(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 4 471 768 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
04.12.2024 Bulletin 2024/49

(21) Application number: **23176031.5**

(22) Date of filing: **30.05.2023**

(51) International Patent Classification (IPC):
G10L 21/0308 $^{(2013.01)}$ *G10L 25/18* $^{(2013.01)}$

(52) Cooperative Patent Classification (CPC):
G10L 21/0308; G10L 25/18

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC ME MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA
Designated Validation States:
KH MA MD TN

(71) Applicant: **Koninklijke Philips N.V.**
5656 AG Eindhoven (NL)

(72) Inventors:
- **JANSE, Cornelis Pieter**
  Eindhoven (NL)
- **BLOEMENDAL, Brian Brand Antonius Johannes**
  Eindhoven (NL)
- **JANSSEN, Rik Jozef Martinus**
  Eindhoven (NL)

(74) Representative: **Philips Intellectual Property & Standards**
High Tech Campus 52
5656 AG Eindhoven (NL)

(54) ADAPTIVE, FREQUENCY-DOMAIN SPATIAL DECORRELATION OF AUDIO SIGNALS

(57) An audio apparatus comprises a receiver (101) arranged to receive a set of input audio signal which are segmented by a segmenter (103). An output signal generator (105) generates output audio signals by for each segment performing the steps of generating (201) a frequency bin representation of input audio signals; generating (203) a frequency bin representation of the output audio signals as a weighted combination of frequency bin values of the input audio signals for the frequency bin; converting the frequency bin representation to the time domain. An adapter (107) updates the weights for a given weight in response to a correlation measure between output frequency bin values for the output audio signal being generated and an output audio signal linked to the input signal being weighted by the weight. The approach may typically provide improved frequency domain adaptive spatial decorrelation of audio signals.

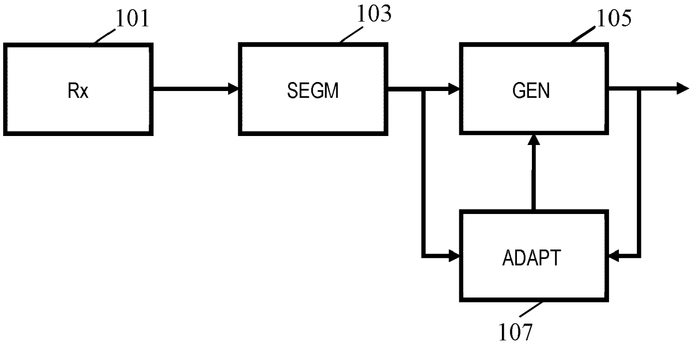

FIG. 1

Processed by Luminess, 75001 PARIS (FR)

Description

FIELD OF THE INVENTION

[0001] The invention relates to an apparatus and a method for generating audio output signals, and in particular, but not exclusively, to generating decorrelated audio signals for e.g., speech signals.

BACKGROUND OF THE INVENTION

[0002] Capturing audio, and in particularly speech, has become increasingly important in the last decades. For example, capturing speech or other audio has become increasingly important for a variety of applications including telecommunication, teleconferencing, gaming, audio user interfaces, etc. However, a problem in many scenarios and applications is that the desired audio source is typically not the only audio source in the environment. Rather, in typical audio environments there are many other audio/noise sources which are being captured by the microphone. Audio processing is often desired to improve the capture of audio, and in particular to post-process the captured audio time interval improve the resulting audio signals. For example, speech enhancement, beamforming, noise attenuation and other functions are often used.

[0003] In many embodiments, audio may be represented by a plurality of different audio signals that reflect the same audio scene or environment. In particular, in many practical applications, audio is captured by a plurality of microphones at different positions. For example, a linear array of a plurality of microphones is often used to capture audio in an environment, such as in a room. The use of multiple microphones allows spatial information of the audio to be captured. Many different applications may exploit such spatial information allowing improved and/or new services.

[0004] However, an issue that in practice may tend to reduce the performance and capture is that different audio sources tend to be captured (differently) in the different microphones depending on the specific properties and position of the audio source relative to the microphones. This may typically make it difficult and resource demanding to process the audio signals and in particular to determine which and how different audio sources contribute to the captured audio for the different microphones. It is often challenging to process multiple captured audio signals that are correlated, and typically with an unknown correlation.

[0005] One approach that may seek to address such issues is to try to separate audio sources by applying beamforming to form beams directed towards the direction of arrival of audio from specific audio sources. However, although this may provide advantageous performance in many scenarios, it is not optimal in all cases. For example, it may not provide optimal source separation in some cases or indeed in some applications such a spatial beamforming does not provide audio properties that are ideal for further processing to achieve a given effect.

[0006] It has been proposed that in some cases where the relationship between audio sources and microphones are known (e.g. by the acoustic impulse response between the audio source and the capture position being known) then this information can be used to generate decorrelated signals from the microphone signal. The known acoustic impulse responses from the different audio sources to the different microphone signals can be used to convert the microphone signals into signals where the audio from individual audio sources are more concentrated in different signals. Such signals may in some cases provided an improved audio or speech capture or processing.

[0007] However, whereas such an operation may be desirable in many situations and scenarios, there are also substantial issues, challenges, and disadvantages. Indeed, in most practical applications the acoustic relationship and specifically the acoustics transfer functions/impulse responses between audio sources and microphones are simply not known. Even small differences between the actual and used values can result in significant degradation in the generated signals and the subsequent processing.

[0008] Hence, an improved approach would be advantageous, and in particular an approach allowing reduced complexity, increased flexibility, facilitated implementation, reduced cost, improved audio capture, improved spatial perception/differentiation of audio sources, improved audio source separation, improved audio/speech application support, improved spatial decorrelation of audio signals, reduced dependency on known or static acoustic properties, improved flexibility and customization to different audio environments and scenarios, and/or improved performance would be advantageous.

SUMMARY OF THE INVENTION

[0009] Accordingly, the Invention seeks to preferably mitigate, alleviate or eliminate one or more of the above mentioned disadvantages singly or in any combination.

[0010] According to an aspect of the invention there is provided an audio apparatus for generating a set of output audio signals, the audio apparatus comprising: a receiver arranged to receive a set of input audio signals; a segmenter arranged to segment the set of input audio signals into time segments; an output signal generator arranged to generate the set of output audio signals, each output audio signal of the set of output audio signals being linked with an input audio signal of the set of input audio signals, the output signal generator being arranged to for each time segment perform the steps of: generating a frequency bin representation of the set of input audio signals, each frequency bin of the frequency bin representation of the set of input audio signals comprising a frequency bin value for each of the input audio signals of the set of input audio signals; generating a frequency bin representation of the set of output audio signals, each frequency bin of the frequency bin representation of a set of output audio signals comprising a frequency bin value for each of the set of output audio signals, the frequency bin value for a given output audio signal of the set of output audio signals for a given frequency bin being generated as a weighted combination of frequency bin values of the set of input audio signals for the given frequency bin; generating a time domain representation for each output audio signal from the frequency bin representation of the set of output audio signals; an adapter arranged to update weights of the weighted combination; wherein the adapter is arranged to update a first weight for a contribution to a first frequency bin value of a first frequency bin for a first output audio signal linked with a first input audio signal from a second frequency bin value of the first frequency bin for a second input audio signal linked to a second output audio signal in response to a correlation measure between a first previous frequency bin value of the first output audio signal for the first frequency bin and a second previous frequency bin value of the second output audio signal for the first frequency bin.

[0011] The approach may provide improved operation and/or performance in many embodiments. It may provide an advantageous generation of output audio signals with typically increased decorrelation in comparison to the input signals. The approach may provide an efficient adaptation of the operation resulting in improved decorrelation in many embodiments. The adaptation may typically be implemented with low complexity and/or resource usage. The approach may specifically apply a local adaptation of individual weights yet achieve an efficient global adaptation.

[0012] The generation of the set of output signals may be adapted to provide increased decorrelation relative to the input signals which in many embodiments and for many applications may provide improved audio processing. For example, in many situations speech enhancement, noise suppression, or beamforming may be improved if such audio processing is performed on audio signals that are less correlated.

[0013] The first and second output audio signals may typically be different output audio signals.

[0014] In accordance with an optional feature of the invention, the adapter is arranged to update the first weight in response to a product of a first value and a second value, the first value being one of the first previous frequency bin value and the second previous frequency bin value and the second value being a complex conjugate of the other of the first previous frequency bin value and the second previous frequency bin value.

[0015] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios.

[0016] In accordance with an optional feature of the invention, the adapter is arranged to update a second weight being for a contribution to the first frequency bin value from a third frequency bin value being a frequency bin value of the first frequency bin for the first input audio signal in response to a magnitude of the first previous frequency bin value.

[0017] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios. It may in particular provide an improved adaptation of the generated output signals. In many embodiments, the updating of the weight reflecting the contribution to an output signal from the linked input signal may be dependent on the signal magnitude/amplitude of that linked input signal. For example, the updating may seek to compensate the weight for the level of the input signal to generate a normalized output signal.

[0018] The approach may allow a normalization/ signal compensation/level compensation to provide e.g., a desired output level.

[0019] In accordance with an optional feature of the invention, the adapter is arranged to set a second weight being for a contribution to the first frequency bin value from a third frequency bin value being a frequency bin value of the first frequency bin for the first input audio signal to a predetermined value.

[0020] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios. It may in many embodiments provide improved adaptation while ensuring convergence of the adaptation towards a non-zero signal level. It may interwork very efficiently with the adaptation of weights that are not for linked signal pairs.

[0021] In many embodiments, the adapter may be arranged to keep the weight constant and with no adaptation or updating of the weight.

[0022] In accordance with an optional feature of the invention, the adapter is arranged to constrain a weight for a contribution to the first frequency bin value from a third frequency bin value being a frequency bin value of the first frequency bin for the first input audio signal to be a real value.

[0023] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios.

[0024] The weights between linked input/output signals may advantageously be determined as/ constrained to be a real valued weight. This may lead to improved performance and adaptation ensuring convergence on a non-zero level solution.

[0025] In accordance with an optional feature of the invention, the adapter is arranged to set a third weight being a weight for a contribution to a fourth frequency bin value of the first frequency bin for the second output audio signal from the first input audio signal to be a complex conjugate of the first weight.

[0026] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios.

[0027] The two weights for two pairs of input/output signals may be complex conjugates of each other in many embodiments.

[0028] In accordance with an optional feature of the invention, weights of the weighted combination for other input audio signals than the first input audio signal are complex valued weights.

[0029] This may provide improved performance and/or operation in many embodiments. The use of complex values for weights for non-linked input signals provide an improved frequency domain operation.

[0030] In accordance with an optional feature of the invention, the adapter is arranged to determine output bin values for the given frequency bin ω from:

$$\mathbf{y}(\omega) = \mathbf{W}(\omega)\mathbf{x}(\omega)$$

where $\mathbf{y}(\omega)$ is a vector comprising the frequency bin values for the output audio signals for the given frequency bin ω; $\mathbf{x}(\omega)$ is a vector comprising the frequency bin values for the input audio signals for the given frequency bin ω; and $\mathbf{W}(\omega)$ is a matrix having rows comprising weights of a weighted combination for the output audio signals.

[0031] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios.

[0032] The matrix $\mathbf{W}(\omega)$ may advantageously be Hermitian. In many embodiments, the diagonal of the matrix $\mathbf{W}(\omega)$ may be constrained to be real values, may be set to a predetermined value(s), and/or may not be updated/adapted but may be maintained as a fixed value. The weights/coefficients outside the diagonal may generally be complex values.

[0033] In accordance with an optional feature of the invention, the adapter is arranged to adapt weights w_{ij} of the matrix $\mathbf{W}(\omega)$ according to:

$$w_{ij}(k+1,\omega) = w_{ij}(k,\omega) - \eta(k,\omega)\left[y_i(k,\omega)\, y_j^*(k,\omega)\right]$$

where i is a row index of the matrix $\mathbf{W}(\omega)$, j is a column index of the matrix $\mathbf{W}(\omega)$, k is a time segment index, ω represents the frequency bin, and $\eta(k, \omega)$ is a scaling parameter for adapting an adaptation speed.

[0034] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios.

[0035] In accordance with an optional feature of the invention, the adapter is arranged to compensate the correlation value for a signal level of the first frequency bin.

[0036] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios. It may allow a compensation of the update rate for signal variations.

[0037] In accordance with an optional feature of the invention, the adapter is arranged to initialize the weights for the weighted combination to comprise at least one zero value weight and one non-zero value weight.

[0038] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios. It may allow a more efficient and/or quicker adaptation and convergence towards advantageous decorrelation. In many embodiments, the matrix $\mathbf{W}(\omega)$ may be initialized with zero values for weights or coefficients for nonlinked signals and fixed non-zero real values for linked signals. Typically, the weights may be set to e.g., 1 for weights on the diagonal and all other weights may initially be set to zero.

[0039] In accordance with an optional feature of the invention, the weighted combination comprises applying a time domain windowing to a frequency representation of weights formed by weights for the first input audio signal and the second input audio signal for different frequency bins.

[0040] This may provide improved performance and/or operation in many embodiments. It may typically provide improved adaptation leading to increased decorrelation of the output audio signals in many scenarios.

[0041] Applying the time domain windowing to the frequency representation of weights may comprise: converting the frequency representation of weights to a time domain representation of weights; applying a window to the time domain representation to generate a modified time domain representation; and converting the modified time domain representation to the frequency domain.

[0042] According to an aspect of the invention, there is provided a method of generating a set of output audio signals: receiving a set of input audio signals; segmenting the set of input audio signals into time segments; generating the set of output audio signals, each output audio signal of the set of output audio signals being linked with one input audio signal of the set of input audio signals, wherein generating the set of output audio signals comprises for each time segment performing the steps of: generating a frequency bin representation of the set of input audio signals, each frequency bin of the frequency bin representation of the set of input audio signals comprising a frequency bin value for each of the input audio signals of the set of input audio signals; generating a frequency bin representation of the set of output audio signals, each frequency bin of the frequency bin representation of a set of output audio signals comprising a frequency bin value for each of the output audio signals, the frequency bin value for a given output audio signal of the set of output audio signals for a given frequency bin being generated as a weighted combination of frequency bin values of the set of input audio signals for the given frequency bin; generating a time domain representation for each output audio signal from the frequency bin representation of the set of output audio signals; and the method further comprises updating weights of the weighted combination including updating a first weight for a contribution to a first frequency bin value of a first frequency bin for a first output audio signal linked with a first input audio signal from a second frequency bin value of the first frequency bin for a second input audio signal linked to a second output audio signal in response to a correlation measure between a first previous frequency bin value of the first output audio signal for the first frequency bin and a second previous frequency bin value of the second output audio signal for the first frequency bin.

