



(11) **EP 4 826 716 A1**

(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
21.10.2026 Bulletin 2026/43

(51) International Patent Classification (IPC):
G10L 21/0272^(2013.01) G10L 21/0208^(2013.01)

(21) Application number: **25170686.7**

(52) Cooperative Patent Classification (CPC):
G10L 21/0272; G10L 21/0208; G10L 2021/02087

(22) Date of filing: **15.04.2025**

(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:
GE KH LA MA MD TN

(72) Inventors:
• **VENKATESH, Satvik**
London, N6 5EG (GB)
• **ROSKAM, Frederic**
London, N6 5EG (GB)

(74) Representative: **Novagraaf Technologies**
2 rue Sarah Bernhardt
CS90017
92665 Asnières-sur-Seine Cedex (FR)

(71) Applicant: **L-Acoustics UK Limited**
London N6 5EG (GB)

(54) **AUDIO SOURCE SEPARATION**

(57) An ML-based approach for audio source separation applicable to two or more microphones M1 to M4 and audio sources is disclosed herein. This audio source separation approach is based on a proximity-based assignment rule according to which each audio source is assigned to a single microphone which is the closest microphone. At the output of the trained ML model 110, for a given microphone, the processed audio signal PS1 to PS4 includes the source audio content emitted by any audio source assigned to the considered microphone, if any, and separated from the other source audio content(s), if any. The ML model 110 is trained using a training data set for various scenarios on the basis of this proximity-based assignment rule.

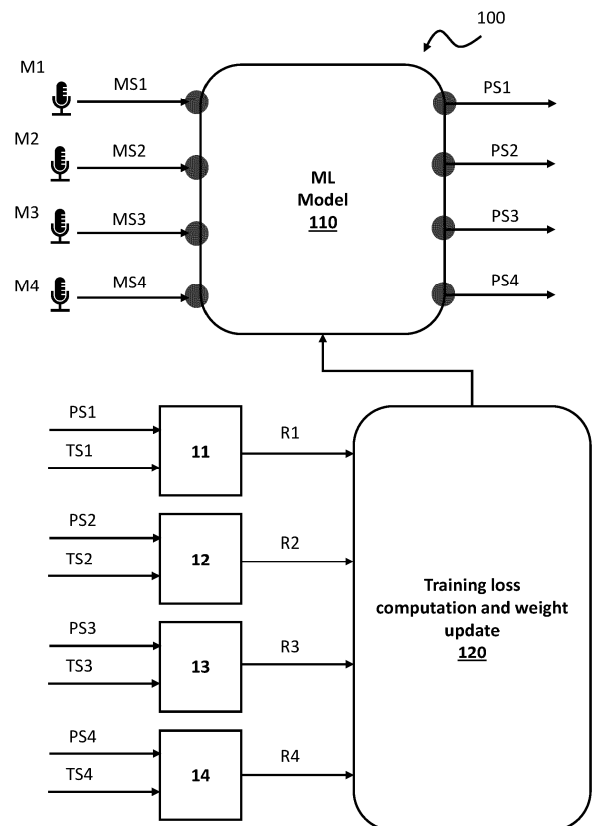


FIG. 1

EP 4 826 716 A1

Description

TECHNICAL FIELD

[0001] Various example embodiments relate generally to a method for training a Machine Learning (ML) model for audio source separation, a method for audio source separation and corresponding processing devices.

BACKGROUND

[0002] Audio Source Separation can be broadly defined as the process of separating an audio signal into separate audio contents corresponding respectively to distinct audio sources. Audio Source Separation comprises various sub-problems such as blind source separation, music source separation, speech separation.

[0003] Audio source separation can be easily performed in scenarios where the audio sources are of different classes, such as, on the one hand, (1) drums, bass, vocals, and on the other hand, (2) dry voice, reverb of voice, and residual noise. In these cases, a ML model may be trained to learn a unique representation for each of the audio source types and learn to separate them from the mixture captured by a microphone.

[0004] However, the audio source separation problem becomes more complicated when two audio sources are of the same class. The ML model may learn to separate the speech from different speakers, but there is no way to determine which speaker is captured by which microphone. Personalized speech enhancement is a way to solve this problem of separating multiple sources of the same class. The ML model may learn an embedding of a particular speaker and be trained to separate only that specific speaker. A drawback of this approach is that it requires offline recordings of that speaker.

[0005] Another way to solve the audio source separation problem is by using a microphone array or closely spaced microphones to focus on specific directions of sound. The ML model may be trained to enhance an audio source only from a particular direction. However, a few disadvantages of this approach include: (1) the direction of the audio source needs to be predetermined, (2) the microphones need to be closely spaced (such as a few centimeters) for effective beamforming and (3) the delay-and-sum performed on the multiple microphone channels can lead to unpleasant comb-filtering artifacts.

[0006] There appears a need for an improved method for audio source separation that is applicable to any type of audio sources, whatever the distances between the microphones, the directions and/or relative positions of the audio sources with respect to the microphones and any type of microphone.

SUMMARY

[0007] According to some aspects, there is provided the subject matter of the independent claims. Some further aspects are defined in the dependent claims.

[0008] The embodiments, examples and features, if any, described in this specification that do not fall under the scope of protection are to be interpreted as examples useful for understanding the various embodiments or examples that fall under the scope of protection.

[0009] According to a first aspect, a method for obtaining a trained machine learning, ML, model adapted for audio sources separation is disclosed, the method comprising: obtaining a training dataset comprising at least one set of training audio signals, each set of training audio signals corresponding to a scenario according to which N training audio sources and M training microphones have respective spatial positions in a training spatial area, N being an integer equal or greater than 1 and M being an integer equal or greater than 2; wherein a training audio signal in a set of training audio signals comprises audio content received at the spatial position of an associated training microphone when source audio contents are emitted respectively at the spatial positions of the N training audio sources; training a ML model with said training dataset, wherein for each scenario and associated set of training audio signals, said ML model is trained to generate, for each or at least one considered training microphone, an associated processed audio signal, said associated processed audio signal comprising: the source audio content emitted at the spatial position of any training audio source assigned to the considered training microphone, if any, and separated from the other source audio content(s), wherein a training audio source assigned to a considered training microphone is a training audio source for which the considered training microphone is the closest training microphone in the considered scenario; outputting the trained ML model.

[0010] In embodiments, the associated processed audio signal may comprise, for each audio source, if any, that is not assigned to the considered training microphone, either silence or the audio content of the considered audio source at a reduced sound level.

[0011] In embodiments, at least one training audio signal of at least one set of training audio signals may be a simulated training audio signal acoustically simulated based on: (i) the source audio contents emitted respectively at the spatial positions of the N training audio sources; and (ii) the relative spatial position between the N training audio sources and the training microphone associated to the simulated training audio signal.

[0012] In embodiments, the at least one training audio signal is acoustically simulated based at least on one of or a combination of factors among: (i) audio interferences between the emitted audio contents, (ii) a room impulse response associated with the training

spatial area, (iii) a propagation attenuation between a position of a training audio source and a position of a training microphone, (iv) a time delay in arrival of sound between a position of a training audio source and a position of a training microphone; (v) at least one property of the associated training microphone, wherein the at least one property is taken among a transducer type, a directionality, a frequency response, a sensitivity, an impedance, a self-noise, a signal-to-noise ratio, a maximum sound pressure level.

[0013] In embodiments, for at least one scenario, the associated set of training audio signals may further comprise, for the at least one considered training microphone, an associated target audio signal.

[0014] In embodiments, the method may further comprise: comparing, for each or the at least one considered training microphone, the associated processed audio signal and the associated target audio signal to generate a comparison result relative to the at least one scenario.

[0015] In embodiments, the method may further comprise during the training of the ML model: computing a training loss function based on at least one comparison result generated respectively for at least one scenario; and updating the ML model on the basis of the training loss function.

[0016] The target audio signal associated with a given microphone may comprise, for each audio source that is not assigned to the given microphone, either silence or the audio content of the considered audio source at a reduced sound level.

[0017] In embodiments, at least one target audio signal may be a dereverberated audio signal when at least one set of training audio signals is generated based a room impulse response associated with the training spatial area.

[0018] Each target audio signal associated with a considered training microphone may comprise the audio content emitted at the spatial position of each training audio source assigned to the considered training microphone and delayed by a time delay based on a time of flight between the relative spatial position of the considered assigned training audio source and the considered training microphone.

[0019] In embodiments, the training audio signals are such that, over the scenarios:

- a geometry of the training spatial area may vary; and/or
- the number N of training audio sources may vary; and/or
- the spatial position of at least one training microphone may vary; and/or
- the spatial position of at least one training audio source may vary; and/or
- the number M of training microphones may vary; and/or
- at least one characteristic of the audio content

emitted at a spatial position of at least one training audio source may vary, where the at least one characteristic may be in a set of characteristics comprising: a volume of the audio content, a class of the audio content, a frequency content of the audio content, a sample rate of the audio content.

