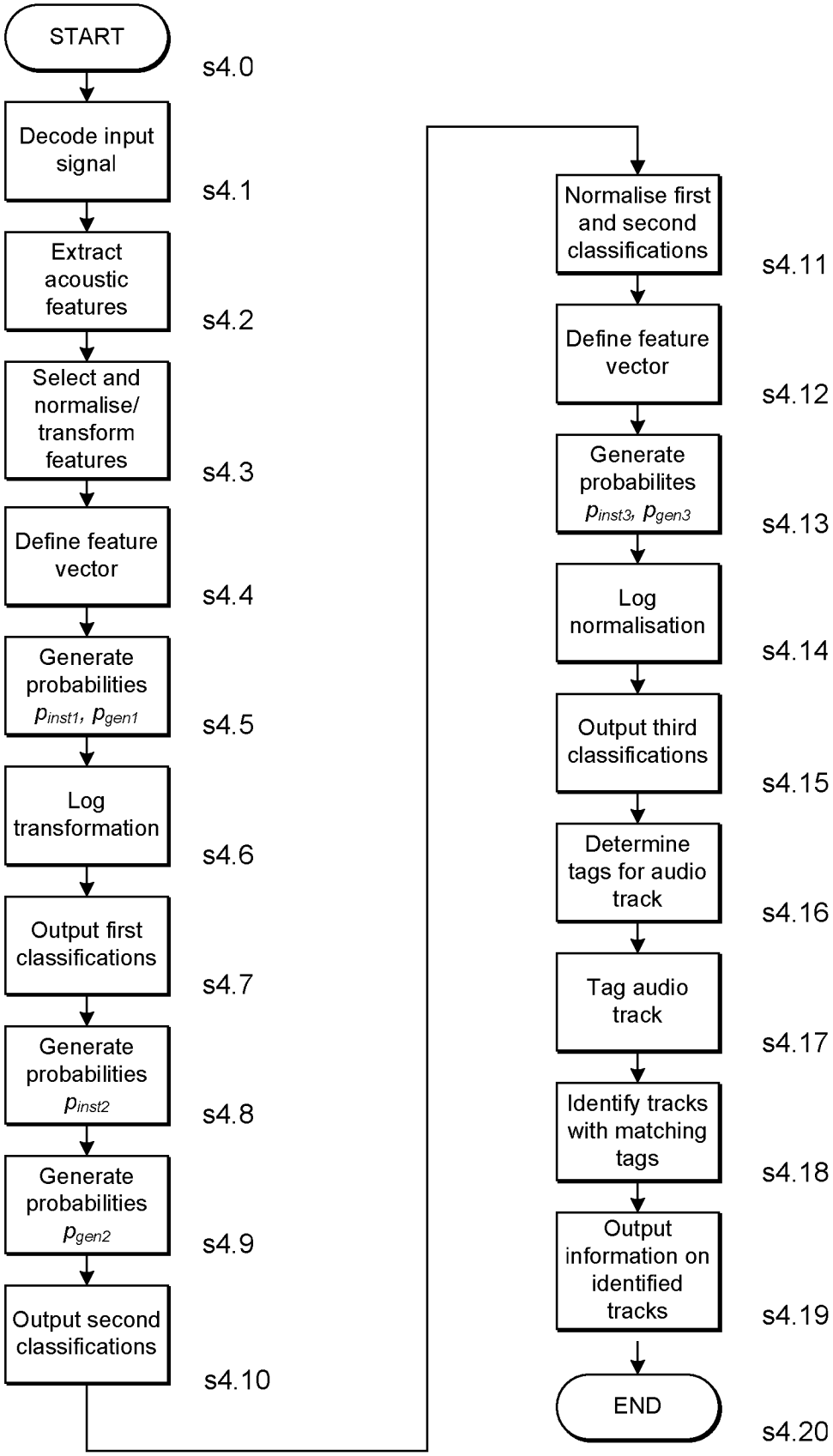
START
s4.0
Decode input signal
s4.1
Extract acoustic features
s4.2
Select and normalise/ transform features
s4.3
Define feature vector
s4.4
Generate probabilities p_{inst1}, p_{gen1}
s4.5
Log transformation
s4.6
Output first classifications
s4.7
Generate probabilities p_{inst2}
s4.8
Generate probabilities p_{gen2}
s4.9
Output second classifications
s4.10
Normalise first and second classifications
s4.11
Define feature vector
s4.12
Generate probabilites p_{inst3}, p_{gen3}
s4.13
Log normalisation
s4.14
Output third classifications
s4.15
Determine tags for audio track
s4.16
Tag audio track
s4.17
Identify tracks with matching tags
s4.18
Output information on identified tracks
s4.19
END
s4.20