[0043] These and other aspects, features and advantages of the invention will be apparent from and elucidated with reference to the embodiment(s) described hereinafter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0044] Embodiments of the invention will be described, by way of example only, with reference to the drawings, in which

FIG. 1 illustrates an example of elements of an apparatus generating a set of output signals from a set of input signals in accordance with some embodiments of the invention;

FIG. 2 illustrates a flowchart of an example of a method of generating a set of output signals from a set of input signals in accordance with some embodiments of the invention;

FIG. 3 illustrates an example of elements of an apparatus generating a set of output signals from a set of input signals in accordance with some embodiments of the invention;

FIG. 4 illustrates an example of a signal flow for an apparatus generating a set of output signals from a set of input signals in accordance with some embodiments of the invention;

FIG. 5 illustrates an example of a signal flow for an apparatus generating a set of output signals from a set of input signals in accordance with some embodiments of the invention;

FIG. 6 illustrates an example of a signal flow for an apparatus generating a set of output signals from a set of input signals in accordance with some embodiments of the invention;

FIG. 7 illustrates an example of a signal flow for an apparatus generating a set of output signals from a set of input signals in accordance with some embodiments of the invention; and

FIG. 8 illustrates some elements of a possible arrangement of a processor for implementing elements of an audio apparatus in accordance with some embodiments of the invention.

DETAILED DESCRIPTION OF SOME EMBODIMENTS OF THE INVENTION

[0045] The following description focuses on embodiments of the invention applicable to audio capturing such as e.g., speech capturing for a teleconferencing apparatus. However, it will be appreciated that the approach is applicable to many other audio signals, audio processing systems, and scenarios for capturing and/or processing audio.

[0046] FIG. 1 illustrates an example of an audio apparatus in accordance with some embodiments of the invention.

[0047] The audio apparatus comprises a receiver 101 which is arranged to receive a set of input audio signals. The input audio signals may be received from different sources, including internal or external sources. In the following an embodiment will be described where the receiver 101 is coupled to a plurality of microphones, such as a linear array of microphones, providing a set of input audio signals in the form of microphone signals.

[0048] The input audio signals are fed to a segmenter 103 which is arranged to segment the set of input audio signals into time segments. In many embodiments, the segmentation may typically be a fixed segmentation into time segments of a fixed and equal duration such as e.g. a division into time segments/intervals with a fixed duration of between 10-20msecs. In some embodiments, the segmentation may be adaptive for the segments to have a varying duration. For example, the input audio signals may have a varying sample rate and the segments may be determined to comprise a fixed number of samples.

[0049] The segmenter 103 is coupled to an audio signal generator 105 which is arranged to generate a set of output audio signals from the input audio signals. In many embodiments, the audio signal generator 105 is arranged to generate the same number of output signals as there are input signals, but it is possible in some embodiments for a different number of output audio signals to be generated.

[0050] The segmentation may typically be into segments with a given fixed number of time domain samples of the input signals. For example, in many embodiments, the segmenter 103 may be arranged to divide the input signals into consecutive segments of e.g., 256 or 512 samples.

[0051] The audio signal generator 105 seeks to generate the output signals to correspond to the input signals but with an increased decorrelation of the signals. The output audio signals are generated to have an increased spatial decorrelation with the cross correlation between audio signals being lower for the output audio signals than for the input audio signals. Specifically, the output signals may be generated to have the same combined energy/power as the combined energy/-power of the input signals (or have a given scaling of this) but with an increased decorrelation (decreased correlation) between the signals. Thus, the output signals may specifically be generated to have decreased coherence (correlation normalized by energy/power) than the input signals.

[0052] The output audio signals may be generated to include all the audio/signal components of the input signals in the output audio signals but with a re-distribution to different signals to achieve increased decorrelation/decreased coherence.

[0053] Each output signal may be created by subtracting/removing correlated source audio components in the input signals from the scaled linked input signal leading to output signals that contain all sources, but show a reduction in mutual correlation.

[0054] Spatial correlation between the input signals may mean that a common (but unknown) signal component can be identified that differs per input signal in amplitude and/or phase. Reducing the correlation at the output may be realized by adding amplitude and/or phase modified input signals from the scaled linked input such that the common input signal component is reduced.

[0055] The audio signal generator 105 is arranged to perform an adaptive spatial decorrelation where the processing is arranged to dynamically adapt the processing and the spatial decorrelation to reflect properties of the captured audio and specifically to e.g. adapt to the acoustic environment and the relationships between the audio sources and microphone signals. Accordingly, the audio apparatus further comprises an adapter 107 which is arranged to adapt the spatial filtering.

[0056] The audio signal generator 105 may typically be arranged to apply a frequency domain based cross-signal spatial filtering to the input signals to generate the audio signals with the coefficients of the filtering being adapted based on signal properties, and specifically based on signal properties of the output signals. In particular, for each segment, a set of input signal samples may be converted into a set of output signal samples based on a spatial filtering, and a set of update values for the spatial filtering may be determined by the adapter 107. The spatial filtering is then updated for the next segment based on the update values.

[0057] FIG. 2 illustrates a method that may be performed by the audio signal generator 300.

[0058] In step 201, the audio signal generator 105 is arranged to generate a frequency bin representation of the set of input audio signals. The audio signal generator 105 is arranged to perform a frequency domain processing of a frequency domain representation of the input audio signals. The signal representation and processing are based on frequency bins and thus the signals are represented by values of frequency bins and these values are processed to generate frequency bin values of the output signals. In many embodiments, the frequency bins have the same size, and thus cover frequency intervals of the same size. However, in other embodiments, frequency bins may have different bandwidths, and for example a perceptually weighted bin frequency interval may be used.

[0059] In some embodiments, the input audio signals may already be provided in a frequency representation and no further processing or operation is required. In some such cases, however, a rearrangement into suitable segment representations may be desired, including e.g. using interpolation between frequency values to align the frequency representation to the time segments.

[0060] In other embodiments, a filter bank, such as a Quadrature Mirror Filter, QMF, may be applied to the time domain input signals to generate the frequency bin representation. However, in many embodiments, a Discrete Fourier Transform (DFT) and specifically a Fast Fourier Transform, (FFT) may be applied to generate the frequency representation.

[0061] Step 201 is followed by step 203 in which the audio signal generator 105 generates a frequency bin representation of the set of output signals. The processing of the input signals is typically performed in the frequency domain and for each frequency bin, an output frequency bin value is generated from one or more input frequency bin values of one or more input signals as will be described in more detail in the following. The output signals are generated to (typically/on average) reduce the correlation between signals relative to the correlation of the input signals.

[0062] The audio signal generator 105 is arranged to filter the input audio signals. The filtering is a spatial filtering in that for a given output signal, the output value is determined from a plurality of, and typically all of the input audio signals (for the same time/segment and for the same frequency bin). The spatial filtering is specifically performed on a frequency bin basis such that a frequency bin value for a given frequency bin of an output signal is generated from the frequency bin values of the input signals for that frequency bin. The filtering/weighted combination is across the signals rather than being a typical time/frequency filtering.

[0063] Specifically, the frequency bin value for a given frequency bin is determined as the weighted combination of the frequency bin values of the input signals for that frequency bin. The combination may specifically be a summation and the frequency bin value may be determined as a weighted summation of the frequency bin values of the input signals for that frequency bin. The determination of a bin value for a given frequency bin may be determined as the vector multiplication of a vector of the weights/coefficients of the weighted summation and a vector comprising the bin values of the input signals:

$$[y] = \begin{bmatrix} w_1 & w_2 & w_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

where y is the output bin value, w_1- w_3 are the weights of the weighted combination, and x_1- x_3 are the input signal bin values.

[0064] Representing the output bin values for a given frequency bin ω as a vector **y**(ω), the determination of the output signals may be determined as:

$$\boldsymbol{y}(\omega) = \boldsymbol{W}(\omega)\boldsymbol{x}(\omega)$$

where the matrix **W**(ω) represents the weights/coefficients of the weighted summation for the different output signals and **x** (ω) is a vector comprising the input signal values.

[0065] For example, for an example with only three input signals and output signals, the output bin values for the frequency bin ω may be given by

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

where y_n represents the output bin values, w_{ij} represents the weights of the weighted combinations, and x_m represents the input signal bin values.

[0066] Step 203 is followed by step 205 in which a time domain representation of each output signal is generated from the frequency bin representation of the output signals as generated in step 203.

[0067] Typically, the reverse operation of the conversion from the time domain to the frequency domain is applied, e.g. an IFFT or iQMF operation is performed. The resulting output is accordingly a set of time domain output signals that represents the audio of the input audio signals but with increased decorrelation between the signals.

[0068] Step 205 is followed by step 207 wherein the adapter 107 is arranged to determine update values for the weights of the weighted combination(s). Thus, specifically, in step 207, update values may be determined for the matrix ***W***(ω).

[0069] The method may then return to step 201 via step 209 to process the next segment. Step 209 may initiate the processing of the next step including updating the weights of the weighted combination based on the update values.

[0070] The adapter 107 is arranged to apply an adaptation approach that determines update values which may allow the output signals to represent the audio of the input signals but with the output signals typically being more decorrelated than the input signals.

[0071] The adapter 107 is arranged to use a specific approach of adapting the weights based on the generated output signals. The operation is based on each output audio signal being linked with one input audio signal. The exact linking between output signals and input signals is not essential and many different (including in principle random) linkings/pairings of each output signal to an input signal may be used. However, the processing is different for a weight that reflects a contribution from an input signal that is linked to/paired with the output signal for the weight than the processing for a weight that reflects a contribution from an input signal that is not linked to/paired with the output signal for the weight. For example, in some embodiments, the weights for linked signals (i.e. for input signals that are linked to the output signal generated by the weighted combination that includes the weight) may be set to a fixed value and not updated, and/or the weights for linked signals may be restricted to be real valued weights whereas other weights are generally complex values.

[0072] The adapter 107 uses an adaptation/update approach where an update value is determined for a given weight that represents the contribution to a bin value for a given output signal from a given non-linked input signal based on a correlation measure between the output bin value for the given output signal and the output bin value for the output signal that is linked with the given (non-linked) input signal. The update value may then be applied to modify the given weight in the subsequent segment, or the update value for weight in a given segment is determined in response to two output bin values of a (and typically the immediate) prior segment, where the two output values represent respectively the input signal and the output signal to which the weight relate.

[0073] The described approach is typically applied to a plurality, and typically all, of the weights used in determining the output bin values based on a non-linked input signal. For weights relating to the input signal that is linked to the output signal for the weight, other considerations may be used, such as e.g. setting the weight to a fixed value as will be described in more detail later.

[0074] Specifically, the update value may be determined in dependence on a product of the output bin value for the weight and the complex conjugate of the output bin value linked to the input signal for the weight, or equivalently in dependence on a product of the complex conjugate of the output bin value for the weight and the output bin value linked to the input signal for the weight.

[0075] As a specific example, an update value for segment k+1 for frequency bin ω may be determined in dependence on the correlation measure given by:

$$\left[y_i(k,\omega)\ y_j^*(k,\omega)\right]$$

or equivalently by:

$$\left[\ y_i^*(k,\omega)y_j(k,\omega)\ \right]$$

where $y_i(k,\omega)$ is the output bin value for the output signal i being determined based on the weight; and $y_j(k,\omega)$ is the output bin value for the output signal j that is linked to the input signal from which the contribution is determined (i.e. the input signal bin value that is multiplied by the weight to determine a contribution to the output bin value for signal i).

[0076] The measure of $\left[y_i(k,\omega)\ y_j^*(k,\omega)\right]$ (or the conjugate value) indicates the correlation of the time domain signal in the given segment. In the specific example, this value may then be used to update and adapt the weight $w_{i,j}$ (k+1,ω).

[0077] As previously mentioned, the audio signal generator 105 may be arranged to determine the output bin values for the output signals for the given frequency bin ω from:

$$\mathbf{y}(\omega)\ =\ \mathbf{W}(\omega)\mathbf{x}(\omega)$$

where **y**(ω) is a vector comprising the frequency bin values for the output signals for the given frequency bin ω; **x**(ω) is a vector comprising the frequency bin values for the input audio signals for the given frequency bin ω; and **W**(ω) is a matrix having rows comprising weights of a weighted combination for the output audio signals.

[0078] In the example, adapter 107 may specifically be arranged to adapt at least some of the weights w_{ij} of the matrix **W** (ω) according to:

$$w_{ij}(k+1,\omega) = w_{ij}(k,\omega)\ -\ \eta(k,\omega)\left[y_i(k,\omega)\ y_j^*(k,\omega)\right]$$

where i is a row index of the matrix **W**(ω) , j is a column index of the matrix **W**(ω) , k is a time segment index, ω represents the frequency bin, and $\eta(k,\omega)$ is a scaling parameter for adapting an adaptation speed. Typically, the adapter 107 may be arranged to adapt all the weights that are not relating the input signal with the linked output signal (i.e. "cross-signal" weights).

[0079] In some embodiments, the adapter 107 may be arranged to adapt the update rate/speed of the adaptation of the weights. For example, in some embodiments, the adapter may be arranged to compensate the correlation measure for a given weight in dependence on the signal level of the output bin value to which a contribution is determined by the weight.

[0080] As a specific example, the compensation value $\left[y_i(k,\omega)\ y_j^*(k,\omega)\right]$ may be compensated by the signal level of the output bin value, $|y_i(k,\omega)|$. The compensation may for example be included to normalize the update step values to be less dependent on the signal level of the generated decorrelated signals.

[0081] In many embodiments, such a compensation or normalization may specifically be performed on a frequency bin basis, i.e. the compensation may be different in different frequency bins. This may in many scenarios improve the operation and may typically result in an improved adaptation of the weights generating the decorrelated signals.

[0082] The compensation may for example be built into the scaling parameter $\eta(k,\omega)$ of the previous update equation. Thus, in many embodiments, the adapter 107 may be arranged to adapt/change the scaling parameter $\eta(k,\omega)$ differently in different frequency bins.

[0083] In many embodiments, the arrangement of the input signal vector **x**(ω) the output signal vector γ(ω) is such that linked signals are at the same position in the respective vectors, i.e. specifically y_1 is linked with x_1, y_2 with x_2, y_3 is with x_3, etc. In this case, the weights for the linked signals are on the diagonal of the weight matrix W(*k*, ω). The diagonal values may in many embodiments be set to a fixed real value, such as e.g. specifically be set to a constant value of 1.

[0084] In many embodiments, the weights/spatial filters/weighted combinations may be such that the weights for a contribution to a first output signal from a first input signal (not linked to the first output signal) is a complex conjugate of the contribution to a second output signal being linked with the first input signal from a second input signal being linked with the first input signal. Thus, the two weights for two pairs of linked input/output signals are complex conjugates.