[0020] The training dataset may comprise at least one specific set of training audio signals comprising only audio contents of at least one specific class of audio sources, wherein the trained ML model is a specialized trained ML model trained with said at least one specific set of training audio signals.

[0021] According to a second aspect, a method for separating audio sources is disclosed, the method comprising: obtaining a trained ML model configured for audio sources separation using the method according to the first aspect; obtaining a set of audio signals captured respectively by a plurality of microphones in a spatial area including audio sources configured to generate respective source audio contents; processing, with the trained ML model, the set of audio signals to generate, for each or at least one considered microphone among the plurality of microphones, an associated processed audio signal, said associated processed audio signal comprising: the source audio content emitted by any audio source assigned to the considered microphone if any and separated from the other source audio content(s), wherein an audio source assigned to the considered microphone is an audio source for which the considered microphone is the closest microphone.

[0022] In embodiments, the associated processed audio signal may comprise, for each audio source, if any, that is not assigned to the considered microphone, either silence or the audio content of the considered audio source at a reduced sound level.

[0023] The processing of the set of audio signals may be carried out in real time while the relative spatial position of at least one audio source varies in the spatial area.

[0024] In embodiments, the method according to the second aspect may comprise: outputting at least one among: the number of audio sources in each set of audio signals, the different classes of audio sources present in the audio signals, a list of audio sources assigned to a given microphone.

[0025] In embodiments, the method according to the second aspect may comprise denoising at least one processed audio signal, wherein denoising said at least one processed audio signal comprises suppressing audio content relative to at least one undesirable class of audio sources from said at least one processed audio signal.

[0026] The additional technical features related to the method for obtaining a trained ML model previously described are also applicable to the method for separating audio sources described above.

[0027] According to another aspect, a processing device comprises means for performing one or more or all steps of any method disclosed herein, for example a method according to the first aspect or according to the second aspect. The means may for example include at least one processor and at least one memory storing instructions that, when executed by the at least one processor, cause the processing device to perform one or more or all steps of any method disclosed herein. The means may for example include circuitry (e.g., processing circuitry) configured to perform one or more or all steps of any method disclosed herein.

[0028] According to another aspect, a computer program comprises instructions that, when executed by a processing device, cause the processing device to perform one or more or all steps of any method disclosed herein. The instructions may cause the processing device to perform one or more or all steps of any method disclosed herein.

[0029] According to another aspect, a non-transitory computer readable medium comprises computer program instructions stored thereon for causing a processing device to perform one or more or all steps of any method disclosed herein. The computer program instructions may cause the processing device to perform one or more or all steps of any method disclosed herein.

BRIEF DESCRIPTION OF THE DRAWINGS

[0030] Some example embodiments will now be described with reference to the accompanying drawings.

FIG. 1 is a block diagram of a system for training a ML model for audio source separation according to an example.

FIG. 2 is a schematic diagram illustrating a scenario according to an example.

FIG. 3 is a schematic diagram illustrating the generation of target (also called ground truth) audio signals in an example scenario.

FIG. 4 is a schematic diagram illustrating the generation of target (also called ground truth) audio signals in an example scenario.

FIG. 5 is a schematic diagram illustrating the generation of target (also called ground truth) audio signals in an example scenario.

FIG. 6 is a schematic diagram illustrating aspects of a method for audio source separation.

FIG. 7 is a block diagram of a ML model adapted for audio source separation according to an example.

FIG. 8 is a flowchart illustrating a method for obtaining a trained ML model adapted for audio sources separation according to an example.

FIG. 9 is a flowchart illustrating a method for separating audio sources during an inference phase according to an example embodiment.

FIG. 10 is a block diagram illustrating an exemplary hardware structure of a processing device according to an example.

[0031] It should be noted that these drawings are intended to illustrate various aspects of devices, methods and structures used in example embodiments described herein. The use of similar or identical reference numbers in the various drawings is intended to indicate the presence of a similar or identical element or feature.

DETAILED DESCRIPTION

[0032] Detailed example embodiments are disclosed herein. The example embodiments are given by way of illustration only and thus are not limiting of this disclosure. These example embodiments may be embodied in many alternate forms, with various modifications, and should not be construed as limited to only the embodiments set forth herein. In addition, the figures and descriptions may have been simplified to illustrate elements and / or aspects that are relevant for a clear understanding of the present invention, while eliminating, for purposes of clarity, many other elements that may be well known in the art or not relevant for the understanding of the invention.

[0033] An ML-based approach for audio source separation applicable to any number of microphones (the number of microphones being $M \geq 2$) and any number of audio sources (the number of audio sources being $N \geq 1$) is disclosed herein. This audio source separation approach is based on a proximity-based assignment rule according to which each audio source is assigned to a single microphone which is the closest microphone. At the output of the trained ML model, for a given microphone, the processed audio signal includes the source audio content emitted by any audio source assigned to the considered microphone, if any, and separated from the other source audio content(s), if any. The ML model is trained using a training data set for various scenarios on the basis of this proximity-based assignment rule.

[0034] For example, the processed audio signal comprises, for each audio source that is not assigned to the considered microphone, if any, either silence or the audio content of the considered audio source at a reduced sound level.

[0035] This is useful for various application like panel discussions, live band performance and acoustic monitoring of machines.

[0036] For example, in a scenario with a live band performance with a respective microphone per musician, or the microphone near the drummer, the processed audio signal includes the drum sounds clearly, while other sounds are either silent or at a reduced level.

[0037] For example, in a scenario with a live panel

discussion with a respective microphone per panelist, for the microphone near a specific panelist, the processed audio signal includes that panelist's voice clearly, while other voices are either silent or at a reduced level.

[0038] By isolating the audio content of the closest source(s) for each microphone, the processed audio signal is clearer and less affected by interfering sources. The other audio sources are either silent or at a reduced level, minimizing interference and enhancing the quality of the separated audio source(s). This gives flexibility during audio mixing of live shows, leading to an improved overall output.

[0039] This proximity-based approach does not require to differentiate between near and far audio sources, e.g., based on some distance threshold around a microphone, but focus on assigning the closest microphone to each audio source. This proximity-based approach exploits implicitly the relative distances across all microphones and also relative distances between the audio sources and the microphones rather than trying to determine an appropriate distance threshold from a microphone to exclude the audio content of far audio sources from the audio content captured by this microphone.

[0040] When assigning each audio source to a single microphone based on its physical proximity (e.g., that can be measured in term of time of flight of a sound signal), multiple sources may be assigned to the same microphone; or none of the audio sources may be assigned to a microphone, as these audio sources are closer to other microphones.

[0041] Moreover, the assignment of the audio sources to the microphones may change dynamically. The ML model is trained taking into account this assignment constraint to maximize its audio source separation performance. This proximity-based approach works in real-time, since the positions in space of the microphones and audio sources may vary in any desired manner in a spatial area. The proximity-based approach works for both live and offline processing of audio signals. The trained ML model is adapted for real-time applications such as acoustic machine monitoring, wildlife monitoring, smart homes, and low-latency scenarios like live concerts and music festivals, etc.

[0042] The ML model can also be used for offline scenarios without the real-time constraints such as audio mixing in a studio, by processing data based on the proximity-based assignment rule.

[0043] Further, the proximity-based approach is independent of the type of audio class and can work with a variety of audio classes such as speech, music, machine sounds, animal voice, or noise sources. The proximity-based approach depends only on the proximity between the source and the considered microphone, and not on other factors such as the type of source, type of microphone, or direction of arrival.

[0044] Examples of application of the proximity-based approach include:

- Live sound de-bleed;
- Separation of voices of actors in a theatre, when the actors are using respective microphones;
- Musical instruments such as Drums and Violin on stage;
- Orchestral performances.

[0045] In a first example scenario, the ML model may be trained with microphones placed around a stage and audio sources like singers and instruments. Microphones capture mixed audio signals from the audio sources to generate training audio signals. The ML model generates separated processed audio signals for each microphone, isolating the closest audio source(s).

[0046] In a second example scenario, the ML model may be trained with microphones placed around a conference room and audio sources like speakers and participants. The microphones capture mixed audio signals from multiple speakers to generate training audio signals. The ML model generates separated processed audio signals for each microphone, isolating the closest speaker's voice(s). Particularly, in the case where each speaker/participant has a dedicated microphone next to them (regardless of the nature or of the orientation of the dedicated microphone), the processed audio signal generated by the ML model for a dedicated microphone considered will only comprise the closest speaker's voice.

[0047] The trained ML model can thus handle various types of microphone configurations and scenarios unlike traditional methods that require specific setups such as, (1) microphone arrays to capture sound from certain distances, (2) sound buffers to reduce interference between sources, (3) dependence on microphone directivity to isolate sources. The proximity-based approach simplifies the separation process by focusing on the closest microphone for each audio source.