Figure 4

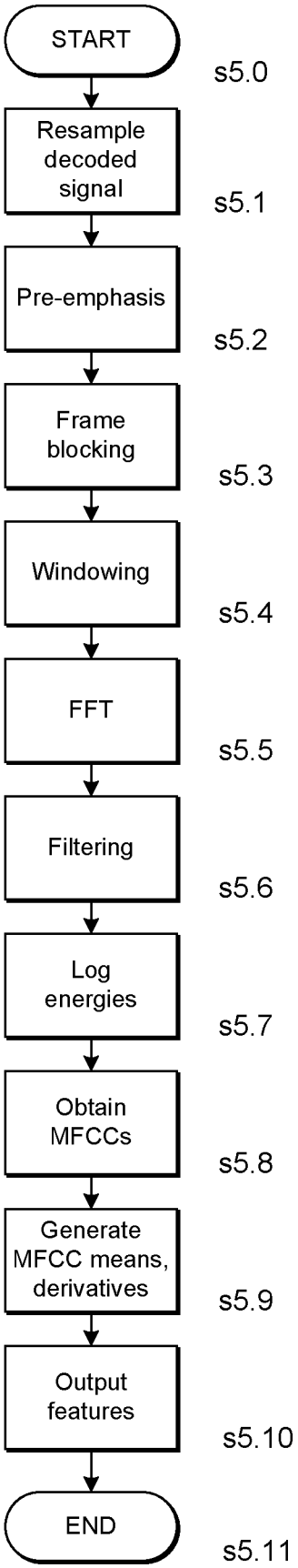
START
s5.0
Resample decoded signal
s5.1
Pre-emphasis
s5.2
Frame blocking
s5.3
Windowing
s5.4
FFT
s5.5
Filtering
s5.6
Log energies
s5.7
Obtain MFCCs
s5.8
Generate MFCC means, derivatives
s5.9
Output features
s5.10
END
s5.11