[0085] In the example of the weights for linked input and output signals being arranged on the diagonal of the weight matrix **W**(ω), this results in a Hermitian matrix. Indeed, in many embodiments, the weight matrix **W**(ω) is a Hermitian matrix. Specifically, the coefficients/weights of the weight matrix **W**(ω) may meet the criterion:

$$w_{ij} = w_{ji}^{*}.$$

[0086] As previously mentioned, the weights for the contributions to an output signal bin value from the linked input signal (corresponding to the values of the diagonal of the weight matrix **W**(ω) in the specific example) are treated differently than the weights for non-linked input signals. The weights for linked input signals will in the following for brevity also be referred to as linked weights and the weights for non-linked input signals will in the following for brevity also be referred to as non-linked weights, and thus in the specific example the weight matrix W(ω) will be a Hermitian matrix comprising the linked weights on the diagonal and the non-linked weights outside the diagonal.

[0087] In many approaches, the adaptation of the non-linked weights is such that it seeks to reduce the correlation measures. Specifically, each update value may be determined to reduce the correlation measure. Overall, the adaptation will accordingly seek to reduce the cross-correlations between the output signals. However, the linked weights are determined differently to ensure that the output signals maintain a suitable audio energy/power/level. Indeed, if the linked weights where instead adapted to seek to reduce the autocorrelation of the output signal for the weight, there is a high risk that the adaptation would converge on a solution where all weights, and thus output signals, are essentially zero (as indeed this would result in the lowest correlations). Further, the audio apparatus is arranged to seek to generate signals with less cross-correlation but do not seek to reduce autocorrelation.

[0088] Thus, in many embodiments, the linked weights may be set to ensure that the output signals are generated to have a desirable (combined) energy/power/level.

[0089] In some cases, the adapter 107 may be arranged to adapt the linked weights, and in other cases the adapter may be arranged to not adapt the linked weights.

[0090] For example, in some embodiments, the linked weights may simply be set to a fixed constant value that is not adapted. For example, in many embodiments, the linked weights may be set to a constant scalar value, such as specifically to the value 1 (i.e. a unitary gain being applied for a linked input signal). For example, the weights on the diagonal of the weight matrix **W**(ω) may be set to 1.

[0091] Thus, in many embodiments, the weight for a contribution to a given output signal frequency bin value from a linked input signal frequency bin value may be set to a predetermined value. This value may in many embodiments be maintained constant without any adaptation.

[0092] Such an approach has been found to provide very efficient performance and may result in an overall adaptation that has been found to provide output signals to be generated that provide a highly accurate representation of the original audio of the input signals but in a set of output signals that have increased decorrelation.

[0093] In some embodiments, the linked weights may also be adapted but will be adapted differently than the non-linked weights. In particular, in many embodiments, the linked weights may be adapted based on the output signals.

[0094] Specifically, in many embodiments, a linked weight for a first input signal and linked output signal may be adapted based on the generated output bin value of the linked audio signal, and specifically based on the magnitude of the output bin value.

[0095] Such an approach may for example allow a normalization and/or setting of the desired energy level for the signals.

[0096] In many embodiments, the linked weights are constrained to be real valued weights whereas the non-linked weights are generally complex values. In particular, in many embodiments, the weight matrix **W**(ω) may be a Hermitian matrix with real values on the diagonal and with complex values outside of the diagonal.

[0097] Such an approach may provide a particular advantageous operation and adaptation in many scenarios and embodiments. It has been found to provide a highly efficient spatial decorrelation while maintaining a relatively low complexity and computational resource.

[0098] The adaptation may gradually adapt the weights to increase the decorrelation between signals. In many embodiments, the adaptation may be arranged to converge towards a suitable weight matrix **W**(ω) regardless of the initial values and indeed in some cases the adaptation may be initialized by random values for the weights.

[0099] However, in many embodiments, the adaptation may be started with advantageous initial values that may e.g. result in faster adaptation and/or result in the adaptation being more likely to converge towards more optimal weights for decorrelating signals.

[0100] In particular, in many embodiments, the weight matrix **W**(ω) may be arranged with a number of weights being zero but at least some weights being non-zero. In many embodiments, the number of weights being substantially zero may be no less than 2,3,5, or 10 times the number of weights that are set to a non-zero value. This has been found to tend to provide improved adaptation in many scenarios.

[0101] In particular, in many embodiments, the adapter 107 may be arranged to initialize the weights with linked weights being set to a non-zero value, such as typically a predetermined non-zero real value, whereas non-linked weights are set to substantially zero. Thus, in the above example where linked signals are arranged at the same positions in the vectors, this will result in an initial weight matrix **W**(ω) having non-zero values on the diagonal and (substantially) zero values outside of the diagonal.

[0102] Such initializations may provide particularly advantageous performance in many embodiments and scenarios. It may reflect that due to the audio signals typically representing audio at different positions, there is a tendency for the input signals to be somewhat decorrelated. Accordingly, a starting point that assumes the input signals are fully correlated is often advantageous and will lead to a faster and often improved adaptation.

[0103] It will be appreciated that the weights, and specifically the non-linked weights, may not necessarily be exactly zero but may in some embodiments be set to low values close to zero. However, the initial non-zero values may be at least 5,10,20 or 100 times higher than the initially substantially zero values.

[0104] The described approach may provide a highly efficient adaptive spatial decorrelator that may generate output signals that represent the same audio as the input signals but with increased decorrelation. The approach has in practice been found to provide a highly efficient adaptation in a wide variety of scenarios and in many different acoustic environments and for many different audio sources. For example, it has been found to provide a highly efficient decorrelation of speaker signals in environments with multiple speakers.

[0105] The adaptation approach is furthermore computationally efficient and in particular allows localized and individual adaptation of individual weights based only on two signals (and specifically on only two frequency bin values) closely related to the weight, yet the process results in an efficient and often substantially optimized global optimization of the spatial filtering, and specifically the weight matrix **W**(ω) . The local adaptation has been found to lead to a highly advantageous global adaptation in many embodiments.

[0106] A particular advantage of the approach is that it may be used to decorrelate convolutive mixtures and is not limited to only decorrelate instantaneous mixtures. For a convolutive mixture, the full impulse response determines how the signals from the different audio sources combine at the microphones (i.e. the delay/timing characteristics are significant) whereas for instantaneous mixes a scalar representation is sufficient to determine how the audio sources combine at the microphones (i.e. the delay/timing characteristics are not significant). By transforming a convolutive mixture into the frequency domain, the mixture can be considered a complex-valued instantaneous mixture per frequency bin.

[0107] In the following a more detailed and mathematical description of an example of a spatial decorrelator following some or all of the above described principles will be provided. In the example, linked input signals and output signals are denoted by the same index and position in the input and output vectors **x**(ω), **y**(ω) and a de-mixing/weight matrix **W**(ω) is determined for which the linked weights are on the diagonal.

[0108] The exemplary adaptive spatial decorrelator (SDC) works in the frequency domain, and a robust simple learning rule is applied for each single frequency bin to achieve decorrelation for that frequency bin. Yet, the individual and local adaptation results in an overall adaptation of the spatial decorrelation.

[0109] A solution in the frequency domain has the advantage that convolutions and correlations in time domain are replaced by element-wise multiplications. Significantly, a frequency domain approach as described may also allow convolutive mixtures to be considered instantaneous mixtures (per frequency bin) in the frequency domain.

[0110] If we consider input signals being received from a microphone array with *Nmics* microphones, a single noise point interferer *s*(ω), transfer functions $h_i(\omega)$ and microphone signals $x_i(\omega)$, with i the microphone number, we can write:

$$x_i(\omega) = h_i(\omega)s(\omega)$$

[0111] For a pair of microphones *p* and q we can now write for the cross power spectrum (the frequency domain equivalent of the cross-correlation):

$$C_{pq}(\omega) = \mathbb{E}\{x_p(\omega)x_q^*(\omega)\},$$

with (.)* the conjugate operator.

[0112] Let s(k) be white again, with for all ω:

$$C_{ss}(\omega) = E\{s(\omega)s^*(\omega)\} = 1$$

[0113] We can rewrite C_{pq} (ω) as:

$$C_{pq}(\omega) = \mathbb{E}\{h_p(\omega)s(\omega)s^*(\omega)h_q^*(\omega)\}$$

$$= \mathbb{E}\{s(\omega)s^*(\omega)\}h_p(\omega)h_q^*(\omega)$$

$$= h_p(\omega)h_q^*(\omega)$$

(Eq. 1)

[0114] More generally, for a convolutive mixture with *nmics* sources we get, using matrix notation with

$$\boldsymbol{s}(\omega) = \left(\mathrm{s}_1(\omega)..\mathrm{s}_{\mathrm{nmics}}(\omega)\right)^{\mathrm{T}} \text{ and } \boldsymbol{x}(\omega) = \left(x_1(\omega)..x_{nmics}(\omega)\right)^T :$$

$$\boldsymbol{x}(\omega) = \boldsymbol{H}(\omega)\boldsymbol{s}(\omega),$$

where ***H***(ω) is the so-called *nmics x nsources* mixing matrix.

[0115] Now we can determine an *nmics x nmics* covariance matrix $\boldsymbol{R}_{xx}(\omega)$ defined as:

$$\boldsymbol{R}_{xx}(\omega) = \mathbb{E}\{\boldsymbol{x}(\omega)\boldsymbol{x}^H(\omega)\},$$

with H conjugate transpose.

[0116] The microphone signals $x_i(\omega)$ are uncorrelated, if and only if the off-diagonal elements of $\boldsymbol{R}_{xx}(\omega)$ are all zero. When correlated, with non-zero off-diagonal elements, we need a transformation that transforms ***x***(ω) such, that the output signals are decorrelated. This can be done with an *nmics x nmics* de-mixing matrix ***W***(ω) such that the output

$$\boldsymbol{y}(\omega) = \boldsymbol{W}(\omega)\boldsymbol{x}(\omega),$$

has a covariance matrix $\boldsymbol{R}_{yy}(\omega)$defined as:

$$\boldsymbol{R}_{yy}(\omega) = \mathbb{E}\{\boldsymbol{y}(\omega)\boldsymbol{y}^H(\omega)\},$$

with all off-diagonal elements equal to zero.
We can rewrite $\boldsymbol{R}_{yy}(\omega)$ as:

$$\boldsymbol{R}_{yy}(\omega) = \mathbb{E}\{\boldsymbol{W}(\omega)\boldsymbol{x}(\omega)\boldsymbol{x}^H(\omega)\boldsymbol{W}^H(\omega)\}$$

$$= \boldsymbol{W}(\omega)\boldsymbol{R}_{xx}(\omega)\boldsymbol{W}^H(\omega)$$

$$= \boldsymbol{\Lambda}(\omega),$$

with Λ(ω) a diagonal matrix with elements $\lambda_1(\omega)..\lambda_{\mathrm{nmics}}(\omega)$ that are real. Often Λ(ω) is chosen as Λ(ω) = *I*, the identity matrix. While this is a good choice, when you have multiple input choices $s_i(\omega)$, this is not the best choice when you have a single, or a dominant input noise source.

[0117] From the above, we have for the (ij)'th element of $\boldsymbol{R}_{xx}(\omega)$ for a single source s(ω):

$$r_{xx,ij}(\omega) = \mathbb{E}\{s(\omega)s^*(\omega)\}h_i(\omega)h_j^*(\omega)$$

$$= \rho_{ss}^2(\omega)h_i(\omega)h_j^*(\omega)$$

[0118] We introduce $\boldsymbol{R}_{xx}(\omega) = \rho_{ss}^2\tilde{\boldsymbol{R}}_{xx}(\omega)$, with $\tilde{R}_{xx}(\omega)$ the coherence matrix, a normalized covariance matrix. The elements of $\tilde{\boldsymbol{R}}_{xx}(\omega)$ only depend on the transfer functions $h_{i,j}(\omega)$, not on the signal characteristics of the source $s(\omega)$. This means that with $\boldsymbol{W}(\omega)\tilde{\boldsymbol{R}}_{xx}(\omega)\boldsymbol{W}^H(\omega)$ = **I,** also ***W***(ω) will only depend on the transfer functions $h_{i,j}(\omega)$. This is advantageous, since the changes in the acoustic paths are normally much slower, when compared to the changes in the spectrum of the noise source (assuming non-stationarity). Since we will not use the coherence matrix directly, we will get the same effect if we choose $\mathbf{\Lambda}(\omega) = \rho_{ss}^2(\omega)\mathbf{I}$, since:

$$\boldsymbol{R}_{yy}(\omega) = \boldsymbol{W}(\omega)\boldsymbol{R}_{xx}(\omega)\boldsymbol{W}^H(\omega)$$
$$= \rho_{ss}^2(\omega)\boldsymbol{W}(\omega)\widetilde{\boldsymbol{R}}_{xx}(\omega)\boldsymbol{W}^H(\omega)$$
$$= \rho_{ss}^2\mathbf{I}$$

[0119] Later-on, we will see that the update rule is simplified when we choose $\Lambda(\omega) = \alpha(\omega)\rho_{ss}^2(\omega)\mathrm{I}$, with $\alpha(\omega)$ a scalar independent of $\rho_{ss}^2(\omega)$.

[0120] The decorrelation/weight matrix **W** may be symmetric positive definite. Symmetric means $w_{ij} = w_{ji}$ and *if* R_{xx} is positive definite (all eigenvalues should be positive, which is normally the case, certainly when there is some uncorrelated (sensor)noise present), then, with **W** symmetric, **W** will also be positive definite (Suppose the real positive eigenvalues are $\lambda_1 \ldots . . \lambda_{nmics}$, then the eigenvalues of **W** are $\frac{1}{\sqrt{\lambda_1}} \ldots . \frac{1}{\sqrt{\lambda_{nmics}}}$).

[0121] An adaptation/learning approach/ rule may be based on the equation:

$$\boldsymbol{W}(k+1) = \boldsymbol{W}(k) + \eta(k)\left(\Lambda - \boldsymbol{y}(k)\boldsymbol{y}^T(k)\right),$$

or in scalar form:

$$w_{ij}(k+1) = w_{ij}(k) + \eta(k)\left[\delta_{ij}\lambda_i - y_i(k)y_j(k)\right]$$

with k the time index, $\eta(k)$ the step size or learning rate, and δ_{ij} the Kronecker symbol. This learning rule may be called a local learning rule, since for calculating $w_{ij}(k+1)$ we only need the signals y_i(k) and $y_j(k)$. Note that when at initialization $\boldsymbol{W}(0) = \boldsymbol{I}$, $\boldsymbol{W}(k)$ will be symmetric, since $\boldsymbol{y}(k)\boldsymbol{y}^T$ (k) will be symmetric.