[0048] Further, since the spatial position of a microphone in a spatial area determines which audio sources will be assigned to this microphone and separated from the other audio sources, the proximity-based approach enables applications such as:

- use of a "discard microphone" positioned close to an audio source (such as the crowd, a noisy source, etc.) to be excluded from the audio signals captured by the other microphones, as this audio source will be assigned to the "discard microphone" and be therefore separated from the other audio sources;
- capture of sound sources of various classes, using one or more microphones dedicated to a given class and positioned close to the audio sources of the given class to achieve separation of the

classes of audio sources (e.g., in city transport, nature, transport engineering, noise pollution, etc.).

[0049] The word "microphone" is used herein to designate an audio capturing device. An audio capturing device is configured to capture sound waves and convert the sound waves into electrical signals, which can be stored as digital or analog audio signals. Any type of audio capturing device may be considered. For example, a microphone may correspond to a microphone array including a plurality of audio sensors, each capturing a respective sound wave: in such a case, the microphone array generates an output signal from one or more of the captured sound waves.

[0050] Although a microphone is illustrated herein as a standalone device, it can be integrated within a computing device, a speaker, a headset, a mobile communication device, a camera, etc.

[0051] As used herein, an audio signal may correspond to any digital or analog signal carrying information representative of a sound wave.

[0052] FIG. 1 is a block diagram of a system 100 for training a ML model 110 for audio source separation according to an example.

[0053] The system includes a ML model 110 and an optional functional block 120 for computation of a training loss and adjustment of the weights of the ML model 110.

[0054] The system 100 is described in an example in which the number of microphones is equal to $M=4$ but the description can be generalized to any number M of microphones, with M being an integer greater than or equal to 2.

[0055] The microphones used or simulated during the training phase of the ML model are referred to herein as the "training microphones" to distinguish them from the microphones that may be used during the inference phase.

[0056] The ML model receives $M=4$ input audio signals MS1 to MS4, each input audio signal MS1 to MS4 corresponding to an audio signal received and/or captured at a spatial position of a corresponding training microphone M1 to M4. An input audio signal MS1 to MS4 is also referred to herein as a "microphone signal". In the context of the training of a ML model, the input audio signals MS1 to MS4 are also referred to herein as "training audio signals".

[0057] Two or more audio sources are used for the training of the ML model and are referred to herein as "training audio sources" to distinguish them from the audio sources that may be used during the inference phase. The number of training audio sources is noted N , with N being an integer greater than or equal to 1.

[0058] Each microphone M1 to M4 may receive audio content from one or more training audio sources and generate an audio signal corresponding to a mixture of the captured audio contents.

[0059] As known in the art, a ML model may be a

computational program and/or circuitry that learns to perform some processing on input data (e.g., make predictions or decisions) without being explicitly programmed to do so. The ML model is created by training an algorithm using a training dataset.

[0060] The ML model 110 is a ML model adapted for performing audio source separation. This ML model may comprise one or more neural networks. The ML model may comprise a combination of neural networks.

[0061] The ML model 110 may be configured to merge the input audio signals MS1 to MS4 into a latent space representation and to draw relationships between the input audio signals MS1 to MS4. The ML model 110 may be trained to separate the input audio signals MS1 to MS4 according to the proximity-base assignment rule.

[0062] The ML model 110 may be a neural network with recurrent layers, neural network with convolutional layers, neural network with transformer layers. For example, the ML model 110 may be any model configured for multi-channel sound source separation, like multi-channel speech separation and/or multi-channel music source separation, such as: a LSTM-TasNet, a Conv-TasNet, a HS-TasNet, a MTFAA-Net.

[0063] A LSTM-TasNet is a deep learning model adapted for speech separation, specifically time-domain audio separation. Also called TasNet (Time-domain Audio Separation Network), this particular deep learning model is based on Long Short-Term Memory (LSTM) units for improved sequential processing.

[0064] A Conv-TasNet is a deep learning model designed for speech separation in the time domain, using convolutional networks (CNNs) instead of recurrent networks (like LSTMs). It is an improvement over LSTM-TasNet, offering better efficiency and performance in separating mixed speech signals.

[0065] A HS-TasNet (Hybrid Spectrogram Time-domain Audio Separation Network) is a deep learning model adapted for real-time, low-latency music source separation. It combines the strengths of both spectrogram-based and time-domain (waveform-based) approaches to effectively isolate individual components, such as vocals, drums, bass, and other instruments, from a mixed audio signal.

[0066] A MTFAA-Net, or Multi-Scale Temporal Frequency Convolutional Network with Axial Attention, is a deep learning model adapted for speech enhancement tasks. It integrates multi-scale temporal and frequency feature extraction with axial attention mechanisms to effectively suppress noise and improve speech quality.

[0067] Various classes of training audio source may be used for the training. Examples of classes of audio source include:

- human or animal vocal sources;
- instrumental audio sources;
- noise sources;
- machine sounds.

[0068] Instrumental audio sources may include for example but not limited to:

- string instrument such as violins, viola, cello, double bass, erhu, guitar, harp, banjo, mandolin, lute, piano, dulcimer, sarod, sitar, veena;
- wind instrument such as trumpet, trombone, tuba, French horn, flute, clarinet, saxophone, oboe bassoon, recorder;
- percussion instruments such as: xylophone, marimba, vibraphone, timpani, snare, drum, bass drum, cymbals, triangle, tambourine;
- keyboard instrument such as: piano, organ, harpsichord, synthesizer;
- electronic instrument such as: synthesizer, theremin, electric guitar, drum machine; or
- folk and/or traditional instrument such as: djembe, kora, balfon, sitar, shamisen, guzheng, oud, ney, darbuka, charango, cuotro, bandoneon, mridangam, tabla, ghatam.

[0069] Machine sounds may include for example: fan, gearbox, bearing, Toy Car, etc.

[0070] Noise sources may include for example: fan, air conditioner, typing, door shutting, clatter noise, kitchen noise (noise from kitchen utensils, dish scrubbing etc.), dish washer, running water, opening chips bags, munching or eating, creaking chair, heavy breathing, copy machine, baby crying, dog barking, inside-car noise (e.g., sitting on a passenger seat in a car which is being driven by someone else), mouse clicks, mouse scroll wheel, touch pad clicks, etc.

[0071] To train the ML model, a training dataset is generated. The training dataset can either be generated before the training or on-the-fly while training the ML model.

[0072] The training dataset may cover various scenarios for which the ML model is to be trained. As used herein, a scenario corresponds to a configuration according to which the N training audio sources and the M training microphones have respective spatial positions in a training spatial area.

[0073] For each scenario, a set of training audio signals MS1 to MS4 is collected. Each training audio signal (MS1 to MS4) represents the audio content captured by the associated training microphone (respectively M1 to M4) at the spatial position of the associated training microphone when the N training audio sources are emitting source audio contents from their respective spatial positions.

[0074] The training audio signals MS1 to MS4 provided as input to the ML model for training may include (i) real world audio signals captured by the training microphones in a training spatial area including N audio sources and/or (ii) acoustically simulated audio signals.

[0075] There are a few challenges when using real-world data for training: for a live audio source such as a guitar amid other interfering sources, it is not possible

to simultaneously obtain the target audio signals and the training audio signals. While it is possible to use one close-microphone and one distant-microphone to obtain the target and training audio signals respectively because the close-microphone will have less interference from neighboring audio sources. Also, with real-world data, the audio content for audio sources can be pre-recorded and emitted through speakers, the target audio signals can be known in advance, and the training audio signals can be recorded. However, it is time-consuming to explore a variety of training scenarios, and the speaker may have its own polar pattern.

[0076] Using acoustically simulated training audio signals allows the creation of many various training scenarios without the need for real audio capture in every step. This speeds up the ML model training process by generating a large amount of training data through simulation.

[0077] Simulated training audio signals can be generated (e.g., using software) to generate audio signals as those that would be captured by microphones placed in the training area for each scenario. For example, in a training scenario, the microphones may be placed around a concert hall and audio sources like a symphony orchestra.

[0078] When acoustically simulated, the training audio signals MS1 to MS4 may be generated (e.g., by a processing device) from M source audio signals (whether real audio signals or synthesized audio signals) corresponding respectively to the audio content emitted at the spatial positions of the audio sources S1 to SN (not represented). The acoustically simulated training audio signals MS1 to MS4 may be generated for a training spatial area including the N audio sources. The acoustically simulated training audio signals MS1 to MS4 may correspond to audio signals received at the respective spatial positions of M simulated microphones (in Fig. 1, M=4) when the N audio sources have respective spatial position in the training spatial area. The acoustically simulated training audio signals MS1 to MS4 thus represents the audio content that would have been captured by respective training microphones positioned at the considered spatial positions.