Figure 5

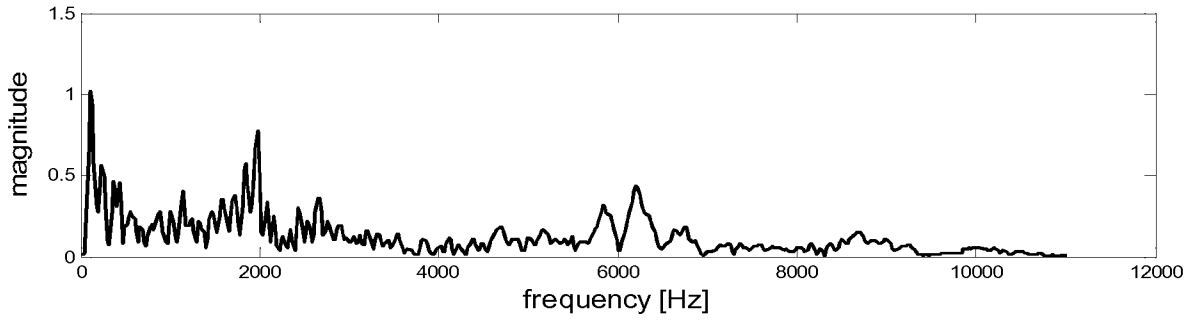

Figure 7

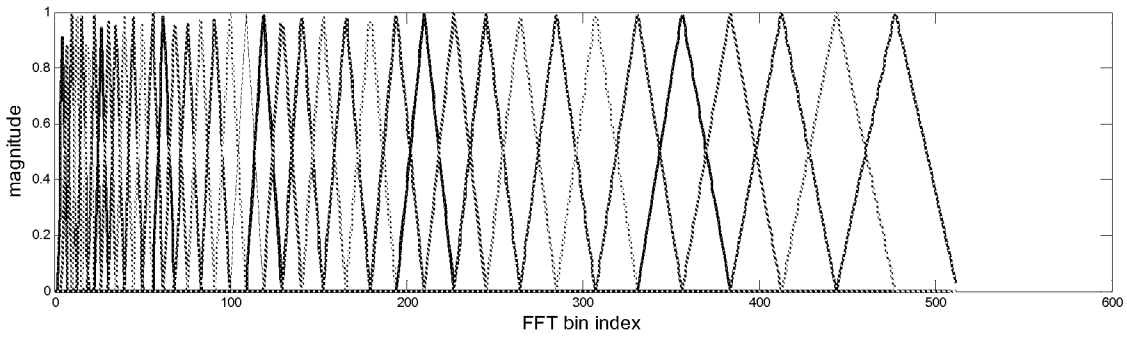

Figure 8

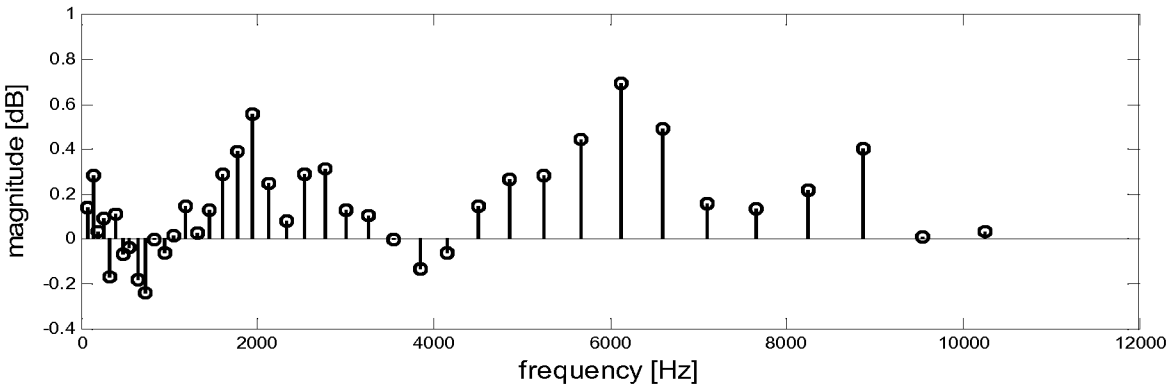

Figure 9

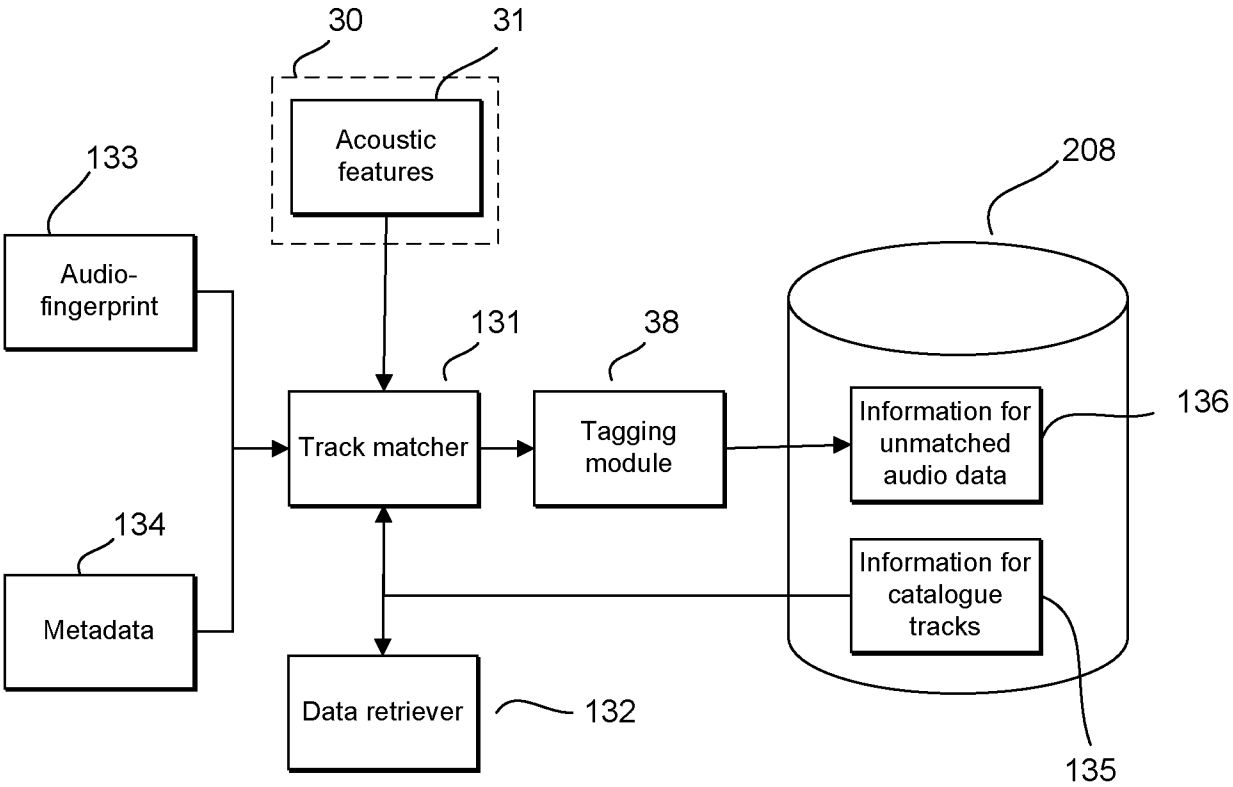

Figure 13

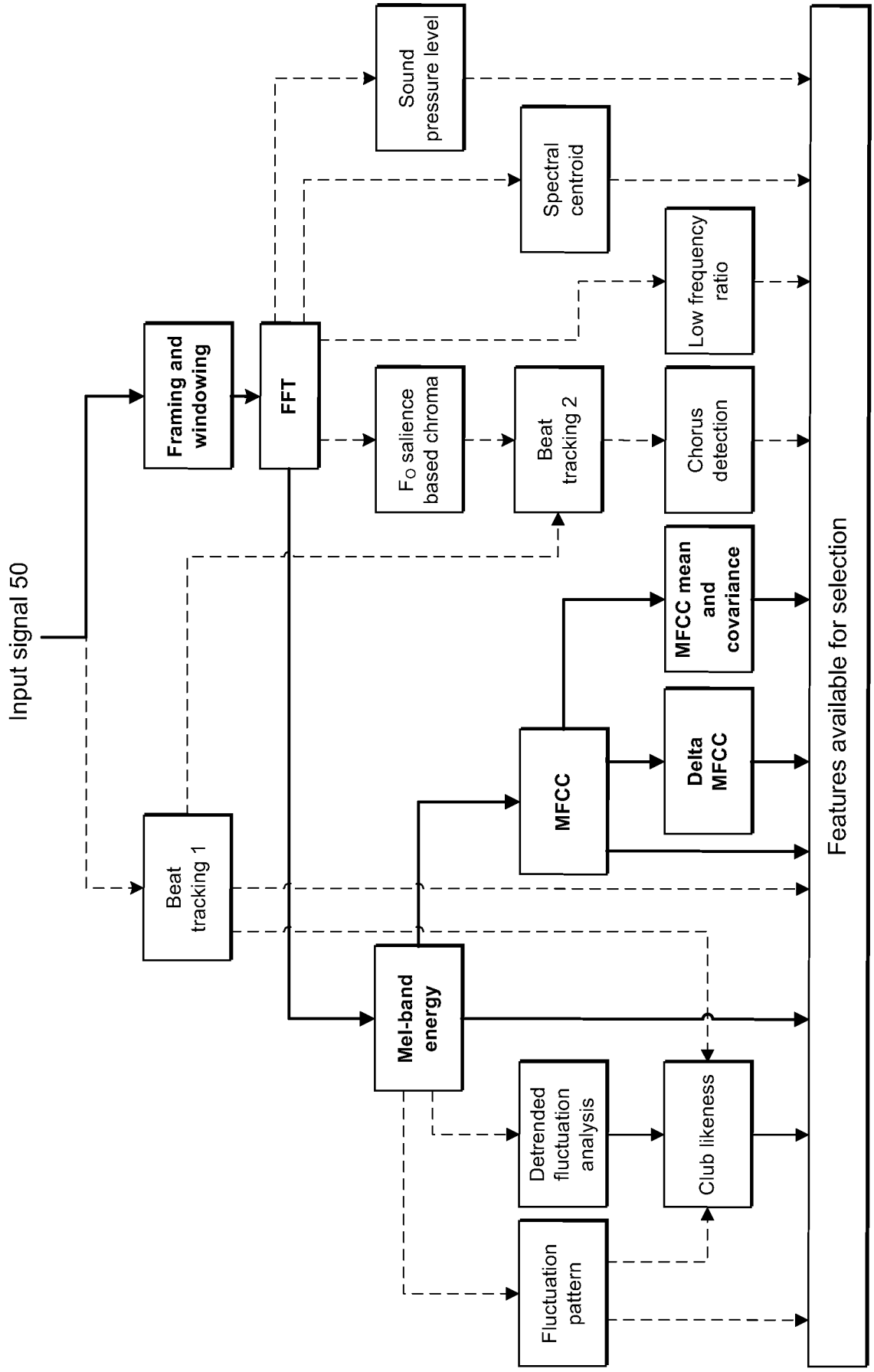
Input signal 50
Framing and windowing
FFT
Sound pressure level
Spectral centroid
Low frequency ratio
F0 salience based chroma
Beat tracking 2
Chorus detection
MFCC mean and covariance
Delta MFCC
MFCC
Beat tracking 1
Mel-band energy
Detrended fluctuation analysis
Club likeness
Fluctuation pattern
Features available for selection