[0122] For a time domain based system addressing *instantaneous* mixtures with real signal values similar approaches have been indicated in A. Cichocki and S. Amari, "Adaptive Blind Signal and Image Processing", Chichester (West ussex, England): John Wiley & Sons, Ltd, 2002.

[0123] For the learning rule in the frequency domain, complex numbers should preferably be used instead of real values. To apply the local learning rule the matrix $\boldsymbol{W}(\omega)$ is preferably Hermitian, with $w_{ij}(\omega) = w_{ji}^*(\omega)$.

[0124] Since $\boldsymbol{R}_{xx}(\omega)$ is also Hermitian and positive definite, with positive real eigenvalues, **W**(ω) will also be positive definite, and the learning rule for complex numbers can be applied:

$$\boldsymbol{W}(k+1,\omega) = \boldsymbol{W}(k,\omega) + \eta(k)\left(\mathbf{\Lambda} - \boldsymbol{y}(k,\omega)\boldsymbol{y}^H(k,\omega)\right),$$

or in scalar form:

$$w_{ij}(k+1,\omega) = w_{ij}(k,\omega) + \eta(k)\left[\delta_{ij}\lambda_i - y_i(k,\omega)\, y_j^*(k,\omega)\right]$$

(Equation 1)

Note than when $\boldsymbol{W}(0, \omega)) = \boldsymbol{I}$, $\boldsymbol{W}(k+1, \omega)$ remains Hermitian, since $\boldsymbol{y}(k, \omega)\boldsymbol{y}^H(k, \omega)$ is Hermitian.

[0125] Here k is the frame index. Block/segment processing can be used where the time data is divided into blocks or frames of e.g. 256 points before each block or frame is being transformed to the frequency domain.

[0126] To understand some further aspects of this solution, we will take as an example a two microphone solution as illustrated in FIG. 3. We will start with a point noise source $s(\omega)$, with transfer functions $h_1(\omega)$ and $h_2(\omega)$, a 2x2 matrix W and outputs $y_1(\omega)$ and $y_2(\omega)$.

[0127] We initialize ***W***(0, ω) = ***I***, and for the moment do not update w_{11}(ω) and w_{22}(ω) Further, we assume $\mathbb{E}\{s(k,\omega)s^*(k,\omega)\} = 1$, for all *k* and ω. We frequently use w_{21}(*k*, ω) = $w_{12}^*(k,\omega)$.

[0128] If we look at the expected gradient $\mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\}$ we have:

$$\begin{aligned}&\mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\} = \mathbb{E}\{y_1(k,\omega)y_2^*(k,\omega)\}\\ &= \mathbb{E}\{\big(x_1(k,\omega) + w_{12}(k,\omega)x_2(k,\omega)\big)\big(x_2^*(k,\omega) + w_{21}^*(k,\omega)x_1^*(k,\omega)\big)\}\\ &= \big(h_1(\omega) + w_{12}(k,\omega)h_2(\omega)\big)\big(h_2^*(\omega) + w_{21}^*(k,\omega)h_1^*(\omega)\big)\mathbb{E}\{s(k,\omega)s^*(k,\omega)\}\\ &= \big(h_1(\omega) + w_{12}(k,\omega)h_2(\omega)\big)\big(h_2^*(\omega) + w_{21}^*(k,\omega)h_1^*(\omega)\big)\end{aligned}$$

(Equation 2)

For $\mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\} = 0$ we have two solutions:

$$w_{12}(k,\omega) = -\frac{h_1(\omega)}{h_2(\omega)}$$

and

$$w_{21}(k,\omega) = -\frac{h_2(\omega)}{h_1(\omega)}$$

[0129] The amplitudes of the two solutions are the inverse of each other, so we will always have a solution with amplitude smaller or equal to one.

[0130] It can be shown that with the adaptation approach above, we find the solution with the smallest amplitude, and that this solution is stable, except for |h_1(ω)| = |h_2(ω)|, when in (Eq. 2) both terms become zero. The same is true (with |h_1 (ω)| < |h_2(ω)|) when we choose

$$w_{22}(\omega) = \frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}$$

The righthand side of (Eq. 2) then can be written as:

$$w_{22}^*(\omega)h_2^*(\omega) + w_{21}^*(k,\omega)h_1^*(\omega)$$

and will become zero if

$$w_{12}(k,\omega) = w_{21}^*(k,\omega) = -\tfrac{h_1(\omega)}{h_2(\omega)}$$

This solution is typically not stable.

[0131] Stability issues may in some cases arise in case of a single point source the data correlation matrix $\tilde{\boldsymbol{R}}_{xx}$ is not positive definite, but semi-positive definite instead, with eigen values that are zero. Then it may not be appropriate to use the simple local learning rule, which requires ***W***(ω) to be positive definite. In the case with one point source, $\tilde{\boldsymbol{R}}_{xx}$ only contains one non-zero eigen value that belongs to the point source. If we have additional noise sources (both correlated and uncorrelated) then $\tilde{\boldsymbol{R}}_{xx}$ becomes positive definite and the simple learning rule can be applied.

[0132] In acoustic applications there will typically always be additional noise sources. Firstly, we have microphone sensor noise in each microphone. These noises are uncorrelated and have approximately equal variances. Secondly, we have remaining signal energy due to the point noise source. To understand this, we revert to time domain where we described the microphone signal as

$$x_i(k) = \sum_{m=0}^{\mathrm{m}=\infty} s(k-m)h_i(m),$$

and for the example with 2 microphones above we can write for $y_1(k)$:

$$y_1(k) = \sum_{m=0}^{\mathrm{m}=\infty} s(k-m)h_1(m) - \sum_{m=0}^{\mathrm{m=Nfir}} x_2(k-m)w_{12}(m),$$

where $w_{12}(m)$ is an FIR filter of length Nfir. After convergence we can write:

$$\sum_{m=0}^{\mathrm{m=Nfir}} x_2(k-m)w_{12}(m) = \sum_{m=0}^{\mathrm{m=Nfir}} s(k-m)h_1(m)$$

and rewrite $y_1(k)$ as:

$$y_1(k) = \sum_{m=0}^{\mathrm{m=Nfir}} s(k-m)h_1(m) + \sum_{m=Nfir}^{\mathrm{m}=\infty} s(k-m)h_1(m) - \sum_{m=0}^{\mathrm{m=Nfir}} s(k-m)h_1(m)$$

$$= \sum_{m=Nfir}^{\mathrm{m}=\infty} s(k-m)h_1(m),$$

where we split $x_1(k)$ into two parts: a part that can be dealt with by the decorrelator and a tail or diffuse part, that cannot be dealt with by the decorrelator (because of the finite FIR length).

[0133] All microphone signals i will have a tail or diffuse part, generated in the same way as the one for $x_1(k)$, with now $h_i(n)$ instead of $h_1(n)$.

[0134] If we look at the tail powers, then we can write for a white noise point source s:

$$\mathbb{E}\{x_{i,tail}^2(k)\} = \mathbb{E}\{s^2(k)\} \sum_{m=Nfir}^{\mathrm{m}=\infty} \mathbb{E}\{h_i^2(m)\}$$

[0135] The expected value $\mathbb{E}\{h_i^2(m)\}$, where the expectation is now over all possible source-microphone combinations in a room with a certain reverberation time T_{60} is given by:

$$\mathbb{E}\{\mathrm{h}_\mathrm{i}^2(\mathrm{m})\} = \mathrm{K}\,\mathrm{e}^{\frac{-\mathrm{m}}{\mathrm{F_s\,T_{60}}}}$$

with K a constant and F_s the sampling frequency.

[0136] The variance for a certain realization (impulse response between a certain source - microphone position) will be high, but due to the summation the variance of $\sum_{m=Nfir}^{\mathrm{m}=\infty} h_i^2(m)$ will be much lower. This means that as a first approximation we can assume that the contributions have equal powers.

[0137] When we have in addition to the point source uncorrelated (sensor) noise at the inputs, the off-diagonal filter in $\boldsymbol{W}(\omega)$ (in our example $w_{12}(\omega)$ and $w_{21}(\omega)$ cause correlation of the noise at the outputs. As a result, the solution will be different from the solution without uncorrelated noise at the inputs. It can be shown that the optimal solution, when applying the learning rule of (Eq. 1) with $i \neq j$ is given by:

$$\mathbb{E}\{w_{12,opt}\} = -\frac{\mathrm{h}_1(\omega)}{\mathrm{h}_2(\omega)}(1-\beta)$$

where β is real and $0 < \beta \leq 1$.

[0138] β = 0, corresponds to the solution where there is no uncorrelated noise at the inputs, β = 1 correponds to the solution where there is only uncorrelated noise at the inputs. This solution is stable, also for the case with $|h_1(\omega)| = |h_2(\omega)|$, The reason is that with the added uncorrelated noise $\tilde{\boldsymbol{R}}_{xx}$ will be positive definite and as a result, also $\boldsymbol{W}$ will be positive definite, as is required to apply our learning rule.

[0139] Now we can also change $w_{12}(\omega)$ and $w_{21}(\omega)$ without having stability problems.

[0140] It can be shown that when uncorrelated noise signals at the inputs have equal variances, the outputs will also have equal variances. We know that sensor noise and tail contributions can be considered as having, as a first approximation, equal variances. This means that choosing $w_{11}(k, \omega) = \mathrm{w}_{22}(\mathrm{k}, \omega) = 1$ is a valid option. It means that the update rule can be simplified to:

$$w_{ij}(k+1,\omega) = w_{ij}(k,\omega) - \eta(k)\left[y_i(k,\omega)\, y_j^*(k,\omega)\right], \quad \text{(Equation 3)}$$

with $i \neq j$.

[0141] In use cases where the described decorrelator is combined with e.g. an adaptive beamformer this has proven to be very robust. The zeroing of the off-diagonal components in $\boldsymbol{R}_{yy}(\omega)$ is often much more important than having equal elements in $\Lambda(\omega)$. Note that when the tail contributions, that scale linearly with the source strength $s(\omega)$ are dominant over the sensor noise, the solution is only determined by the acoustic impulse responses and does not depend on the signal characteristics of the source. Since the variations in the acoustic paths are much slower when compared with the variations in the source spectrum or source amplitudes, it means that the SDC can converge to a stable solution.

[0142] For the use case with a single point source and the tails dominating the sensor noise and we want to equalize the elements in $\Lambda(\omega)$ such that we can write $\boldsymbol{\Lambda}(\omega) = \alpha(\omega)\rho_{ss}^2(\omega)\mathbf{I}$, with $\alpha(\omega)$ not depending on $s(\omega)$, we can follow the following procedure:
First the average power is determined, weighted by the squared inverse of w_{ii}:

$$P_{avg}(\omega) = \sum_{i=0}^{i=Nmics-1} \frac{P_{y_i}(\omega)}{w_{ii}^2(\omega)}$$

with $P_{y_i}(\omega)$ the power of output i calculated as $y_i(\omega)y_i^*(\omega)$ (ω) Next an update of w_{ii} is calculated:

$$\mathrm{w_{ii}}(\mathrm{k}+1,\omega) = (1-\beta)\mathrm{w_{ii}}(\mathrm{k},\omega) + \beta\sqrt{\frac{P_{avg}(\omega)}{P_{y_i}(\omega)}},$$

β is a small smoothing parameter, such that the updates of the off-diagonal weights $\mathrm{w_{ij}}$ with $i \neq j$ can easily track the changes in $\mathrm{w_{ii}}$.

[0143] In many embodiments, instead of choosing a pre-defined diagonal matrix $\Lambda(\omega)$, the decorrelator may be initialized with the diagonal elements of $\mathbf{W}(\omega)$ equal to one and then the audio apparatus may proceed to not adapt these weights. In some embodiments, we may want to write $\Lambda(\omega) = \alpha(\omega)\rho_{ss}^2(\omega)$, with $\alpha(\omega)$ independent from $s(\omega)$ to equalize the outputs. The diagonal elements of $\boldsymbol{W}(\omega)$ will not generally be equal to one then.

[0144] The adaptive SDC is most efficiently implemented in the frequency domain with block processing. As a specific example, at 16 a kHz sampling frequency we may use a block/segment size of 256 points. This means that also the decorrelation will be over 256 points.

[0145] When filtering in frequency domain using FFT's resulting in discrete sampled frequencies, we have circular convolutions and circular correlations instead of linear ones.

[0146] Another aspect to consider is that the applied filters are, by definition, a-causal since $w_{ij}(\omega) = w_{ji}^{*}(\omega)$, which corresponds to a reversal in the time domain.

[0147] Regarding the circular versus linear convolution and correlations, techniques may be used for implementing linear convolutions and correlations in the frequency domain, using overlap-add or overlap-save techniques. An overlap-save technique for applying an FIR filter with N=B coefficients may include the following processing:

1) The input signal x(n) is partitioned in blocks/segments of B samples and concatenated with the previous block of B samples.
2) The resulting block of 2B samples is transformed to the frequency domain.
3) The impulse response of the FIR filter is extended with B zeroes, and is transformed by an FFT of size 2B to the frequency domain.
4) The outputs of the two FFTs are bin-wise multiplied, after which a 2B IFFT is applied.
5) It can be shown that the first B samples contain circular convolution artifacts, whereas the second B samples contain the result of only a linear convolution.
6) The second B samples are used as the output block.

[0148] The signal flow for such an example may be as illustrated in FIG. 4. The approach may provide a linear convolution using an overlap-save technique.

[0149] We use here an overlap of 50% but other overlaps may of course be used. Suppose we have a filter length N = 384 points. Then it is also possible to use a block size B=128 points and an (I)FFT of size N+B = 512 points. At the input a frame of B samples must be concatenated with 3 previous blocks (overlap 75%), giving a block of 512 points, the weighting/filter vector is extended with 128 zeroes and the last 128 samples of the 512 points output is selected as the output.

[0150] A linear correlation may be implemented similarly to the linear convolution as illustrated in FIG. 5 for an overlap of 50%.

[0151] Now suppose we have an impulse response that is a-causal and has non-zero values for |n| < N/2; In a circular representation this corresponds to a fundamental interval with $\tilde{h}(n) = h(n)$ and $\tilde{h}(N - n) = h(-n)$

[0152] Filtering with an a-causal filter in the frequency domain can then be performed as follows (with B=N):

1) The input signal x(n) is partitioned in blocks/ samples of B samples and concatenated with the previous block of B samples.
2) The resulting block of 2B samples is transformed to the frequency domain.
3) Create a vector w with $w(n) = h(n)$ for $0 \le n < N/2$, $w(n) = 0$ for $N/2 \le n \le 3N/2$ and $w(2N - n) = h(-n)$ for $0 < n < N/2$ and transform it to the frequency domain with a 2N point FFT.
4) The outputs of the two FFTs are bin-wise multiplied, after which a M=2N IFFT is applied.
5) It can be shown that now the first B/2 samples contain circular convolution artifacts, as well as the last B/2 samples in the block. The B points samples in the middle contain the result of only a linear convolution.
6) The B points in the middle are used as the output block.