[0079] The ML model 110 is trained to generate, for each training audio signal MS1 to MS4 or, equivalently, for each training microphone M1 to M4, an associated processed audio signal, PS1 to PS4. In embodiments, however, the ML model 110 may generate a lower number of processed audio signals. For example, a user may select one or more microphones of interest for which a processed audio signal is to be generated. For example, the ML model 110 may be configured to not generate a processed audio signal when the processed audio signal would include only silence or audio at a reduced sound level, preferably a very low sound level (e.g., below a threshold).

[0080] A processed audio signal PS1 to PS4, asso-

ciated with a training microphone M1 to M4, comprises the source audio content emitted at the spatial position of any training audio source assigned to the considered training microphone, if any, and separated from the other source audio content(s), if any. A proximity-based assignment rule is used here to assign each of the training audio sources to a single training microphone, such that a training audio source assigned to a given training microphone is a training audio source for which the given training microphone is the closest training microphone in the considered scenario.

[0081] For at least one or each scenario, the associated set of training audio signals MS1 to MS4 may comprise, for each or at least one considered training microphone M1 to M4, an associated target audio signal TS1 to TS4. A target audio signal TS1 to TS4 corresponds to a target (also called ground truth) audio signal that the ML model learns to generate for the considered training microphone M1 to M4 and training audio signal MS1 to MS4.

[0082] The target audio signal TS1 to TS4 provided as ground truth for training may be generated (e.g., by a processing device) from source audio signals (whether real audio signals or synthesized audio signals) corresponding respectively to the audio content emitted at the spatial positions of the audio sources.

[0083] Using target audio signals TS1 to TS4 helps the ML model to become better at separating audio sources by understanding what the ideal separated audio content should be. It provides a clear reference for training, guiding the ML model to achieve high-quality separation. By comparing the processed audio signals with the target signals, the ML model can be optimized to minimize the error.

[0084] For example, in a training scenario with real recordings of an orchestra, the target audio signals used as ground truth for training may be the ideally isolated audio content produced by each instrument.

[0085] For example, in a training scenario with real recordings of a wild life monitoring setup, the target audio signal used as ground truth for training may be the ideally isolated audio content of each animal sound.

[0086] Furthermore, metrics such as mean squared error (MSE) or scale-invariant signal-to-distortion ratio (SI-SDR) can also be calculated to assess how well the ML model is performing.

[0087] The system 100 of Fig. 1 may include at least one comparison block 11, 12, 13, 14. Each block 11, 12, 13, 14 is adapted to compare, for a training microphone, the associated processed audio signal (e.g., PS1) and the associated target audio signal (e.g., TS1) to generate a comparison result (e.g., R1) for the considered scenario. Examples of comparison result generated may include a loss (such as mean absolute error, means square error, signal to distortion ratio, signal to noise ratio...), a distance metric, an error.

[0088] The functional block 120 is adapted to compute a training loss based on one or more comparison

results R1 to R4, generated for one or more scenario(s), and to update the ML model 110 (e.g., weights of the ML model) on the basis of the training loss.

[0089] FIG. 2 is a schematic diagram illustrating a scenario according to an example. Such a scenario may correspond to a real live scenario or a training scenario.

[0090] In this scenario, there are M=4 microphones M1 to M4 and N=4 audio sources S1 to S4, having respective spatial positions in a spatial area such that, according to the proximity-based assignment rule:

- the audio source S1 is assigned to the closest microphone M1;
- the audio source S2 is assigned to the closest microphone M2;
- the audio source S3 is assigned to the closest microphone M3;
- the audio source S4 is assigned to the closest microphone M4;

[0091] Each microphone M1 to M4 receives audio content from all audio sources S1 to S4, but with a received sound level and/or time-delay that depend on the sound attenuation along the propagation path between the considered audio source and the considered microphone.

[0092] Microphone M1 captures an audio signal MS1 but after audio signal separation, only audio content of audio source S1 is kept in the processed audio signal PS1.

[0093] Likewise, microphone M2 captures an audio signal MS2 in which, after audio signal separation, only audio content of audio source S2 is kept in the processed audio signal PS2. The same applies to microphone M3 which captures an audio signal MS3 in which, after audio signal separation, only audio content of audio source S3 is kept in the processed audio signal PS3 and to microphone M4 which captures an audio signal MS4 in which, after audio signal separation, only audio content of audio source S4 is kept in the processed audio signal PS4.

[0094] Each of FIGS. 3 to 5 is a schematic diagram illustrating the generation of target audio signals (or processed audio signals) in an example scenario.

[0095] Each scenario may correspond to a scenario used either (i) during an inference phase to generate processed audio signals PS1 to PS4 with a trained ML model or (ii) for training the ML model based on real and/or simulated training audio signals MS1 to MS4 when target audio signals TS1 to TS4 are used.

[0096] When a scenario is used during the inference phase, the trained ML model is configured to generate processed audio signals PS1 to PS4 corresponding respectively to the target audio signals TS1 to TS4 used for the training of the ML model in the same scenario.

[0097] In the scenario of FIG. 3, there are M=4 microphones M1 to M4 and N=3 audio sources S1 to S3,

having respective spatial positions in a spatial area (also called hereafter the "training" spatial area when used during the training of the ML model) such that, according to the proximity-based assignment rule:

- the audio source S1 (vocal audio source) is assigned to the closest microphone M1;
- the audio source S2 (instrumental audio source) is assigned to the closest microphone M2;
- the audio source S3 (vocal audio source) is assigned to the closest microphone M3;
- no audio source is assigned to the microphone M4.

[0098] When this scenario is used as a training scenario, the target audio signals (i.e. ground truth audio signals) may be generated such that:

- the target audio signal TS1 associated with microphone M1 and training audio signal MS1 includes (in an ideal case) only the audio content of audio source S1; optionally, this target audio signal may further comprise audio content of one or more or all the other audio sources (i.e. S2, S3 and/or environmental noise), but at a reduced sound level;
- the target audio signal TS2 associated with microphone M2 and training audio signal MS2 includes (in an ideal case) only the audio content of audio source S2; optionally, this target audio signal may further comprise audio content of one or more or all the other audio sources (i.e. S1, S3 and/or environmental noise), but at a reduced sound level;
- the target audio signal TS3 associated with microphone M3 and training audio signal MS3 includes (in an ideal case) only the audio content of audio source S3; optionally, this target audio signal may further comprise audio content of one or more or all the other audio sources (i.e. S1, S2 and/or environmental noise), but at a reduced sound level;
- the target audio signal TS4 associated with microphone M4 and training audio signal MS4 includes (in an ideal case) only silence; optionally, this target audio signal may further comprise audio content from one or more or all the audio sources (i.e. S1, S2, S3 and/or environmental noise), but at a reduced sound level.

[0099] In the scenario of FIG. 4, there are M=4 microphones M1 to M4 and N=3 audio sources S1 to S3, having respective spatial positions in a spatial area (also called hereafter the "training" spatial area when used during the training of the ML model) such that, according to the proximity-based assignment rule:

- the audio sources S1 and S2 (vocal audio sources) are assigned to the closest microphone M1;
- the audio source S3 (instrumental audio source) is assigned to the closest microphone M2;
- no audio source is assigned to the microphones

M3 and M4.

[0100] When this scenario is used as a training scenario, the target audio signals (i.e. ground truth audio signals) may be generated such that:

- the target audio signal TS1 associated with microphone M1 and training audio signal MS1 includes (in an ideal case) only the audio contents of audio sources S1 and S2; optionally, this target audio signal may further comprise audio content of the other audio source (i.e. S3 and/or environmental noise), but at a reduced sound level;
- the target audio signal TS2 associated with microphone M2 and training audio signal MS2 includes (in an ideal case) only the audio content of audio source S3; optionally, this target audio signal may further comprise audio content of the one or more or all the other audio sources (i.e. S1, S2 and/or environmental noise), but at a reduced sound level;
- the target audio signal TS3 associated with microphone M3 and training audio signal MS3 includes (in an ideal case) only silence; optionally, this target audio signal may further comprise audio content from one or more or all audio sources (i.e. S1, S2, S3 and/or environmental noise), but at a reduced sound level;
- the target audio signal TS4 associated with microphone M4 and training audio signal MS4 includes (in an ideal case) only silence; optionally, this target audio signal may further comprise audio content from one or more or all audio sources (i.e. S1, S2, S3 and/or environmental noise), but at a reduced sound level.

[0101] In the scenario of FIG. 5, there is M=4 microphones M1 to M4 and N=5 audio sources S1 to S5 having respective spatial positions in a spatial area (also called hereafter the "training" spatial area when used during the training of the ML model), in which, according to the proximity-based assignment rule:

- the audio sources S1 and S2 (vocal audio sources) are assigned to the closest microphone M1;
- no audio source is assigned to the microphone M2;
- the audio source S3 (instrumental audio source) is assigned to the closest microphone M3;
- the audio sources S4 and S5 (vocal audio sources) are assigned to the closest microphone M4.