Figure 10

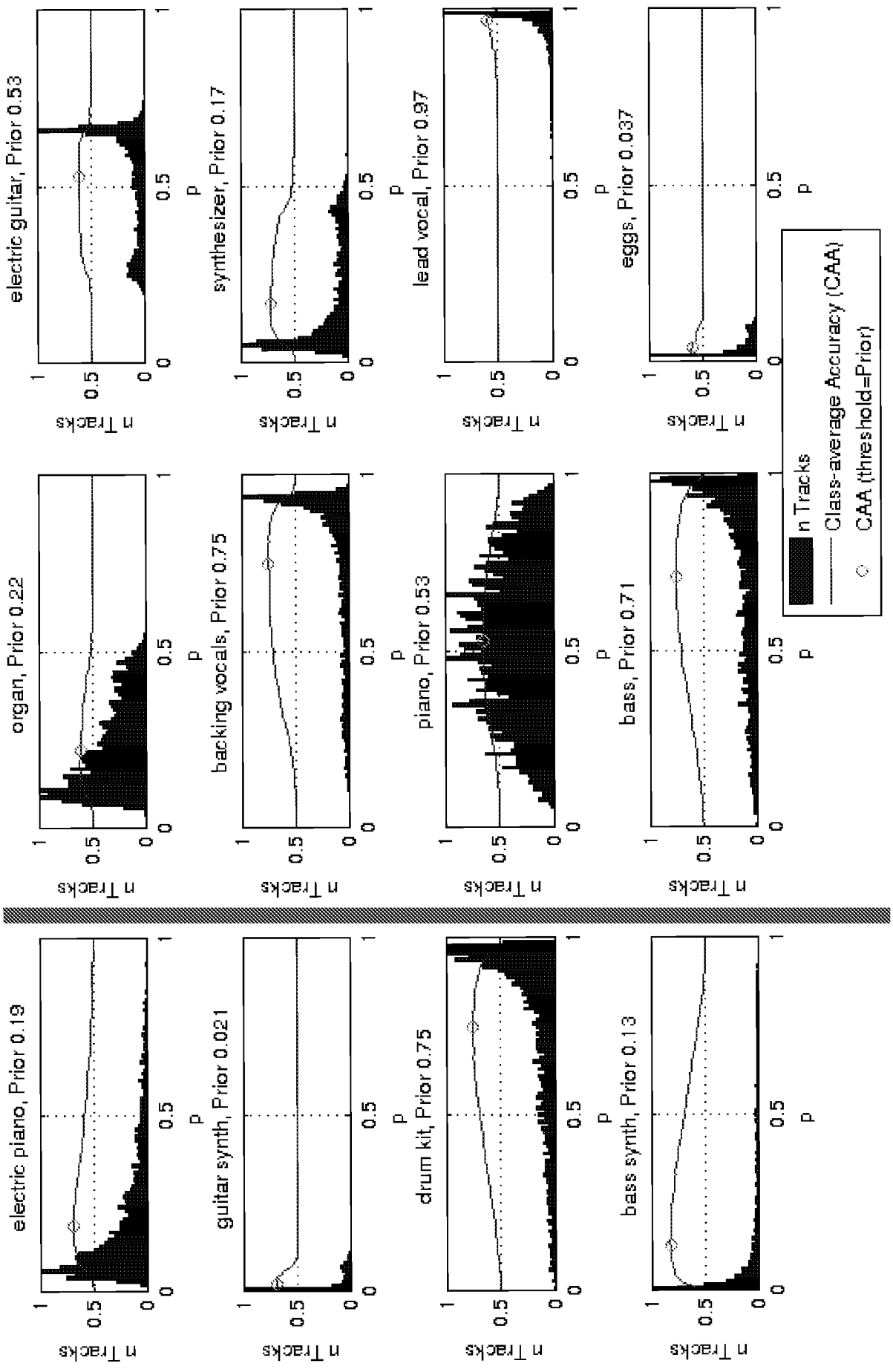
electric guitar, Prior 0.53
synthesizer, Prior 0.17
lead vocal, Prior 0.97
eggs, Prior 0.037
organ, Prior 0.22
backing vocals, Prior 0.75
piano, Prior 0.53
bass, Prior 0.71
electric piano, Prior 0.19
guitar synth, Prior 0.021
drum kit, Prior 0.75
bass synth, Prior 0.13
n Tracks
p
n Tracks
Class-average Accuracy (CAA)
CAA (threshold=Prior)