[0153] A corresponding signal flow is shown in FIG. 6.

[0154] Using such principles an adaptive spatial decorrelator can be developed with linear filtering for the outputs and circular convolutions and correlations for updating the coefficients.

[0155] We can define an *nmics x nmics* matrix ***W***, where each entry $\boldsymbol{w}_{ij}$ consists of a (complex) frequency domain vector of M/2+1 points (we assume real input signals such that an input block of M samples gives M/2+1 unique frequency points). An element of this vector is given by: $w_{ij}(\omega)$. $\boldsymbol{W}(\omega)$ denotes, as before, an *nmics x nmics* matrix for a certain frequency (bin) ω (the weight matrix $\mathbf{W}(\omega)$).

[0156] The main diagonal of W may consist of elements $w_{ii}(\omega) = 1$ for all i and ω.

[0157] The approach may specifically follow the following steps:

1) The input (microphone) signals $x_i(n)$ with $0 \le i <$ Nmics, with Nmics the number of microphones, are partitioned into frames of B samples and concatenated with a previous frame of B samples giving a block of M = 2B samples
2) The Nmics blocks are transformed to frequency domain using an M-point FFT.
3) First the outputs in time domain are calculated using linear filtering. To achieve this the following steps are taken:

a) Each frequency domain vector $\mathbf{w}_{ij}$ is transformed to the time domain, and windowed (multiplied) by a window given by:

$$\mathrm{window(n)} = 1$$

for 0 ≤ n < M/4

$$\mathrm{window(n)} = 0$$

for M/4 ≤ n < 3M/4

$$\mathrm{window(n)} = 1$$

for 3*M*/4 ≤ n < M

b) The windowed time domain vectors are transformed into frequency domain, resulting in the matrix $\tilde{\mathbf{W}}$.

c) For each frequency ω the outputs are calculated according $\tilde{\mathbf{y}}(\omega) = \tilde{\mathbf{W}}(\omega)\mathbf{x}(\omega)$ with $\tilde{\mathbf{y}}(\omega)$ and $\mathbf{x}(\omega)$ column vectors of length Nmics.

d) The resulting Nmics frequency domain vectors $\tilde{y}_i(\omega)$ are converted back to the domain with an M-point IFFT.

e) The final output frames with B samples are extracted from the converted time domain signals by:

$$y_i(\mathrm{n}) = \tilde{y}_i(\mathrm{M/4+n})$$

for M/4 ≤ n < 3M/4

4) The updates are determined as follows:

a) For each ω the outputs are calculated according $\mathbf{y}(\omega) = \boldsymbol{W}(\omega)\boldsymbol{x}(\omega)$

[0158] The update formula is applied for each ω:

$$w_{ij}(k+1,\omega) = w_{ij}(k,\omega) - \eta(k,\omega)\left[y_i(k,\omega)\, y_j^*(k,\omega)\right] \text{ (i.j = 1,2,...Nmics, i} \neq j),$$

where k denotes the frame index

[0159] The steps 1-4 may be repeated for each frame/segment. An example of the operation of Steps 1-3 (the filtering/decorrelation steps) is illustrated in FIG. 7

[0160] In the specific approach, the filtering/weighted combination may thus include applying a time domain windowing to the frequency representation provided by the weights of a specific position in the weight matrix **W**(ω) for different frequency bins. Thus, for a given input signal and output signal, the weights for the different frequency bins form a frequency representation. The audio signal generator 105 may be arranged to generate modified weights that are used in the weighted combination/filtering by applying a time domain window.

[0161] The audio signal generator 105 may specifically take the frequency representation for a given weight (position, for a given input and output signal pair) and convert that to the time domain (e.g. using an FFT). The audio signal generator 105 may then apply a window to the time domain representation to generate a modified time domain representation. In many embodiments, the window may be such that a center portion of the segment of the time domain representation is set to (substantially) zero. The audio signal generator 105 may then convert the modified time domain representation to the frequency domain to generate modified weights. The filtering of the input signals to generate the output signals, i.e. the weighted combination, is based on the modified weights. However, the weight matrix **W**(ω) that is adapted and stored for future segments is unchanged. Thus, in some embodiments, the weight matrix **W**(ω) that is adapted is for the specific filtering modified based on the time domain windowing. Thus, for the filtering, the weight matrix **W** can be modified to $\tilde{\mathbf{W}}$.

[0162] With respect to the learning parameter $\eta(k, \omega)$, the adapter 107 may apply normalization per frequency bin ω, such that the convergence speed does not depend on the (frequency) amplitudes of the inputs. For each frequency ω the audio signal generator 105 can determine the summed power of the Nmics output signals $y_i(\omega)$ and smooth it with a first order recursion with a time constant of 50 msec. If we call this power $P_y(k, \omega)$ the audio signal generator 105 can determine the update constant $\eta(k, \omega) = \alpha/P_y(k, \omega)$ with α typically smaller than 0.1.

[0163] Since $w_{ij}(\omega) = w_{ji}^*(\omega)$ a considerable memory and complexity reduction in the update part can be obtained by storing and calculating only the values $w_{ij}(\omega)$ for j < i.

[0164] Apart from the considerable complexity reduction the implementation with, in the update part, circular convolutions and correlations instead of linear ones is found to be more robust, for the cases where the input is semi-positive definite (mostly in simulations).

In the following a detailed exemplary analysis is provided of the scenario of FIG. 3 with two microphone inputs and using the update rule in Equation 1.

[0165] We initialize ***W***(0, ω) = **I,** and for the moment do not update $w_{11}(\omega)$ and $w_{22}(\omega)$ Further, we assume $\mathbb{E}\{s(k,\omega)s^*(k,\omega)\} = 1$, for all *k* and ω. We frequently use $w_{21}(k, \omega)$ = $w_{12}^*(k,\omega)$.

[0166] In case of a single point noise source, without additional noises we found that for $\mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\} = 0$ we have two solutions:

$$w_{12}(k,\omega) = -\frac{h_1(\omega)}{h_2(\omega)}$$

and

$$w_{21}(k,\omega) = -\frac{h_2(\omega)}{h_1(\omega)}$$

[0167] The amplitudes of the two solutions are the inverse of each other, so we will always have a solution with amplitude smaller or equal to one.

[0168] From now on we assume $|h_2(\omega)| > |h_1(\omega)|$, without loss of generality, since otherwise we will start with w_{21}(k, ω) instead of $w_{12}(k, \omega)$. The case $|h_2(\omega)| = |h_1(\omega)|$ will be discussed later.

[0169] If we look at the first iteration with $w_{1,2}(0, \omega) = 0$ we have:

$$\mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\} = \mathbb{E}\{y_1(0,\omega)y_2^*(0,\omega)\} = h_1(\omega)h_2^*(\omega)$$

$$= \frac{h_1(\omega)}{h_2(\omega)}|h_2(\omega)|^2$$

[0170] For the expected weight update we have:

$$\mathbb{E}\{\Delta w_{1,2}(0,\omega)\} = -\eta\mathbb{E}\{\nabla\, \mathrm{w}_{12}(0,\ \omega)\} = -\eta\frac{h_1(\omega)}{h_2(\omega)}|h_2(\omega)|^2$$

[0171] The update will be in the direction of the solution $-h_1(\omega)/h_2(\omega)$, that is it has the same phase and with $\beta = \eta|h_2(\omega)|^2$ a small positive number we can write for $y_1(1, \omega)$:

$$y_1(1,\omega) = \big(h1(\omega) - \beta h_1(\omega)\big)s(1,\omega)$$

$$= h_1(\omega)(1 - \beta)s(1,\omega)$$

$$h_1(1,\omega)s(1,\omega)$$

and for $y_2(1, \omega)$:

$$y_2(1,\omega) = \left(h_2(\omega) - \beta\frac{h_1^*(\omega)}{h_2^*(\omega)}h_1(\omega)\right)s(1,\omega)$$

$$= h_2(\omega)\left(1\ - \beta\frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right)s(1,\omega)$$

$$= h_2(1,\omega)s(1,\omega)$$

[0172] Compared with $y_1(0, \omega)$ and $y_2(0, \omega)$, we only see that the amplitude of $h_1(1, \omega)$ and $h_2(1, \omega)$ changes when compared with $h_1(\omega)$ and $h_2(\omega)$. Note that the amplitude $h_1(1, \omega)$ decreases faster than the amplitude of $h_2(1, \omega)$.

[0173] In next iterations of the algorithm the amplitudes of $h_1(k, \omega)$ and $h_2(k, \omega)$ decrease further until for certain k, the expected gradient will be zero and have:

$$\mathbb{E}\{w_{12}(\mathrm{k}, \omega)\} = -\frac{h_1(\omega)}{h_2(\omega)}$$

This solution is also a stable solution. Suppose we perturbate the solution with dw (complex) and have (omitting the k index):

$$w_{12}(\omega) = -\frac{h_1(\omega)}{h_2(\omega)} + dw$$

and

$$w_{21}(\omega) = -\frac{h_1^*(\omega)}{h_2^*(\omega)} + dw^*$$

[0174] Then we have for $y_1(\omega)$ and $y_2(\omega)$:

$$y_1(\omega) = dw\, h_2(\omega) s(\omega)$$

and

$$y_2(\omega) = h_2(\omega)\left(1 - \frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right) s(\omega) + dw^* h_1(\omega) s(\omega)$$

[0175] The expected value for the gradient is (neglecting higher order term $(dw)^2$)

$$\mathbb{E}\{y_1(\omega) y_2^*(\omega)\} = dw(|h_2(\omega)|^2 - h_1(\omega)|^2)$$

[0176] In the update we have

$$\mathbb{E}\{w_{12}(\mathrm{k}+1, \omega)\} = -\frac{h_1(\omega)}{h_2(\omega)} + dw - \eta\, dw(|h_2(\omega)|^2 - |h_1(\omega)|^2)$$

[0177] The update is in the opposite direction of dw for all dw ($\eta(|h_2(\omega)|^2 - |h_1(\omega)|^2)$ is positive), and thus the solution is stable.

[0178] Looking at the powers of $y_1(\omega)$ and $y_2(\omega)$ after convergence we have:

$$\mathbb{E}\{y_1(\omega) y_1^*(\omega)\} = 0$$

and

$$\mathbb{E}\{y_2(\omega) y_2^*(\omega)\} = |h_2(\omega)|^2 \left(1 - \frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right)^2 \mathbb{E}\{s(\omega) s^*(\omega)\}$$

[0179] Depending on the ratio between $|h_1(\omega)|$ and $|h_2(\omega)|$ the power of $y_2(\omega)$ also decreases, but much less than the power of $y_1(\omega)$.

[0180] This is contradictory to what we want to achieve, namely that the output powers are equal. In this scenario this is only possible if also the output power of $y_2(\omega)$ would be zero. This could be done by setting $w_{22}(\omega) = h_1(\omega)/h_2(\omega)$. This change does not influence the update of $w_{12}(\omega)$, but once $w_{12}(\omega)$ has found its optimal value, also the expected power of $y_2(\omega)$ will be zero. The solution is not stable however, since if we perturbate the solution as before with

$$w_{12}(\omega) = -\frac{h_1(\omega)}{h_2(\omega)} + dw$$

we now get

$$y_1(\omega) = dw\, h_2(\omega)s(\omega)$$

and

$$y_2(\omega) = dw^* h_1(\omega)s(\omega)$$

[0181] For the expected value of the gradient:

$$\mathbb{E}\{y_1(\omega)y_2^*(\omega)\} = h_2(\omega)h_1^*(\omega)(dw)^2\mathbb{E}\{s(\omega)s^*(\omega)\}$$

[0182] Now we cannot neglect the higher order term $(dw)^2$ and have for the weight:

$$\mathbb{E}\{w_{12}(\mathrm{k}+1,\omega)\} = -\frac{h_1(\omega)}{h_2(\omega)} + dw - \eta\, h_2(\omega)h_1^*(\omega)(dw)^2$$

$$= -\frac{h_1(\omega)}{h_2(\omega)} + dw(1 - \eta\, h_2(\omega)h_1^*(\omega)dw)$$

[0183] It cannot be guaranteed that $|1 - \eta\, h_2(\omega)h_1^*(\omega)dw| < 1$. For example with dw = $-\alpha h_1(\omega)/h_2(\omega)$, where α is a small positive number $|1 - \eta\, h_2(\omega)h_1^*(\omega)dw| > 1$.

[0184] The solution with two expected powers equal to zero is not stable. The same is true for the case $|h_2(\omega)| = |h_1(\omega)|$ (and $w_{22}(\omega) = 1$) again. Also then the expected values of the power of outputs after convergence will be zero, after perturbation the higher order term $(dw)^2$ cannot be neglected and we have:

$$\mathbb{E}\{w_{12}(\mathrm{k}+1,\omega)\} = -1 + dw(1 - \eta|h_1(\omega)|^2\, dw),$$

which will not be stable for $dw < 0$.

[0185] Such stability issues may be due to that for a single point source the data correlation matrix $\tilde{\boldsymbol{R}}_{xx}$ is not positive definite, but semi-positive definite instead, with eigen values that are zero. Then we are not allowed to use the simple local learning rule, which requires $\boldsymbol{W}(\omega)$ to be positive definite. In the case with one point source $\tilde{\boldsymbol{R}}_{xx}$ only contains one non-zero eigen value that belongs to the point source. If we have additional noise sources (both correlated and uncorrelated) then $\tilde{\boldsymbol{R}}_{xx}$ becomes positive definite and the simple learning rule can be applied.

[0186] Assume that besides the single point noise source $s(\omega)$ we have additionally uncorrelated noise $s_1(\omega)$ in the first microphone and $s_2(\omega)$ in the second microphone with equal variances, i.e. $\mathbb{E}\{s_1(\omega)s_1^*(\omega)\} = \mathbb{E}\{s_2(\omega)s_2^*(\omega)\} = \rho_s^2(\omega)$. This is typical for sensor noise.