[0102] When this scenario is used as a training scenario, the target audio signals (i.e. ground truth audio signals) may be generated such that:

- the target audio signal TS1 associated with microphone M1 and training audio signal MS1 includes (in an ideal case) only the audio contents of audio sources S1 and S2; optionally, this target audio

signal may further comprise audio content of one or more or all the other audio sources (i.e. S3, S4, S5 and/or environmental noise), but at a reduced sound level;

- the target audio signal TS2 associated with microphone M2 and training audio signal MS2 includes (in an ideal case) only silence; optionally, this target audio signal may further comprise audio content from one or more or all audio sources (i.e. S1, S2, S3, S4, S5 and/or environmental noise), but at a reduced sound level;
- the target audio signal TS3 associated with microphone M3 and training audio signal MS3 includes (in an ideal case) only the audio content of audio source S3; optionally, this target audio signal may further comprise audio content of one or more or all the other audio sources (i.e. S1, S2, S4, S5 and/or environmental noise), but at a reduced sound level;
- the target audio signal TS4 associated with microphone M4 and training audio signal MS4 includes (in an ideal case) only the audio contents of audio sources S4 and S5; optionally, this target audio signal may further comprise audio content of the one or more or all the other audio sources (i.e. S1, S2, S3 and/or environmental noise), but at a reduced sound level.

[0103] In the example scenarios of FIGS. 3 to 5, any of the microphones may act as discard microphones, strategically placed to capture and isolate unwanted noise sources, ensuring they do not interfere with the primary audio signals. For example, in a concert setting, a discard microphone can be positioned close to a noisy crowd. This microphone captures the crowd noise, and the machine learning (ML) model assigns this noise to the discard microphone, effectively excluding this noise from the audio signals captured by other microphones. Similarly, in an outdoor environment, a discard microphone can be placed near a source of environmental noise, such as traffic or wind. The ML model assigns this environmental noise to the discard microphone, ensuring it is separated from the audio signals captured by other microphones. This approach enhances the overall audio quality by reducing interference and improving clarity, making it particularly useful in dynamic and noisy environments.

[0104] FIG. 6 is a schematic diagram illustrating aspects of a method for audio source separation.

[0105] This figure illustrates the differences between audio source separation based on a distance threshold around a microphone and audio source separation based on the proximity-based assignment rule as described herein.

[0106] In an example scenario, a spatial area includes M=3 microphones M1 to M3 and N=9 audio sources S1 to S9.

[0107] In the case of Fig. 6, if we use an audio source separation method that would be based on a differen-

tiation between near and far audio sources using a distance threshold:

- a spatial zone Z1 is defined around the microphone M1 to separate near audio sources from far audio sources with respect to microphone M1, such that only the audio content of the near audio sources S1, S2 and S3 located in the spatial zone Z1 would be kept by audio source separation in the audio signal captured by the microphone M1;
- likewise, a spatial zone Z2 is defined around the microphone M2 to separate near audio sources from far audio sources with respect to microphone M2, such that only the audio content of the near audio sources S3 and S4 located in the spatial zone Z2 would be kept by audio source separation in the audio signal captured by the microphone M2;
- likewise, a spatial zone Z3 is defined around the microphone M3 to separate near audio sources from far audio sources with respect to microphone M3, such that only the audio content of the near audio sources S6, S7 and S8 located in the spatial zone Z3 would be kept by audio source separation in the audio signal captured by the microphone M3.

[0108] Consequently, after audio separation based on a differentiation between near and far audio sources using a distance threshold, the audio content of audio sources S5 and S9 would not be kept after audio source separation in any of audio signal captured by the microphones. The audio content of audio sources S5 and S9 will therefore be "lost".

[0109] Further the audio content of audio source S3 will be kept after audio source separation both in the audio signal captured by the microphone M1 and also in the audio signal captured by the microphone M2. When summing electronically or acoustically the processed audio signals obtained for the microphones M1 and M2, the same audio source appears in both microphones' signals but with different delays (due to their different distances to the source). This creates unpleasant comb filtering effects (perceived as colouration) and transients' distortions (multiple time of arrivals).

[0110] In contrast, with a proximity-based approach as disclosed herein, the audio source S3 is assigned only to the closest microphone M1 and its audio content is kept, after audio source separation by the ML model, only in the audio signal captured by the microphone M1, but not in the audio signal captured by the microphone M2. Further, the audio source S5 is assigned to the closest microphone M2 and its audio content is kept, after audio source separation by the ML model, in the audio signal captured by the microphone M2. Likewise, the audio source S9 is assigned to the closest microphone M3 and its audio content is kept, after audio source separation by the ML model, in the audio signal captured by the microphone M3.

[0111] FIG. 7 is a block diagram of a ML model 700

adapted for audio source separation according to an example.

[0112] The ML model 700 is based on a HSTN architecture. The HSTN uses spectral and waveform features. This ML model 700 is adapted for audio separation for a variety of sources, such as speech, singing voice, musical instruments, and machine sounds, etc. The waveform branch processes audio directly in the time domain, which is beneficial for capturing transient and percussive sounds like drums, bass and other rhythmic elements. The spectral branch processes audio in the frequency domain, which is effective for separating harmonic and tonal components like vocals and melodic instruments. The architecture of FIG. 7 is a hybrid framework that harnesses the advantages of both the spectral and waveform features, especially by concatenating the outputs of two encoders 711, 712, as explained below.

[0113] This example ML model 700 has a very short latency, which is below 20 ms, e.g., between 5 to 10 ms, with input audio signals sampled at 48 kHz.

[0114] The ML model 700 comprises a first encoder 711 which is a spectrogram encoder that generates a spectrogram as a function of time and frequency for each input audio signal. The decoder 751 performs an inverse function and generates a processed audio signal in time domain.

[0115] The ML model 700 comprises a second encoder 712 which is a trained convolution encoder that generates a series of filtered signals as a function of time for each input audio signal by applying a filter bank of FIR (Finite Impulse Response) filters to the input audio signal. The decoder 752 performs an inverse function and generates a processed audio signal in time domain.

[0116] The ML model 700 accepts M microphone signals MS1, MS2, ..., as inputs. These microphone signals are fed into the respective encoders 711, 712 to generate encoded signals. Subsequently, these encoded signals are concatenated by block 721 and 722, along the feature dimension to generate concatenated signals. The concatenated signals may be passed through a linear layer to match the hidden dimension size of the LSTM.

[0117] The concatenated signals are fed to a series of LSTM layers 731 to 735. The output of LSTM layers 731 and 732 are summed at block 71 and provided to LSTM layer 733. The output of LSTM layers 733 is fed to both LSTM layers 734 and 735. Identity skip connections may be added between the inputs of LSTM layers 731 (respectively 732) and the outputs of LSTM layers 734 (respectively 735) (shown as dotted lines in FIG. 7).

[0118] From the output of LSTM layers 734 and 735, a mask is predicted for each microphone. This mask has the same dimensionality as the encoded representation at output of the encoder 711 and 712 and typically has values between 0 and 1. Therefore, a linear

layer is applied on the LSTM outputs to obtain the dimensionality as the encoded representation. Further, a sigmoid activation may be applied to ensure the values are between 0 and 1.

[0119] The predicted masks generated for the microphones are multiplied (by multipliers 73 and 74) with the respective encoded representations at output of the encoder 711 and 712 respectively. The signals at the output of the multipliers 73 and 74 are decoded by the decoders 751 and 752 respectively. After summing at block 75 the outputs of the decoders 751 and 752, M processed microphone signals PS1, PS2, ..., are obtained as the output of the network.

[0120] The ML model 700 is trained to separate the audio sources such that only the one or more sources assigned to a microphone are kept in the audio signal captured by a microphone, where the assignment is performed according to a proximity-based assignment rule as described herein.

[0121] Other examples of ML model adapted for audio source separation may be used. These ML models may also be trained to separate the audio sources such that only the one or more sources assigned to a microphone are kept in the audio signal captured by a microphone, where the assignment is performed according to a proximity-based assignment rule as described herein.

[0122] FIG. 8 is a flowchart illustrating a method for obtaining a trained ML model adapted for audio sources separation according to an example.

[0123] The method may be implemented by a processing device comprising means for performing the steps of the method. The means may for example include at least one processor and at least one memory storing instructions that, when executed by the at least one processor, cause the processing device to perform one or more or all steps of the method disclosed herein. The means may for example include circuitry (e.g., processing circuitry) configured to perform one or more or all steps of the method.

[0124] Any example or embodiment of ML model disclosed herein may be used.

[0125] In step 810, a training dataset is obtained.

[0126] In step 820, a ML model is trained with the training dataset.

[0127] In step 830, the trained ML model is outputted.