Figure 11

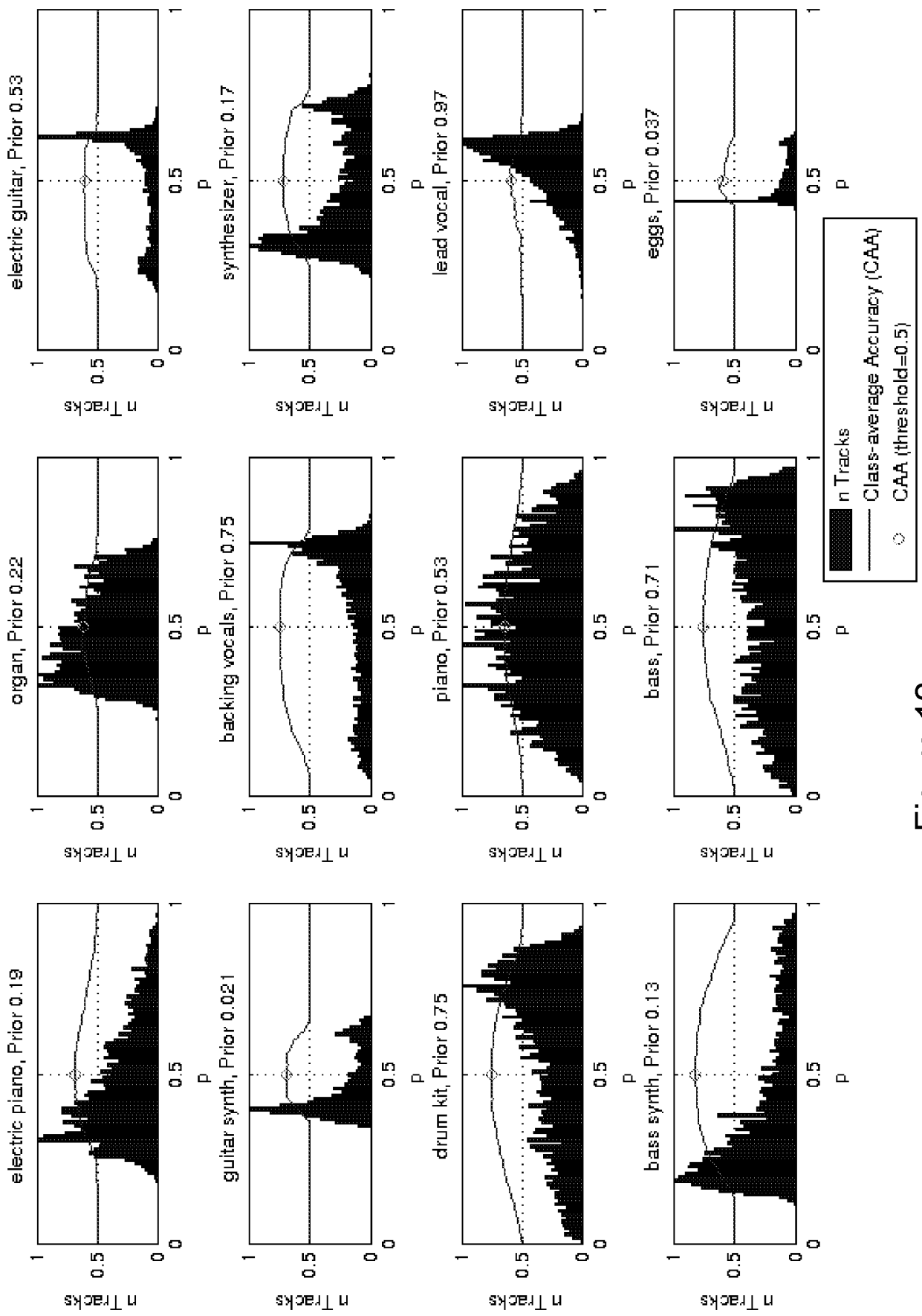
electric guitar, Prior 0.53
synthesizer, Prior 0.17
lead vocal, Prior 0.97
eggs, Prior 0.037
organ, Prior 0.22
backing vocals, Prior 0.75
piano, Prior 0.53
bass, Prior 0.71
electric piano, Prior 0.19
guitar synth, Prior 0.021
drum kit, Prior 0.75
bass synth, Prior 0.13
n Tracks
p
0
0.5
1
n Tracks
Class-average Accuracy (CAA)
CAA (threshold=0.5)