[0187] If we look at the noise contributions of $s_1(\omega)$ and $s_2(\omega)$ in the outputs of $y_1(\omega)$ and $y_2(\omega)$ then we have:

$$y_1(\omega) = s_1(\omega) + w_{1,2}(\omega)s_2(\omega)$$

$$y_2(\omega) = s_2(\omega) + w_{2,1}(\omega)s_1(\omega)$$

[0188] Because $w_{2,1}(\omega) = w_{1,2}^*(\omega)$, the powers of $y_1(\omega)$ and $y_2(\omega)$ will be equal and for the expected gradient (only due to the noise contributions) we get:

$$\mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\} \ = \ \mathbb{E}\{y_1(\omega)y_2^*(\omega)\} = \mathrm{w}_{12}(\omega)\big(\rho_s^2(\omega) + \rho_s^2(\omega)\big)$$

[0189] In case the point noise source $s(\omega)$ would not be present we would have for the update:

$$w_{12}(\mathrm{k}+1, \omega) \ = w_{12}(\mathrm{k}, \omega)\left(1 - \eta\big(\rho_s^2(\omega) + \rho_s^2(\omega)\big)\right)$$

[0190] Thus $|\mathrm{w}_{12}(\mathrm{k} + 1,\ \omega)|$ and $|w_{21}(\mathrm{k} + 1,\ \omega)|$ decrease towards zero, what is desired for uncorrelated noise.

[0191] When the point noise source $s(\omega)$ is present the optimal solution changes, since the gradient of the point noise must compensate the gradient that is due to the uncorrelated noise in the inputs.

[0192] Since the gradient due to the noise is in the direction of $w_{12}(\mathrm{k},\ \omega)$ the optimal solution will be in the direction of $-h_1(\omega)/h_2(\omega)$, the optimal solution without noise (we still assume $|h_2(\omega)| > |h_1(\omega)|$)

[0193] Suppose the deviation of the optimal solution without noise is $\beta h_1(\omega)/h_2(\omega)$. The gradient for the noise then is:

$$\mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\} \ = \ \mathbb{E}\{y_1(\omega)y_2^*(\omega)\} = -\tfrac{h_1(\omega)}{h_2(\omega)}(1-\beta)\big(\rho_s^2(\omega) + \rho_s^2(\omega)\big)$$

[0194] For the gradient that is due to the point source we derive:

$$y_1(\omega) = \beta h_1(\omega)s(\omega)$$

$$y_2(\omega) \ = \ h_2(\omega)\left(1 - (1-\beta)\frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right)s(\omega)$$

$$\mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\} = h_1(\omega)h_2^*(\omega)\beta\left(1 - (1-\beta)\tfrac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right)\mathbb{E}\{s(\omega)s^*(\omega)\}$$

$$= \frac{h_1(\omega)}{h_2(\omega)}\beta|h_2(\omega)|^2\left(1 - (1-\beta)\frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right)$$

[0195] For $0 < \beta \le 1$,

$$\beta|h_2(\omega)|^2\left(1 - (1-\beta)\frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right)$$

is positive and increases monotonically with increasing β, whereas

$$(1-\beta)\big(\rho_s^2(\omega) + \rho_s^2(\omega)\big)$$

is also positive, but decreases monotonically and is zero for $\beta = 1$. This means there is a solution where the expected gradient is zero, and the solution is given by

$$\mathbb{E}\{w_{12,opt}\} = -\frac{h_1(\omega)}{h_2(\omega)}(1-\beta)$$

with $0 < \beta \le 1$.

[0196] This solution is also stable. Suppose we perturbate the optimal solution with dw (complex values)

$$w_{12}(\omega) = -\frac{h_1(\omega)}{h_2(\omega)}(1-\beta) + dw$$

[0197] We then get for $y_1(\omega)$:

$$y_1(\omega) = (\beta \mathrm{h}_1(\omega) + \mathrm{h}_2(\omega) dw) s(\omega) \ + \ s_1(\omega) + w_{12}(\omega) s_2(\omega)$$

and for $y_2(\omega)$:

$$y_2(\omega) = \left(h_2(\omega)\left(1 - (1-\beta)\frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right) + dw^*(\omega) h_1(\omega) \right) s(\omega) + w_{12}^* s_1(\omega) + s_2(\omega)$$

[0198] For the gradient we only must consider the extra components due to *dw* and can neglect the higher order term $(dw)^2$

$$\begin{aligned} \mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\} &= dw|h_2(\omega)|^2\left(1 - (1-\beta)\frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right)\mathbb{E}\{s(\omega)s^*(\omega)\} \ + \ dw\left(\rho_{s}^{2}(\omega) + \rho_{s}^{2}(\omega)\right) \\ &= dw(\ |h_2(\omega)|^2\left(1 - (1-\beta)\frac{|h_1(\omega)|^2}{|h_2(\omega)|^2}\right) + \left(\rho_{s}^{2}(\omega) + \rho_{s}^{2}(\omega)\right) \end{aligned}$$

[0199] With $0 < \beta \leq 1$the righthand side is always positive, also for $|h_1(\omega)| = |h_2(\omega)|$, such that the update

$$\Delta w_{12}(k, \omega\} = -\eta \mathbb{E}\{\nabla\, \mathrm{w}_{12}(\mathrm{k},\ \omega)\},$$

is always opposite to dw, which guarantees stability.

[0200] The audio apparatus(s) may specifically be implemented in one or more suitably programmed processors. The different functional blocks may be implemented in separate processors and/or may, e.g., be implemented in the same processor. An example of a suitable processor is provided in the following.

[0201] FIG. 8 is a block diagram illustrating an example processor 800 according to embodiments of the disclosure. Processor 800 may be used to implement one or more processors implementing an apparatus as previously described or elements thereof. Processor 800 may be any suitable processor type including, but not limited to, a microprocessor, a microcontroller, a Digital Signal Processor (DSP), a Field ProGrammable Array (FPGA) where the FPGA has been programmed to form a processor, a Graphical Processing Unit (GPU), an Application Specific Integrated Circuit (ASIC) where the ASIC has been designed to form a processor, or a combination thereof.

[0202] The processor 800 may include one or more cores 802. The core 802 may include one or more Arithmetic Logic Units (ALU) 804. In some embodiments, the core 802 may include a Floating Point Logic Unit (FPLU) 806 and/or a Digital Signal Processing Unit (DSPU) 808 in addition to or instead of the ALU 804.

[0203] The processor 800 may include one or more registers 812 communicatively coupled to the core 802. The registers 812 may be implemented using dedicated logic gate circuits (e.g., flip-flops) and/or any memory technology. In some embodiments the registers 812 may be implemented using static memory. The register may provide data, instructions and addresses to the core 802.

[0204] In some embodiments, processor 800 may include one or more levels of cache memory 810 communicatively coupled to the core 802. The cache memory 810 may provide computer-readable instructions to the core 802 for execution. The cache memory 810 may provide data for processing by the core 802. In some embodiments, the computer-readable instructions may have been provided to the cache memory 810 by a local memory, for example, local memory attached to the external bus 816. The cache memory 810 may be implemented with any suitable cache memory type, for example, Metal-Oxide Semiconductor (MOS) memory such as Static Random Access Memory (SRAM), Dynamic Random Access Memory (DRAM), and/or any other suitable memory technology.

[0205] The processor 800 may include a controller 814, which may control input to the processor 800 from other processors and/or components included in a system and/or outputs from the processor 800 to other processors and/or components included in the system. Controller 814 may control the data paths in the ALU 804, FPLU 806 and/or DSPU

808. Controller 814 may be implemented as one or more state machines, data paths and/or dedicated control logic. The gates of controller 814 may be implemented as standalone gates, FPGA, ASIC or any other suitable technology.

[0206] The registers 812 and the cache 810 may communicate with controller 814 and core 802 via internal connections 820A, 820B, 820C and 820D. Internal connections may be implemented as a bus, multiplexer, crossbar switch, and/or any other suitable connection technology.

[0207] Inputs and outputs for the processor 800 may be provided via a bus 816, which may include one or more conductive lines. The bus 816 may be communicatively coupled to one or more components of processor 800, for example the controller 814, cache 810, and/or register 812. The bus 816 may be coupled to one or more components of the system.

[0208] The bus 816 may be coupled to one or more external memories. The external memories may include Read Only Memory (ROM) 832. ROM 832 may be a masked ROM, Electronically Programmable Read Only Memory (EPROM) or any other suitable technology. The external memory may include Random Access Memory (RAM) 833. RAM 833 may be a static RAM, battery backed up static RAM, Dynamic RAM (DRAM) or any other suitable technology. The external memory may include Electrically Erasable Programmable Read Only Memory (EEPROM) 835. The external memory may include Flash memory 834. The External memory may include a magnetic storage device such as disc 836. In some embodiments, the external memories may be included in a system.

[0209] It will be appreciated that the approach may further advantageously in many embodiments further include an adaptive canceller which is arranged to cancel a signal component of the beamformed audio output signal which is correlated with the at least one noise reference signal. For example, similarly to the example of FIG. 1, an adaptive filter may have the noise reference signal as an input and with the output being subtracted from the beamformed audio output signal. The adaptive filter may, e.g. be arranged to minimize the level of the resulting signal during time intervals where no speech is present.

[0210] It will be appreciated that the above description for clarity has described embodiments of the invention with reference to different functional circuits, units and processors. However, it will be apparent that any suitable distribution of functionality between different functional circuits, units or processors may be used without detracting from the invention. For example, functionality illustrated to be performed by separate processors or controllers may be performed by the same processor or controllers. Hence, references to specific functional units or circuits are only to be seen as references to suitable means for providing the described functionality rather than indicative of a strict logical or physical structure or organization.

[0211] The invention can be implemented in any suitable form including hardware, software, firmware or any combination of these. The invention may optionally be implemented at least partly as computer software running on one or more data processors and/or digital signal processors. The elements and components of an embodiment of the invention may be physically, functionally and logically implemented in any suitable way. Indeed, the functionality may be implemented in a single unit, in a plurality of units or as part of other functional units. As such, the invention may be implemented in a single unit or may be physically and functionally distributed between different units, circuits and processors.

[0212] Although the present invention has been described in connection with some embodiments, it is not intended to be limited to the specific form set forth herein. Rather, the scope of the present invention is limited only by the accompanying claims. Additionally, although a feature may appear to be described in connection with particular embodiments, one skilled in the art would recognize that various features of the described embodiments may be combined in accordance with the invention. In the claims, the term comprising does not exclude the presence of other elements or steps.

[0213] According to an aspect of the invention, the method of the claims and/or description is a method excluding a method for performing mental acts as such.

[0214] Furthermore, although individually listed, a plurality of means, elements, circuits or method steps may be implemented by, e.g., a single circuit, unit or processor. Additionally, although individual features may be included in different claims, these may possibly be advantageously combined, and the inclusion in different claims does not imply that a combination of features is not feasible and/or advantageous. Also, the inclusion of a feature in one category of claims does not imply a limitation to this category but rather indicates that the feature is equally applicable to other claim categories as appropriate. Furthermore, the order of features in the claims do not imply any specific order in which the features must be worked and in particular the order of individual steps in a method claim does not imply that the steps must be performed in this order. Rather, the steps may be performed in any suitable order. In addition, singular references do not exclude a plurality. Thus, references to "a", "an", "first", "second" etc. do not preclude a plurality. Reference signs in the claims are provided merely as a clarifying example shall not be construed as limiting the scope of the claims in any way.

Claims

1. An audio apparatus for generating a set of output audio signals, the audio apparatus comprising:

a receiver (101) arranged to receive a set of input audio signals;

a segmenter (103) arranged to segment the set of input audio signals into time segments;
an output signal generator (105) arranged to generate the set of output audio signals, each output audio signal of the set of output audio signals being linked with an input audio signal of the set of input audio signals, the output signal generator (105) being arranged to for each time segment perform the steps of:

generating (201) a frequency bin representation of the set of input audio signals, each frequency bin of the frequency bin representation of the set of input audio signals comprising a frequency bin value for each of the input audio signals of the set of input audio signals;
generating (203) a frequency bin representation of the set of output audio signals, each frequency bin of the frequency bin representation of a set of output audio signals comprising a frequency bin value for each of the set of output audio signals, the frequency bin value for a given output audio signal of the set of output audio signals for a given frequency bin being generated as a weighted combination of frequency bin values of the set of input audio signals for the given frequency bin;
generating (205) a time domain representation for each output audio signal from the frequency bin representation of the set of output audio signals;

an adapter (107) arranged to update weights of the weighted combination;

wherein the adapter (107) is arranged to update a first weight for a contribution to a first frequency bin value of a first frequency bin for a first output audio signal linked with a first input audio signal from a second frequency bin value of the first frequency bin for a second input audio signal linked to a second output audio signal in response to a correlation measure between a first previous frequency bin value of the first output audio signal for the first frequency bin and a second previous frequency bin value of the second output audio signal for the first frequency bin.

2. The audio apparatus of claim 1 wherein the adapter (107) is arranged to update the first weight in response to a product of a first value and a second value, the first value being one of the first previous frequency bin value and the second previous frequency bin value and the second value being a complex conjugate of the other of the first previous frequency bin value and the second previous frequency bin value.

3. The apparatus of claim 1 or 2 wherein the adapter (107) is arranged to update a second weight being for a contribution to the first frequency bin value from a third frequency bin value being a frequency bin value of the first frequency bin for the first input audio signal in response to a magnitude of the first previous frequency bin value.

4. The apparatus of claim 1 or 2 wherein the adapter (107) is arranged to set a second weight being for a contribution to the first frequency bin value from a third frequency bin value being a frequency bin value of the first frequency bin for the first input audio signal to a predetermined value.

5. The apparatus of any previous claim wherein the adapter (107) is arranged to constrain a weight for a contribution to the first frequency bin value from a third frequency bin value being a frequency bin value of the first frequency bin for the first input audio signal to be a real value.

6. The apparatus of any previous claim wherein the adapter (107) is arranged to set a third weight being a weight for a contribution to a fourth frequency bin value of the first frequency bin for the second output audio signal from the first input audio signal to be a complex conjugate of the first weight.

7. The apparatus of any previous claim wherein weights of the weighted combination for other input audio signals than the first input audio signal are complex valued weights.

8. The apparatus of any previous claim wherein the adapter (107) is arranged to determine output bin values for the given frequency bin ω from:

$$\mathbf{y}(\omega) = \mathbf{W}(\omega)\mathbf{x}(\omega)$$

where $\mathbf{y}(\omega)$ is a vector comprising the frequency bin values for the output audio signals for the given frequency bin ω; $\mathbf{x}(\omega)$ is a vector comprising the frequency bin values for the input audio signals for the given frequency bin ω; and $\mathbf{W}(\omega)$ is a matrix having rows comprising weights of a weighted combination for the output audio signals.

9. The apparatus of any previous claim wherein the adapter (107) is arranged to adapt weights w_{ij} of the matrix **W**(ω) according to:

$$w_{ij}(k+1,\omega) = w_{ij}(k,\omega) \; - \; \eta(k,\omega)\left[y_i(k,\omega)\, y_j^*(k,\omega)\right]$$

where i is a row index of the matrix **W**(ω) , j is a column index of the matrix **W**(ω) , k is a time segment index, ω represents the frequency bin, and *η(k,* ω) is a scaling parameter for adapting an adaptation speed.