[0128] The training dataset comprises at least one set of training audio signals. Each set of training audio signals corresponds to a scenario according to which N training audio sources and M training microphones have respective spatial positions in a training spatial area, N being an integer equal or greater than 1 and M being an integer equal or greater than 2.

[0129] For each scenario and the associated set of training audio signals, the ML model is trained to generate, for each or at least one considered training microphone, an associated processed audio signal.

[0130] The ML model is trained (step 820) such that

the associated processed audio signal comprises the source audio content emitted at the spatial position of any training audio source assigned to the considered training microphone, if any, and separated from the other source audio content(s).

[0131] The assignment of training audio sources to training microphones is performed according to the proximity-based assignment rule such that a training audio source assigned to a considered training microphone is a training audio source for which the considered training microphone is the closest training microphone in the considered scenario.

[0132] The associated processed audio signal may comprise, for each audio source, if any, that is not assigned to the considered training microphone, either silence or the audio content of the considered audio source at a reduced sound level. An audio content of an audio source at a reduced sound level may for example be an audio content at a very low sound level, i.e., below a threshold level.

[0133] A training audio signal in a set of training audio signals comprises audio content received at the spatial position of an associated training microphone when source audio contents are emitted respectively at the spatial positions of the N training audio sources.

[0134] The training audio signals provided as input to the ML model for training may include (i) real audio signals captured by the training microphones in a training spatial area including N audio sources and/or (ii) simulated audio signals.

[0135] When simulated, the training audio signals used as input for the ML model are simulated as being those captured by the microphones. Each training audio signal contains the audio content emitted at the position of the closest source(s) (which are of interest) and interfering audio content (which need to be removed).

[0136] At least one set of training audio signals of the training dataset may be labelled by assigning each of the training audio sources to the closest training microphone. When acoustic simulation is used for generating the sets of training audio signals, the assignment may be determined as a function of the spatial positions selected for the training microphones and / or trainings audio sources.

[0137] For example, at least one training audio signal of at least one set of training audio signals may be a simulated training audio signal acoustically simulated based on:

- the source audio contents emitted respectively at the spatial positions of the N training audio sources; and
- the relative spatial position between the N training audio sources and the training microphone associated to the simulated training audio signal.

[0138] A simulated training audio signal may for ex-

ample be acoustically simulated based at least on one of or a combination of factors such as:

- audio interferences between the emitted audio contents,
- a room impulse response (RIR) associated with the training spatial area, where the room impulse response may take into account reverberation effects on the audio content of the audio sources depending on the spatial configuration of the training spatial area,
- a propagation attenuation and/or time-delay in arrival of sound between a position of a training audio source and a position of the associated training microphone,
- at least one property of the associated training microphone.

[0139] The property of a microphone may for example be one among a transducer type, a directionality, a frequency response, a sensitivity, an impedance, a self-noise, a signal-to-noise ratio, a maximum sound pressure level, etc.

[0140] Using factors like room impulse response, propagation attenuation, and transducer properties ensures that the simulated signals are realistic and representative of real-world conditions. The ML model can be trained to handle various configurations and positions of microphones and audio sources. Therefore, the ML model is robust to various scenarios and potential domain shifts.

[0141] For example, for simulating a concert hall, factors such as room impulse response, propagation attenuation, and microphone properties can be used to generate realistic training signals for instruments and vocals.

[0142] For example, for simulating a factory environment, factors such as room impulse response, propagation attenuation, and microphone properties can be used to generate realistic training signals for machinery and ambient noise.

[0143] For RIR generation, a room, corresponding to a training spatial area, with random dimensions may be created from a uniform distribution. The minimum and maximum values for x, y, and z dimensions may be selected. Next, M microphones may be placed in the room. For example, M = 4. For each microphone, sources may be randomly positioned in the training spatial area. This may be performed such that the distance between the microphone and speaker is less than two meters. Live audio engineers generally follow the three-to-one ratio, according to which *"the distance between any two microphones must be at least three times as great as the distance between the microphone and the source of sound which it is to pick up"*. We set this ratio to 2:1 instead of 3:1 to make the neural network robust to more challenging scenarios.

[0144] For the training of a ML model, 100, 000 RIRs

may be generated, among which 80, 000 may be used for training, 10, 000 for validation and 10, 000 for testing. For M microphones and N sources, each microphone m has a multi-channel RIR $\mathbb{R}_m^{n \times t}$ associated with it, with t being the time duration of the RIR. Therefore, to obtain the training audio signal for each microphone m , the multi-channel RIR $\mathbb{R}_m^{n \times t}$ is convolved with all the audio sources considered in the training spatial area.

[0145] Data augmentation may be applied to the training audio signals. Data augmentation can be based on random gain, random biquad filtering, random DC offset, random clipping, random resampling, etc.

[0146] For at least one scenario used for the training (step 820) of the ML model, the associated set of training audio signals may further comprise, for the at least one considered training microphone, an associated target audio signal (i.e., ground truth audio signal). The target audio signals may be generated from the audio content of the N audio sources used in the considered scenario.

[0147] The target audio signals (i.e., ground truth audio signals) for each microphone m include n_m signals containing only the audio content of the training audio sources (which are of interest) assigned to the considered training microphone according to the proximity-based assignment rule. In each target (i.e. ground truth) audio signal, the audio content of n_m training audio source is offset by a time delay in arrival of sound due to propagation from the source to the microphone. This delay is typically a few milliseconds.

[0148] The target audio signal associated with a considered training microphone may include, for each training audio source, if any, that is not assigned to the considered training microphone, either silence or the audio content of the considered audio source at a reduced sound level (e.g., at a very low sound level, below a threshold).

[0149] For example, in a scenario corresponding to a live concert, a performer has a microphone for their vocals and a microphone for their guitar. The target audio signal for the vocal microphone has reduced level of guitar signal and the target audio signal for guitar microphone has reduced level of vocal signal.

[0150] For example, in a scenario with multiple speakers, the target audio signal for the associated microphone includes the voice of the closest speaker with the background participant's voice at a reduced level.

[0151] In a spatial area, microphones may be placed close to audio sources to pick up the audio emitted by the audio sources. However, they typically pick up audio from neighboring audio sources, which is undesirable. This contaminates the microphone signals, creating challenges during audio mixing, and worsens the overall audio output. The ML model is trained to correctly clean the audio signal captured by the micro-

phones by only enhancing the desirable audio source (which are the closest sources to each respective microphone). Instead of completely removing the undesirable audio sources (that is silence), the audio content of audio sources not assigned to a microphone can be at a reduced sound level. There are a couple of advantages of this approach: (1) the processed signals sound more "natural" because they still have a sense of connection with the neighboring audio sources; (2) If the interfering audio sources has very high sound level, it is easier for the ML model to reduce the volume of the interfering audio source, instead of complete removal. This also ensures that the desirable audio source does not sound choppy or attenuated.

[0152] A target audio signal may be a dereverberated audio signal when at least one set of training audio signals is generated based on a room impulse response associated with the training spatial area. Training the ML model with such dereverberated audio signals improves the quality of the processed audio signals by minimizing reverberation effects. It also enhances the clarity of the separated audio signals. This may be useful for example in a reverberant concert hall where audio content for each instrument is captured by a microphone so as to generate processed audio signal with reduced reverberation effects for each instrument. This may be useful for example in a smart home to generate processed audio signal with reduced reverberation effects for household sounds.

[0153] The target audio signals may be generated from the audio contents of the training audio sources and so as to comply with the proximity-based assignment rule. Each target audio signal associated with a considered training microphone comprises the audio content emitted at the spatial position of each training audio source assigned to the considered training microphone and delayed by a time delay (e.g., based on a time of flight) and/or propagation attenuation between the relative spatial position of the considered assigned training audio source and the considered training microphone.

[0154] The time delays may be adjusted based on the distance between audio sources and microphones. The audio signal from an audio source arrives first at the closest microphone and arrives later at other microphones. The time-delay for the closest microphone is always lower than the other microphones. This way, the ML model is trained to separate audio sources only based on proximity between audio source and microphones, but no other audio features such as timbre or distance-threshold. This also ensure that the training audio signals and the target audio signals are time-aligned, and thereby helps in a better estimation of the loss function.

[0155] The target audio signal associated with a considered training microphone may be used as ground truth for the processed audio signal generated by the ML model for the considered training microphone.

[0156] A maximum number of sources assigned to a microphone may be set. For each microphone m , there may be a maximum of N_m^{max} nearest sources assigned to this microphone. The maximum number may be set for example to $N_m^{max} = 3$. Therefore, for the microphone m , there are either 0, 1, 2, or 3 audio sources assigned to this microphone. The maximum number of sources for a microphone may be configured as a function of the microphone. The maximum number of sources may be predefined or be configurable (e.g. by a user).