Figure 12

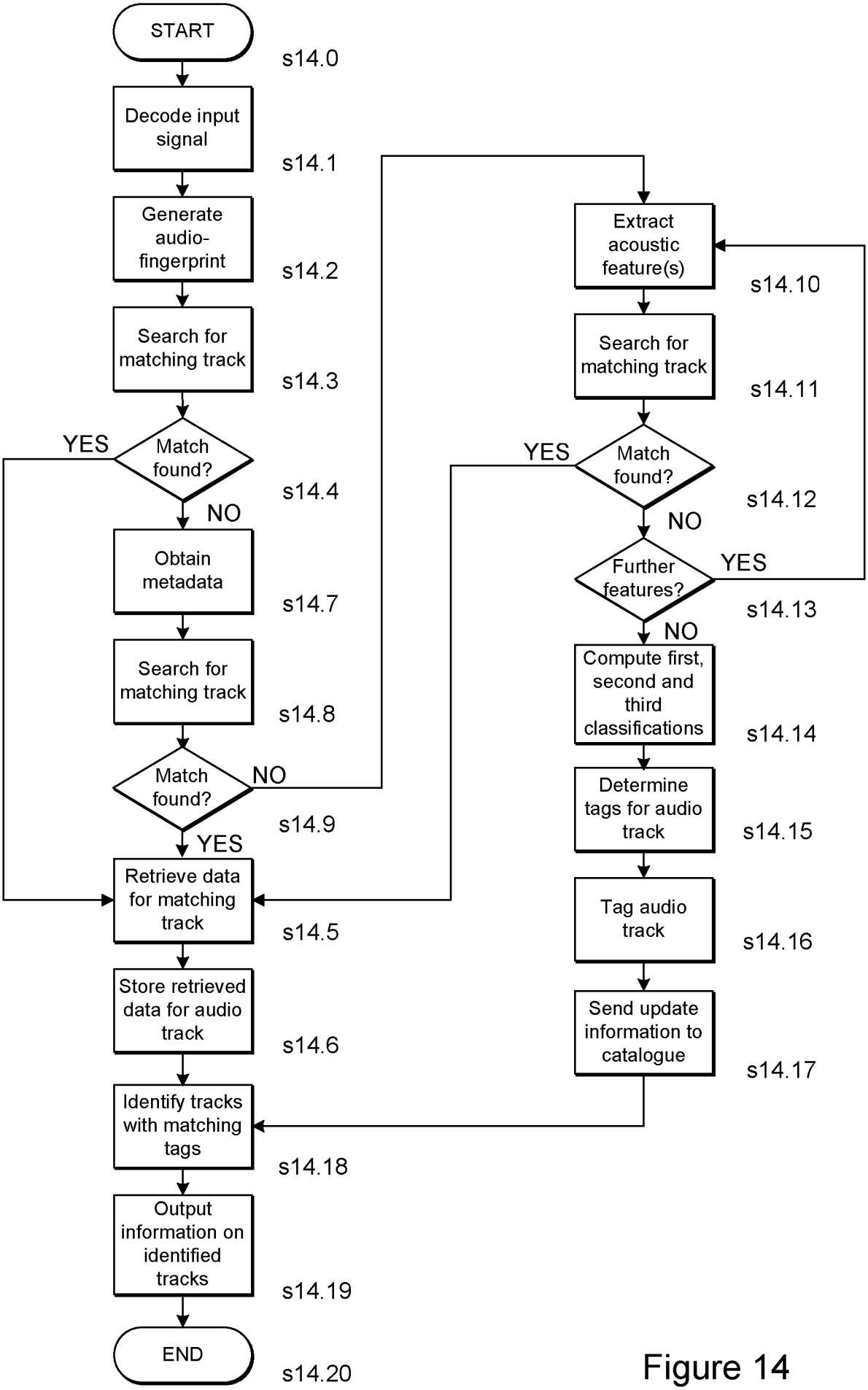
START
s14.0
Decode input signal
s14.1
Generate audio-fingerprint
s14.2
Search for matching track
s14.3
Match found?
YES
s14.4
NO
Obtain metadata
s14.7
Search for matching track
s14.8
Match found?
NO
s14.9
YES
Retrieve data for matching track
s14.5
Store retrieved data for audio track
s14.6
Identify tracks with matching tags
s14.18
Output information on identified tracks
s14.19
END
s14.20
Extract acoustic feature(s)
s14.10
Search for matching track
s14.11
Match found?
YES
s14.12
NO
Further features?
YES
s14.13
NO
Compute first, second and third classifications
s14.14
Determine tags for audio track
s14.15
Tag audio track
s14.16
Send update information to catalogue
s14.17