10. The apparatus of any previous claim wherein the adapter (107) is arranged to compensate the correlation value for a signal level of the first frequency bin.

11. The apparatus of any previous claim where the adapter (107) is arranged to initialize the weights for the weighted combination to comprise at least one zero value weight and one non-zero value weight.

12. The apparatus of any previous claim wherein the weighted combination comprises applying a time domain windowing to a frequency representation of weights formed by weights for the first input audio signal and the second input audio signal for different frequency bins.

13. A method of generating a set of output audio signals:

receiving a set of input audio signals;
segmenting the set of input audio signals into time segments;
generating the set of output audio signals, each output audio signal of the set of output audio signals being linked with one input audio signal of the set of input audio signals, wherein generating the set of output audio signals comprises for each time segment performing the steps of:

generating (201) a frequency bin representation of the set of input audio signals, each frequency bin of the frequency bin representation of the set of input audio signals comprising a frequency bin value for each of the input audio signals of the set of input audio signals;
generating (203) a frequency bin representation of the set of output audio signals, each frequency bin of the frequency bin representation of a set of output audio signals comprising a frequency bin value for each of the output audio signals, the frequency bin value for a given output audio signal of the set of output audio signals for a given frequency bin being generated as a weighted combination of frequency bin values of the set of input audio signals for the given frequency bin;
generating (205) a time domain representation for each output audio signal from the frequency bin representation of the set of output audio signals;

and the method further comprises updating weights of the weighted combination including updating a first weight for a contribution to a first frequency bin value of a first frequency bin for a first output audio signal linked with a first input audio signal from a second frequency bin value of the first frequency bin for a second input audio signal linked to a second output audio signal in response to a correlation measure between a first previous frequency bin value of the first output audio signal for the first frequency bin and a second previous frequency bin value of the second output audio signal for the first frequency bin.

14. A computer program product comprising computer program code means adapted to perform all the steps of claim 13 when said program is run on a computer.

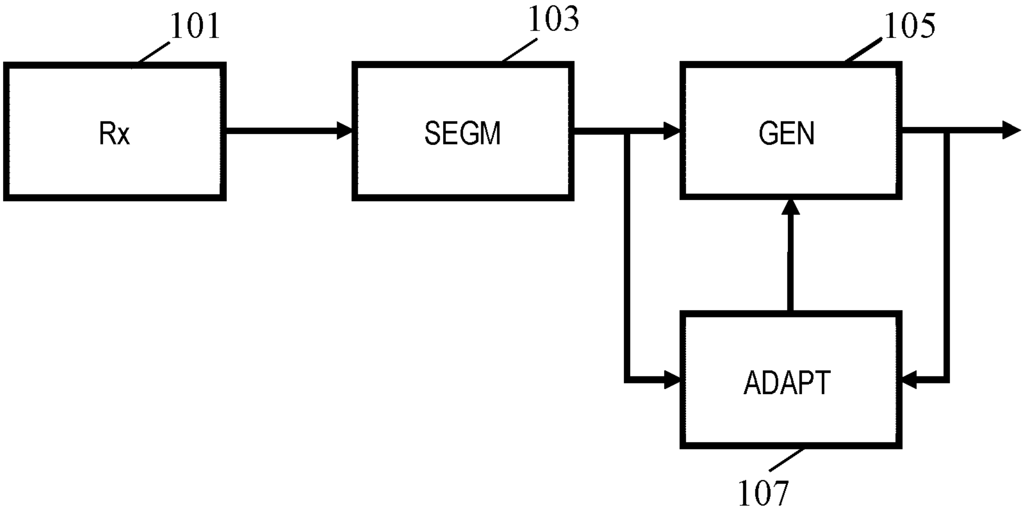

FIG. 1

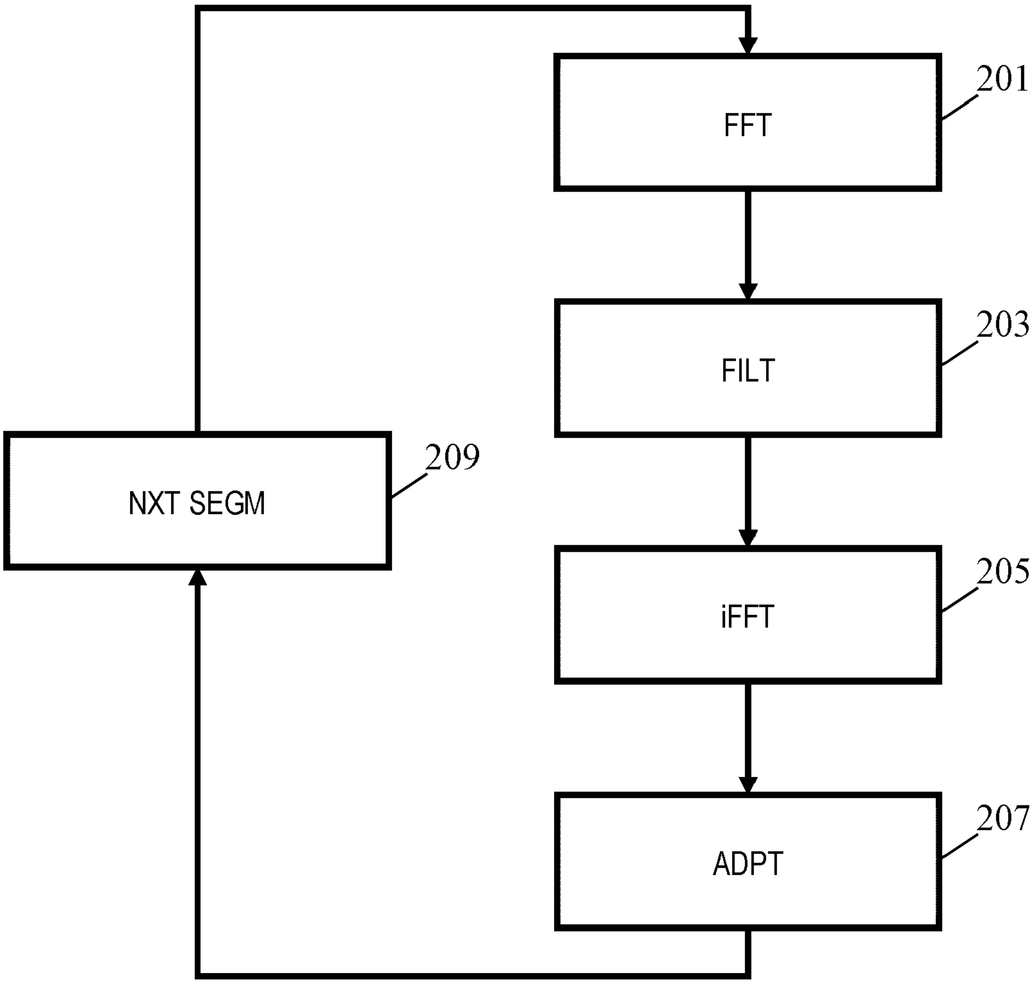

FIG. 2

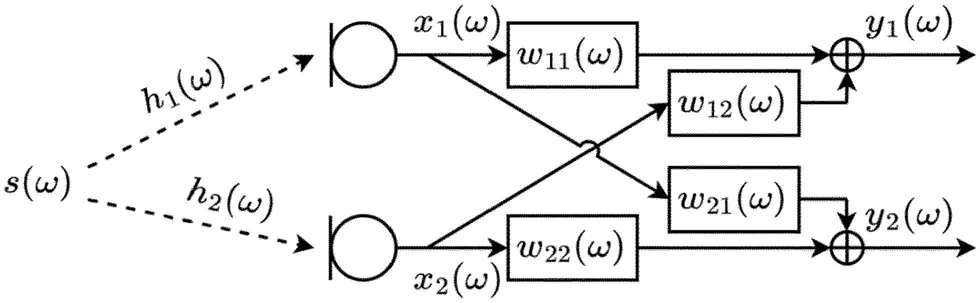
$s(\omega)$
$h_1(\omega)$
$h_2(\omega)$
$x_1(\omega)$
$x_2(\omega)$
$w_{11}(\omega)$
$w_{12}(\omega)$
$w_{21}(\omega)$
$w_{22}(\omega)$
$y_1(\omega)$
$y_2(\omega)$

FIG. 3

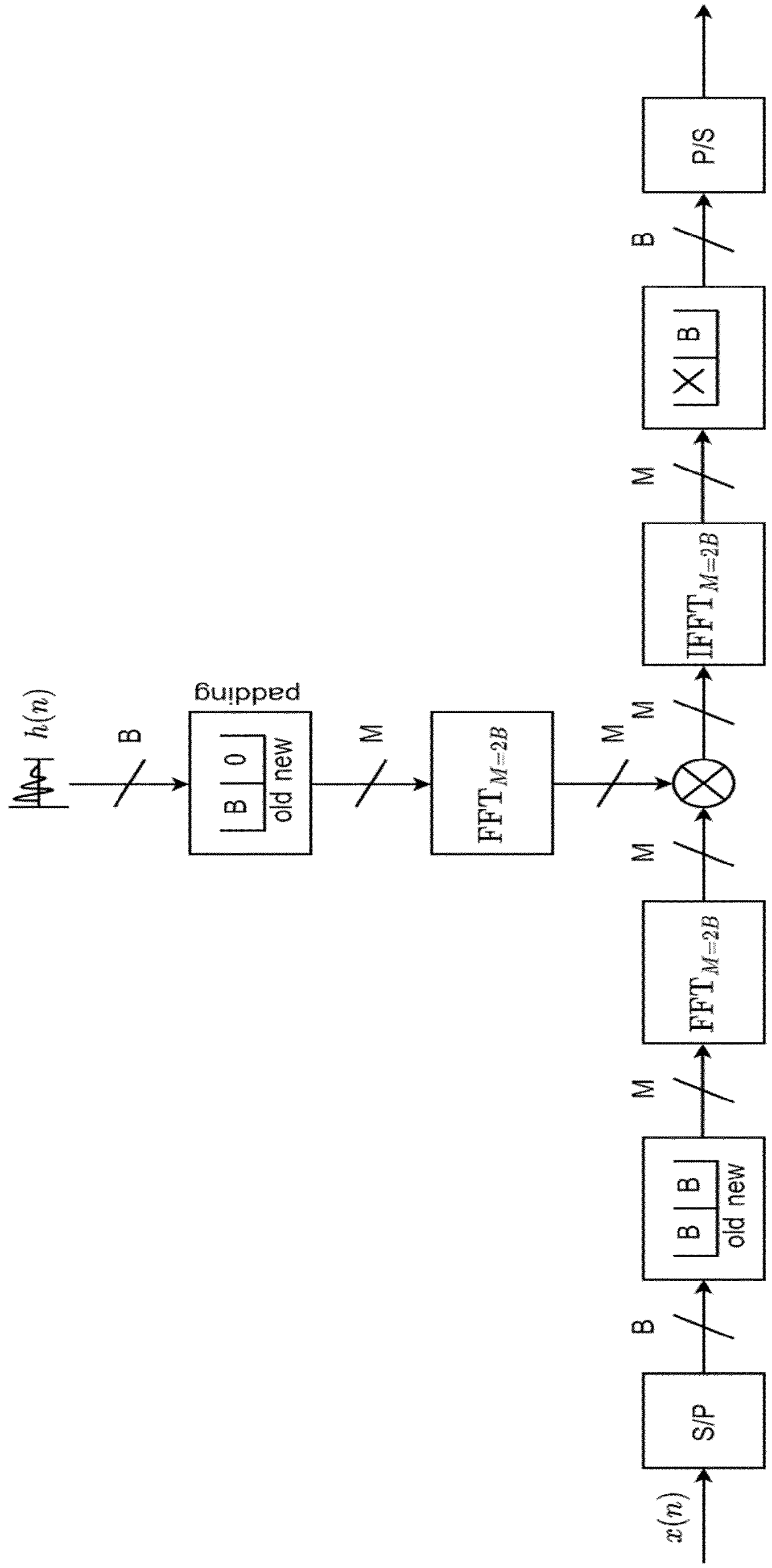
$x(n)$
S/P
B
B
B
old new
overlap
M
$\mathrm{FFT}_{M=2B}$
M
$h(n)$
B
B
0
old new
padding
M
$\mathrm{FFT}_{M=2B}$
M
M
$\mathrm{IFFT}_{M=2B}$
M
B
prune
B
P/S

FIG. 4

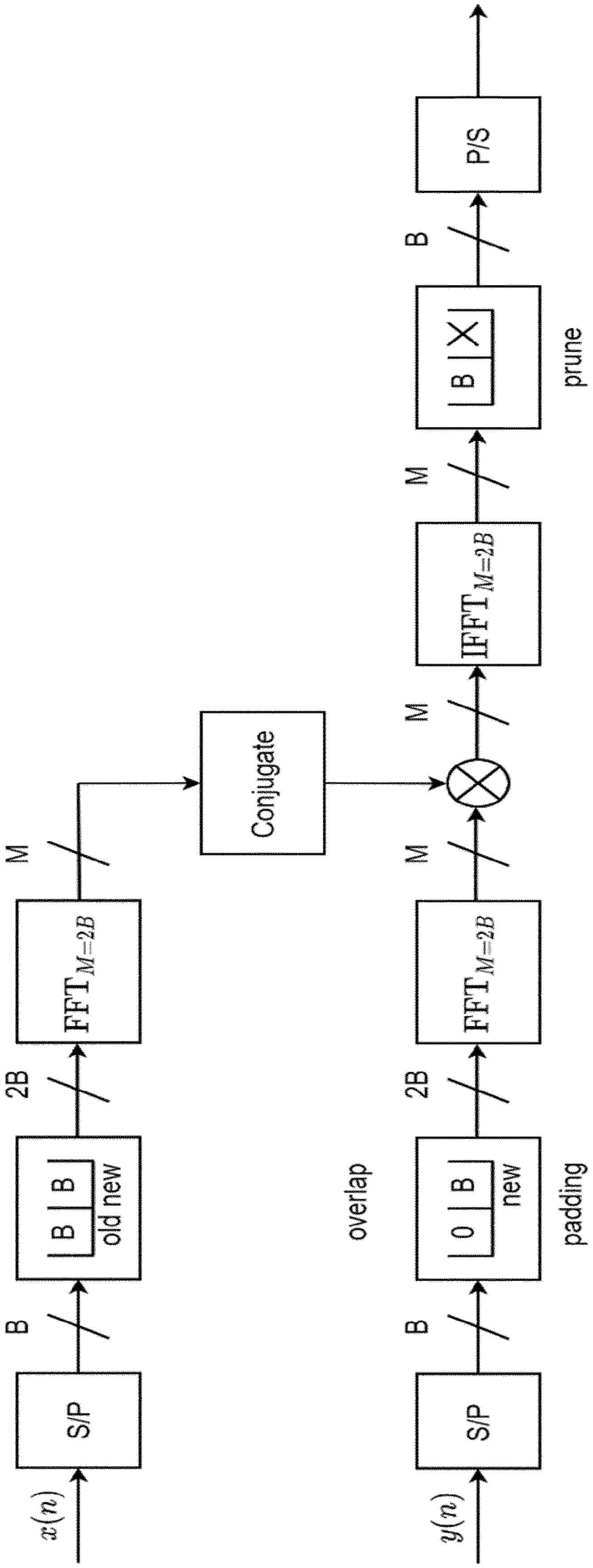
$x(n)$
S/P
B
B
B
old new
2B
$\mathrm{FFT}_{M=2B}$
M
Conjugate
overlap
$y(n)$
S/P
B
0
B
new
padding
2B
$\mathrm{FFT}_{M=2B}$
M
M
$\mathrm{IFFT}_{M=2B}$
M
B
prune
B
P/S