[0157] To generate the target (i.e. ground truth) audio signal for a microphone m , the n_m nearest sources are convolved with the direct path of sound for $\mathbb{R}_m^{n \times t}$. In other words, the target (i.e. ground truth) audio signals only contain audio from the n_m nearest sources, with the time-delays equal to the time of arrival of the direct sounds of the n_m nearest sources.

[0158] The method may comprise, during the training (step 820) of the ML model, a step of comparing, for each or the at least one considered training microphone, the associated processed audio signal and the associated target audio signal to generate a comparison result relative to the at least one scenario.

[0159] During the training of the ML model (step 820), a training loss may be computed based on several comparison results generated respectively for one or more scenarios. The ML model (e.g., its weights) may be updated on the basis of the training loss.

[0160] Various training loss functions may be used. For example, the training loss function may be one among or a combination of the following training loss functions: mean absolute error and/or mean square error such as mean absolute error and/or mean square error on a waveform and/or a spectrogram.

[0161] One or more temporal portions of the processed audio signals PS1 to PS4 may be compared with corresponding temporal portions of the target audio by a loss function. Examples of loss functions include:

- a mean absolute error on the waveform to update model's weights;
- a mean absolute error on the waveform and the spectrogram to update model's weights;
- a mean squared error on the spectrogram to update model's weights.

[0162] The loss function is used to update the weights of the ML model. For each scenario, a higher error indicates poorer performance, and a lower error indicates good performance. The ML model may be repeatedly updated to enhance its performance. A spectrogram-based loss will be useful for harmonic sources such as vocals and a waveform-based loss will be useful for rhythmic and percussive sounds. Furthermore, the loss function can be used to quantify or eval-

uate the ML model's ability to separate audio sources.

[0163] The scenarios that are used for training the ML model (step 820) may vary to be able to train the ML model for various possible configurations of the training audio sources and the training microphones.

[0164] Training the ML model with various scenarios ensures that the model can handle different configurations and environments. The model becomes versatile and adaptable to a wide range of real-world situations.

[0165] For example, over the scenarios that are used for training the ML model:

- a geometry of the training spatial area may vary (e.g., randomly); and/or
- the spatial position of at least one training microphone may vary (e.g., randomly); and/or
- the spatial position of at least one training audio source may vary (e.g., randomly).

[0166] Further, over the scenarios that are used for training the ML model:

- the number N of training audio sources may vary (e.g., randomly); and/or
- the number M of training microphones may vary (e.g., randomly); and/or
- at least one characteristic of the audio content emitted at a spatial position of at least one training audio source varies (e.g., randomly).

[0167] The characteristic may be in a set of characteristics comprising: a volume of the audio content, a class of the audio content, a frequency content of the audio content, a sample rate of the audio content.

[0168] The ML model may be trained (step 820) to be a specialized trained ML model. For example, the training dataset may comprise at least one specific set of training audio signals comprising only audio contents of at least one specific class of audio sources, such that the ML model is specifically trained for this class of audio sources and is a specialized trained ML model obtained by training the ML model with only the at least one specific set of training audio signals.

[0169] For example, in a live band setup, specific classes may be of interest such as voice and musical instruments. Therefore, the proximity-based source separation is optimized for the audio classes of interest. Other undesirable audio classes such as fan noise, projector noise, clapping, traffic, etc., do not appear in any of the microphone signals and are automatically removed irrespective of their location in the spatial area.

[0170] In embodiments, a specialized architecture of the ML model can be used for specific audio classes. One ML architecture maybe better than another for specific audio classes. For example, TasNet and vocoder-based generative models are good for speech separation and HS-TasNet is good for music source se-

paration. Similarly specialized models can be used for acoustic machine monitoring (only machine sounds), wildlife monitoring (only animal sounds), panel discussions (only voice), etc.

[0171] FIG. 9 is a flowchart illustrating a method for separating audio sources during an inference phase according to an example embodiment.

[0172] The method may be implemented by a processing device comprising means for performing the steps of the method. The means may for example include at least one processor and at least one memory storing instructions that, when executed by the at least one processor, cause the processing device to perform one or more or all steps of the method disclosed herein. The means may for example include circuitry (e.g., processing circuitry) configured to perform one or more or all steps of the method.

[0173] In step 910, a trained ML model is obtained. The trained model is adapted for audio sources separation. The trained model may be generated using the method disclosed by reference to FIG. 8 and/or any method disclosed herein.

[0174] In step 920, a set of audio signals are obtained, the audio signals being audio signals captured respectively by a plurality of microphones in a spatial area including audio sources configured to generate respective source audio contents.

[0175] In step 930, the set of audio signals is processed with the trained ML model, to generate, for each or at least one considered microphone among the plurality of microphones, an associated processed audio signal.

[0176] The associated processed audio signal comprises the source audio content emitted by any audio source assigned to the considered microphone if any and separated from the other source audio content(s), if any.

[0177] The assignment of audio sources to microphones is performed according to the proximity-based assignment rule such that an audio source assigned to the considered microphone is an audio source for which the considered microphone is the closest microphone.

[0178] The associated processed audio signal comprises, for each audio source, if any, that is not assigned to the considered microphone, either silence or the audio content of the considered audio source at a reduced sound level.

[0179] The steps 920 and 930 may be repeated over time. For example, the processing of the set of audio signals during the inference phase may be carried out in real time while the relative spatial position of at least one audio source varies in the spatial area with respect to the microphones.

[0180] The audio signals are processed in real-time while the spatial positions of audio sources vary. This provides instant audio separation based on the current location of audio sources and adapts to position

changes in real-time.

[0181] For example, during a live concert, a performer is singing into a microphone M1. A few minutes later, he walks to another microphone M2 and sings into it. In this case, the closest microphone changed from M1 to M2 during the concert. In such a case, the ML model automatically adapts due to the proximity-based assignment rule by keeping the singer's voice in M1 for the first part and in M2 for the second part.

[0182] In embodiments, a post-processing may be applied to one or more or each of processed audio signals generated by the trained ML model.

[0183] The post-processing may include audio filters and/or dynamic range compression and/or intelligent mixing and/or adaptive filters and/or denoising and/or dereverberation. Denoising a processed audio signal may comprise suppressing audio content relative to at least one undesirable class of audio sources from the processed audio signal. The undesirable class may be selected or identified by a user or be provided as an input parameter.

[0184] Suppressing unwanted noise improves the quality of the separated audio content. For example, to capture only voice signals in a live concert, a specialized model may be trained to remove instrumental sounds such as drums, cymbals, guitar, and audience noise.

[0185] In embodiments the ML model may be trained to perform denoising and/or dereverberation. This can be achieved by training the model with post-processed target audio signals. The post-processed target audio signals are target audio signals as disclosed herein to which post-processing (denoising and/or the dereverberation) has been applied.

[0186] In embodiments, metadata may be generated for the processed audio signals generated by the trained ML model. The metadata may for example include:

- the number of audio sources kept at the output of the ML model in each or at least one of the processed audio signals and/or
- an identification of one or more classes of audio sources present in each or at least one of the processed audio signals, and/or
- a list of one or more audio sources assigned to a given microphone, etc.

[0187] This metadata may be outputted as information for a user and/or for being used in a subsequent processing step of the processed audio signal. The metadata allow a user to check and/or understand and/or better use the outputted processed audio signals for further processing.

[0188] For example, the metadata helps users understand and utilize the separated audio signals. This metadata can be used for further processing and analysis. For example, outputting the audio class for a processed

signal can help a mixing engineer apply class-specific audio effects (e.g., different equalization settings for vocals and drums). This can also be passed to an automatic (or intelligent) mixing system with class-specific mixing parameters. Furthermore, outputting the number of sources in each processed signal can be used as metadata to be passed into another source separation algorithm or processing algorithm.

[0189] It should be appreciated by those skilled in the art that any functions, engines, block diagrams, flow diagrams, state transition diagrams, flowchart and / or data structures described herein represent conceptual views of illustrative circuitry embodying the principles of the invention. Similarly, it will be appreciated that any flow charts, flow diagrams, state transition diagrams, pseudo code, and the like represent various processes.

[0190] Although steps of a method or process may be described in a sequential manner, some of the steps may be performed in parallel, concurrently or simultaneously. Also some steps may be omitted, combined or performed in different order.

[0191] One or more or all operation(s) of a method, process, function, engine, block, step described herein may be implemented in hardware, software, firmware, middleware, microcode, or any suitable combination thereof.

[0192] When implemented in software, firmware, middleware or microcode, instructions to perform the considered operation(s) may be stored in a computer readable medium that may be or not included in a processing device configured to execute the instructions. The instructions may be stored in the computer-readable medium and be loaded onto the processing device. The instructions, when executed by at least one processor, are configured to cause the processing device to perform the considered operation(s).

[0193] In embodiments, the processing device may include at least one processor and at least one memory storing the instructions that, when executed by the at least one processor, cause the processing device to perform the considered operation(s). The instructions may correspond to program instructions or a computer program code.