Figure 14

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI2014/051036

A. **CLASSIFICATION OF SUBJECT MATTER**

See extra sheet

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: G06K, G10H, G10L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

FI, SE, NO, DK

Electronic data base consulted during the international search (name of data base, and, where practicable, search terms used)

EPO-Internal, WPI, Google Scholar, IEEE Xplore

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	WANG, L. et al.: "Music genre classification based on multiple classifier fusion", Fourth Int. Conf. on Natural Computation, 18-20 Oct. 2008, pp. 580-583	
X	abstract; sections 1, 3, 4; Fig. 2	1, 5-11, 14-18, 39
Y	abstract; sections 1, 3, 4; Fig. 2	28, 35, 38
	MORENO-SECO, F. et al.: "Comparison of classifier fusion methods for classification in pattern recognition tasks", Structural, Syntactic and Statistical Pattern Recognition, Lecture Notes in Computer Science Vol. 4109, 2006, pp. 705-713	
X	abstract; sections 1-4	1-5, 8-13, 16-18, 39
Y	abstract; sections 1-4	28, 35, 38

☒ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
27 April 2015 (27.04.2015)	30 April 2015 (30.04.2015)

Name and mailing address of the ISA/FI
Finnish Patent and Registration Office
P.O. Box 1160, FI-00101 HELSINKI, Finland
Facsimile No. +358 9 6939 5328

Authorized officer
Timo Laakso
Telephone No. +358 9 6939 500

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	CANO, P: et al.: "A review of algorithms for audio fingerprinting", IEEE Workshop on Multimedia Signal Processing, 9-11 Dec. 2002, pp. 169-173	
X	abstract; sections I-VI; Figs. 1-3	19-27, 29-34, 36-37, 40
Y	abstract; sections I-VI; Figs. 1-3	28, 35, 38
	REGNIER, L. et al.: "Combining classifications based on local and global features: application to singer identification", Int. Conf. on Digital Audio Effects, Sept. 19-23 2011, pp. 127-134	
A	abstract; sections 2, 3, 4; Fig. 1	1-40
	FU, Z. et al.: " A survey of audio-based music classification and annotation", IEEE Trans. on Multimedia, vol. 13 no. 2, April 2011, pp. 303-319	
A	abstract; sections I-V; Figs. 1-3	1-40
	CHATHURANGA, Y. et al.: "Automatic music genre classification of audio signals with machine learning approaches", GSTF Journal on Computing, vol. 3 no. 2, July 2013, pp. 13-24	
A	the whole document	1-40
	FINE, S. et al.: "Enhancing GMM scores using SVM 'hints'", European Conference on Speech Communication and Technology, 3-7 Sept. 2001, pp. 1757-1760	
A	the whole document	1-40

INTERNATIONAL SEARCH REPORT

International application No.
PCT/FI2014/051036

CLASSIFICATION OF SUBJECT MATTER

IPC
G10L 15/14 (2006.01)
G06K 9/62 (2006.01)