FIG. 5

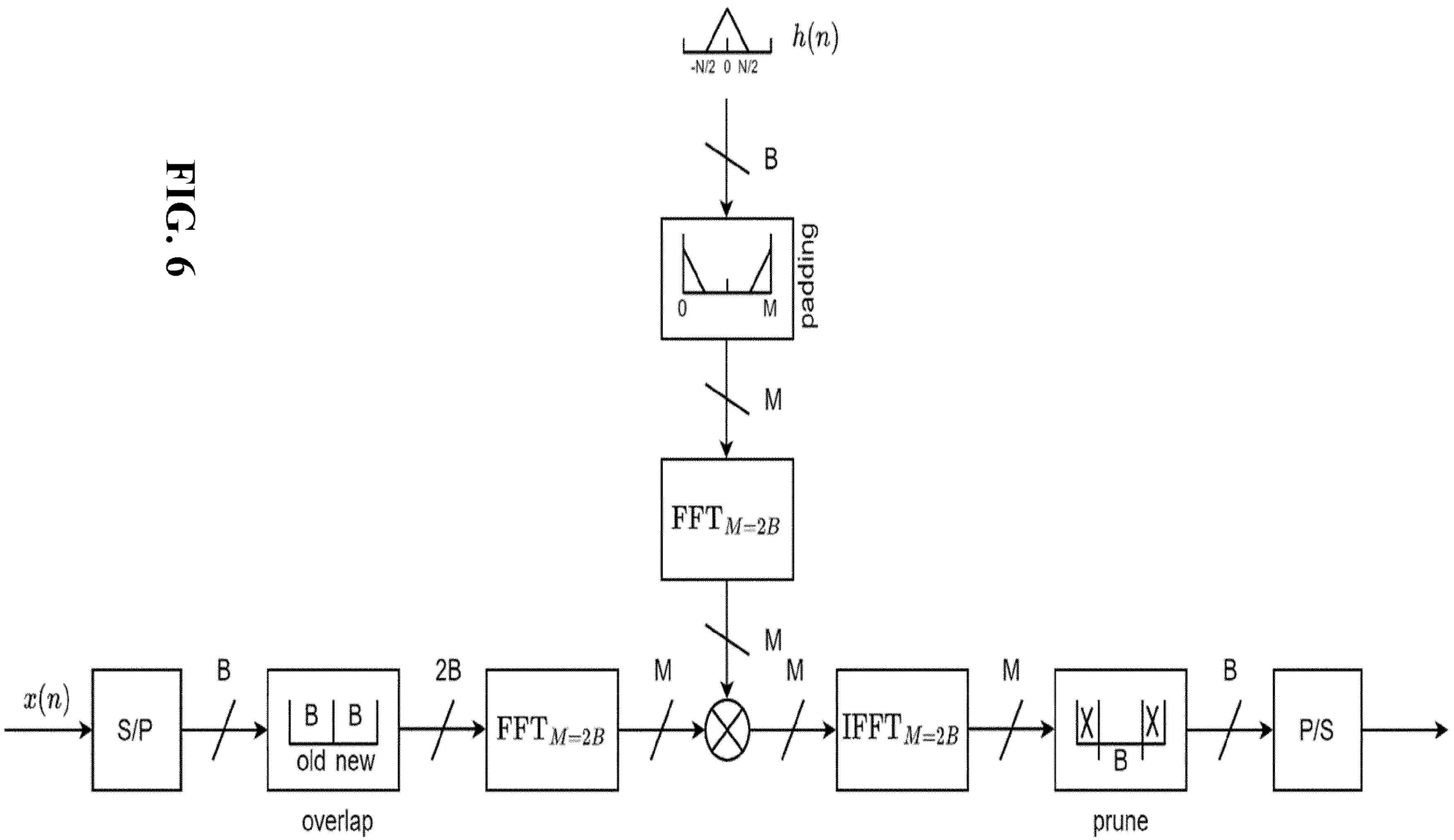
h(n)
-N/2 0 N/2
B
padding
0
M
M
FFT_{M=2B}
M
x(n)
S/P
B
B B
old new
overlap
2B
FFT_{M=2B}
M
M
IFFT_{M=2B}
M
X X
B
prune
B
P/S

FIG. 6

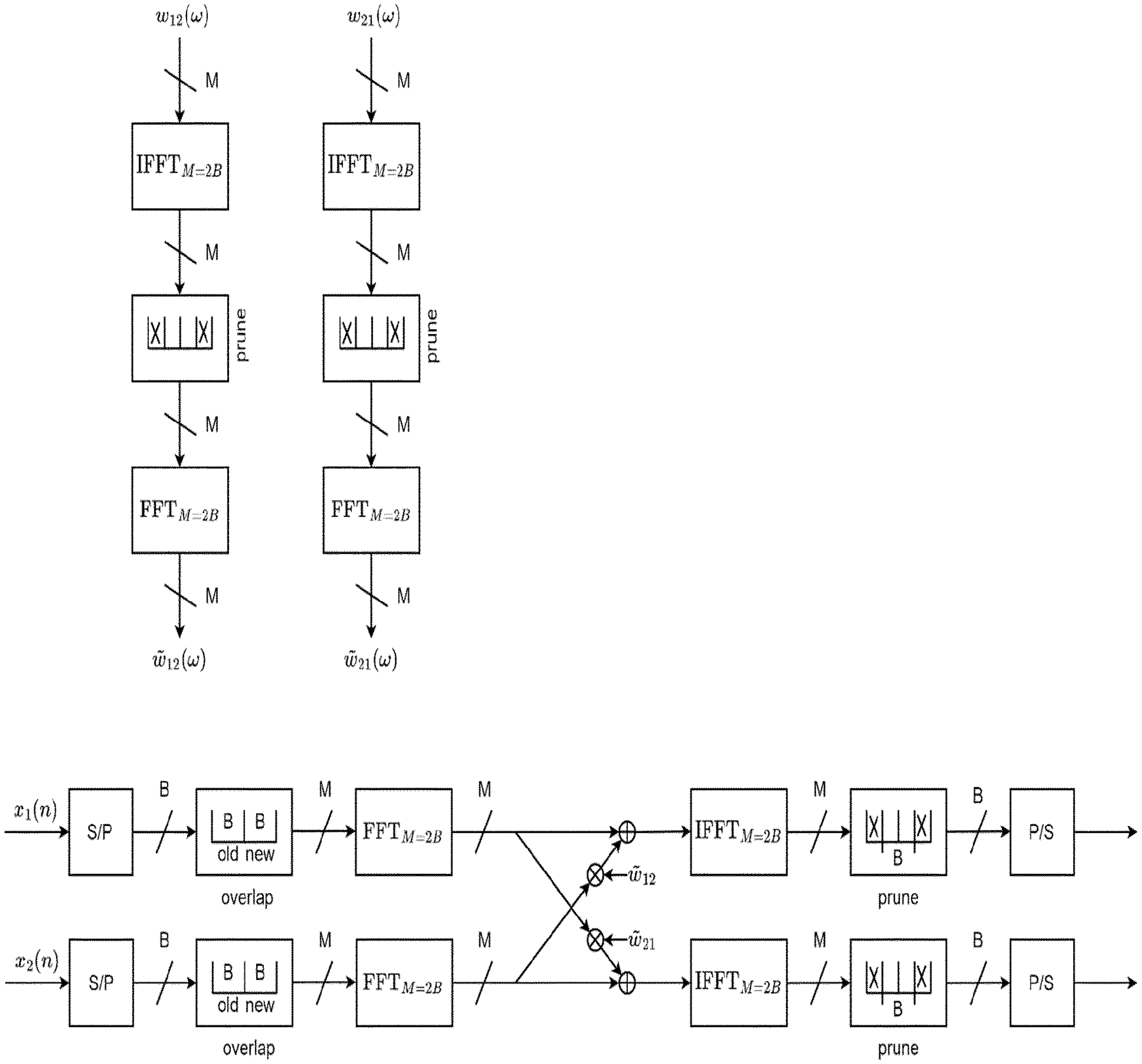
$w_{12}(\omega)$
$w_{21}(\omega)$
M
M
$\mathrm{IFFT}_{M=2B}$
$\mathrm{IFFT}_{M=2B}$
M
M
prune
prune
M
M
$\mathrm{FFT}_{M=2B}$
$\mathrm{FFT}_{M=2B}$
M
M
$\tilde{w}_{12}(\omega)$
$\tilde{w}_{21}(\omega)$
$x_1(n)$
S/P
B
B B
old new
overlap
M
$\mathrm{FFT}_{M=2B}$
M
$\tilde{w}_{12}$
$\mathrm{IFFT}_{M=2B}$
M
B
prune
B
P/S
$x_2(n)$
S/P
B
B B
old new
overlap
M
$\mathrm{FFT}_{M=2B}$
M
$\tilde{w}_{21}$
$\mathrm{IFFT}_{M=2B}$
M
B
prune
B
P/S

FIG. 7

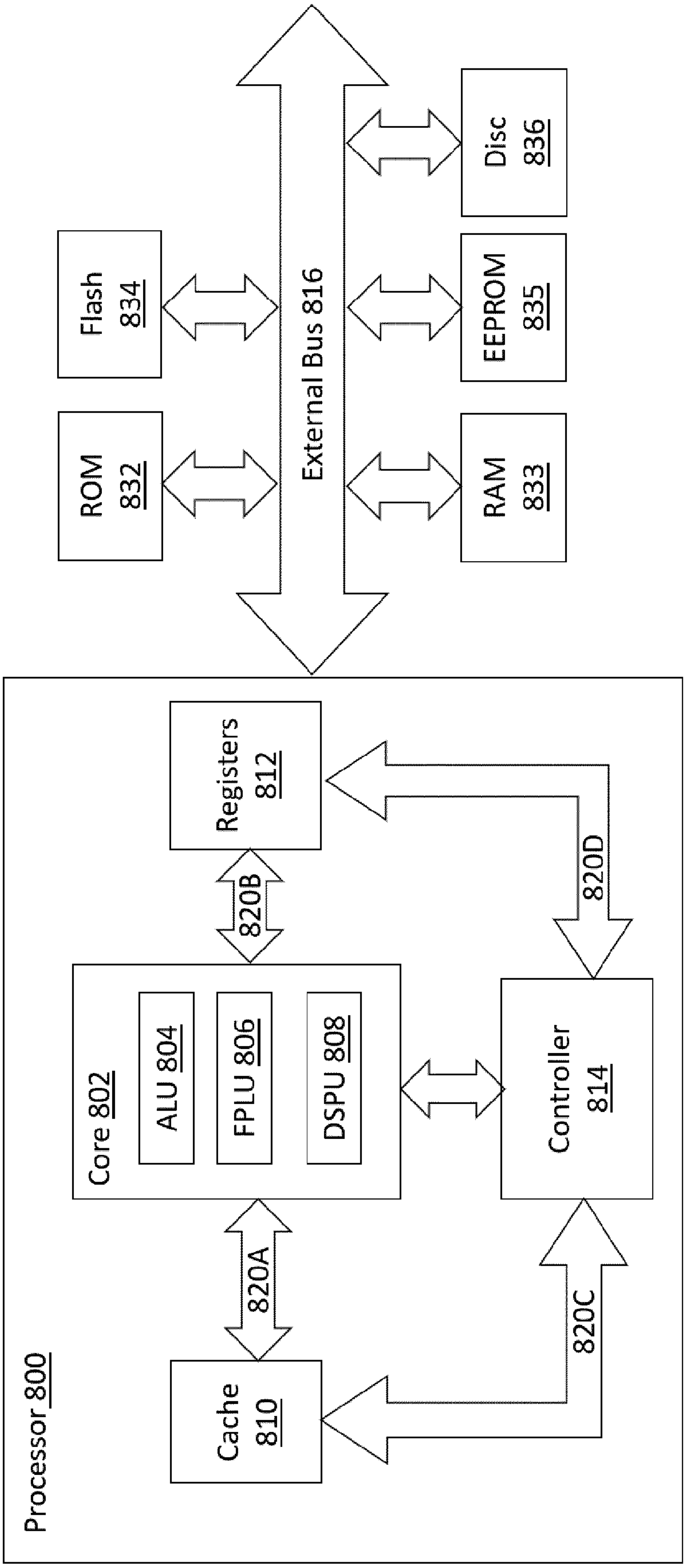
External Bus 816
ROM 832
Flash 834
RAM 833
EEPROM 835
Disc 836
Processor 800
Registers 812
820B
820D
Core 802
ALU 804
FPLU 806
DSPU 808
Controller 814
820A
820C
Cache 810

FIG. 8

Europäisches Patentamt
European Patent Office
Office européen des brevets

EUROPEAN SEARCH REPORT

Application Number
EP 23 17 6031

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
X	EP 2 551 850 A1 (DOLBY LAB LICENSING CORP [US]) 30 January 2013 (2013-01-30) * figure 1 * * paragraphs [0002], [0003], [0026] - [0034], [0048], [0049] * -----	1-14	INV. G10L21/0308 ADD. G10L25/18
			TECHNICAL FIELDS SEARCHED (IPC)
			G10L

The present search report has been drawn up for all claims

1

Place of search	Date of completion of the search	Examiner
Munich	18 September 2023	Ramos Sánchez, U

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons
....
& : member of the same patent family, corresponding document

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO.

EP 23 17 6031

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

18-09-2023

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 2551850 A1	30-01-2013	CN 102903368 A	30-01-2013
		EP 2551850 A1	30-01-2013
		US 2013031152 A1	31-01-2013

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82