[0194] FIG. 10 illustrates an example embodiment of a processing device 9000.

[0195] The processing device 9000 may be configured to perform one or more or all steps of any method disclosed herein. The processing device 9000 is an example of processing device comprising means for performing one or more or all steps of any method disclosed herein. The processing device 9000 may be a computing device or an audio signal processing device, for example dedicated to audio signal processing.

[0196] As represented schematically, the processing device 9000 may include at least one processor 9010 and at least one memory 9020.

[0197] The memory 9020 may be or include a random-access memory (RAM), cache memory, non-vola-

tile memory, backup memory (e.g., programmable or flash memories), read-only memory (ROM), a hard disk drive (HDD), a solid-state drive (SSD) or any combination thereof. The ROM of the memory 9020 may be configured to store, amongst other things, an operating system of the processing device 9000 and / or one or more computer program code of one or more software applications. The RAM of the memory 9020 may be used by the processor 9010 for the temporary storage of data.

[0198] The processor 9010 may be configured to store, read, load, execute and/or otherwise process instructions 9070 stored in a computer-readable storage medium 9060, 9080 and / or in the memory 9020 such that, when the instructions are executed by the processor, causes the processing device 9000 to perform one or more or all steps of a method described herein.

[0199] The term "processor" should not be construed to refer exclusively to hardware capable of executing software and may implicitly include one or more processing circuits, whether programmable or not. A processor or a processing circuit may correspond to a digital signal processor (DSP), a network processor, an application specific integrated circuit (ASIC), a field programmable gate array (FPGA), a System-on-Chips (SoC), a Central Processing Unit (CPU), a Graphical Processing Unit (GPU), an arithmetic logic unit (ALU), a programmable logic unit (PLU), a processing core, a programmable logic, a microprocessor, a microcontroller, a microcomputer, a quantum processor, any circuitry capable of responding to and/or executing instructions in a defined manner and/or according to a logic.

[0200] The processor 9010 may be connected to each of other components 9020, 9030, 9040, 9050 in order to control operation thereof.

[0201] The processing device 9000 may include one or more communication interfaces 9040 connected to the processor 9010. The communication interfaces 9040 may for example include at least one transceiver for receiving and transmitting data. The communication interfaces 9040 may include interfaces for connecting peripherals (e.g., USB interfaces, HDMI interfaces, etc) and/or interfaces for accessing to a wired / wireless network (e.g., Ethernet interface, WIFI interface, Bluetooth interface, etc).

[0202] The processing device 9000 may include user interface devices 9030 (e.g., keyboard, mouse, display screen) connected with the processor 9010. The processing device 9000 may further include one or more media drives 9050 for reading a computer-readable storage medium (e.g., digital storage disc 9060 (CD-ROM, DVD, Blue Ray), USB key 9080). The processor 9010 may be connected to each of the other components 9020, 9030, 9040, 9050 in order to control operation thereof.

[0203] A computer readable medium or computer readable storage medium may be any tangible storage medium suitable for storing instructions readable by a

computer or a processor. A computer readable medium may be more generally any storage medium capable of storing and/or containing and/or carrying instructions and/or data. The computer readable medium may be a non-transitory computer readable medium. A computer readable medium may include one or more storage device like a permanent mass storage device, magnetic storage medium, optical storage medium, digital storage disc (CD-ROM, DVD, Blue Ray, etc), USB key or dongle or peripheral, a memory suitable for storing instructions readable by a computer or a processor.

[0204] A memory suitable for storing instructions readable by a computer or a processor may be for example: read only memory (ROM), a permanent mass storage device such as a disk drive, a hard disk drive (HDD), a solid-state drive (SSD), a memory card, a core memory, a flash memory, or any combination thereof.

[0205] In the present description, the wording "means for performing one or more functions" may correspond to one or more functional blocks comprising circuitry that is adapted for performing the concerned function(s). Performing a function may include carrying out one or more steps of a method disclosed herein. The functional block may perform itself the concerned function(s) or may cooperate and / or communicate with one or more other functional blocks to perform the concerned function(s). The means may include at least one processor and at least one memory storing instructions that, when executed by the at least one processor, cause a host device to perform one or more or all of the considered function(s). The means may include circuitry (e.g., processing circuitry) configured to perform one or more or all of the considered function(s).

[0206] The term "circuitry" (e.g., processing circuitry) may refer to hardware circuit implementations, with or without associated software and/or firmware. The term "circuitry" may refer to one or more circuits. A circuit may be a general-purpose circuit or a specific purpose circuit. A circuit may or not be a programmable circuit. A circuit may include one or more processors or one or more microcontrollers.

[0207] Although the terms first, second, etc., may be used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another. For example, a first element could be termed a second element, and similarly, a second element could be termed a first element, without departing from the scope of this disclosure. As used herein, when the term "and/or" is used in a list of items, it implies that the list may include any and all combinations of one or more of the associated listed items.

[0208] The terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting. As used herein, the singular forms "a," "an," and "the," are intended to include the plural forms as well, unless the context clearly indicates

otherwise. It will be further understood that the terms "comprises," "comprising," "includes," and/or "including," when used herein, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

[0209] It will be appreciated that, as used herein, "at least one of <a list of two or more elements>" and "at least one of the following: <a list of two or more elements>" and similar wording, where the list of two or more elements are joined by "and" or "or", mean at least any one of the elements, or at least any two or more of the elements, or at least all the elements.

[0210] Although aspects have been described with reference to particular embodiments, it is to be understood that these embodiments are merely illustrative of the principles and applications of the present disclosure. It is therefore to be understood that numerous modifications can be made to the illustrative embodiments and that other arrangements can be devised without departing from the spirit and scope of the disclosure as determined based upon the claims and any equivalents thereof.

Claims

1. A method for obtaining a trained machine learning, ML, model (110) adapted for audio sources separation, the method comprising:

- obtaining (810) a training dataset comprising at least one set of training audio signals (MS1, MS2, MS3, MS4), each set of training audio signals corresponding to a scenario according to which N training audio sources and M training microphones (M1, M2, M3, M4) have respective spatial positions in a training spatial area, N being an integer equal or greater than 1 and M being an integer equal or greater than 2;

- wherein a training audio signal (MS1, MS2, MS3, MS4) in a set of training audio signals comprises audio content received at the spatial position of an associated training microphone (M1, M2, M3, M4) when source audio contents are emitted respectively at the spatial positions of the N training audio sources;

- training a ML model (110) with said training dataset (820), wherein for each scenario and associated set of training audio signals, said ML model is trained to generate, for each or at least one considered training microphone (M1, M2, M3, M4), an associated processed audio signal (PS1, PS2, PS3, PS4), said associated processed audio signal (PS1,

PS2, PS3, PS4) comprising:

o the source audio content emitted at the spatial position of any training audio source assigned to the considered training microphone (M1, M2, M3, M4), if any, and separated from the other source audio content(s), wherein a training audio source assigned to a considered training microphone is a training audio source for which the considered training microphone is the closest training microphone in the considered scenario;

- outputting the trained ML model (830).

2. The method according to claim 1, wherein said associated processed audio signal (PS1, PS2, PS3, PS4) comprises, for each audio source, if any, that is not assigned to the considered training microphone, either silence or the audio content of the considered audio source at a reduced sound level.

3. The method according to claim 1 or 2, wherein at least one training audio signal (MS1, MS2, MS3, MS4) of at least one set of training audio signals is a simulated training audio signal acoustically simulated based on:

- the source audio contents emitted respectively at the spatial positions of the N training audio sources; and
- the relative spatial position between the N training audio sources and the training microphone associated to the simulated training audio signal.

4. The method according to claim 3 wherein said at least one training audio signal is acoustically simulated based at least on one of or a combination of factors among:

- (i) audio interferences between the emitted audio contents,
- (ii) a room impulse response associated with the training spatial area,
- (iii) a propagation attenuation between a position of a training audio source and a position of a training microphone,
- (iv) a time delay in arrival of sound between a position of a training audio source and a position of a training microphone;
- (v) at least one property of the associated training microphone, wherein the at least one property is taken among a transducer type, a directionality, a frequency response, a sensitivity, an impedance, a self-noise, a signal-to-noise ratio, a maximum sound pressure level.

5. The method according to any one of claims 1 to 4, wherein, for at least one scenario, the associated set of training audio signals further comprises, for the at least one considered training microphone (M1, M2, M3, M4), an associated target audio signal (TS1, TS2, TS3, TS4), the method further comprising comparing, for each or the at least one considered training microphone (M1, M2, M3, M4), the associated processed audio signal (PS1, PS2, PS3, PS4) and the associated target audio signal (TS1, TS2, TS3, TS4) to generate a comparison result (R1, R2, R3, R4) relative to the at least one scenario.

6. The method according to claim 5, further comprising during the training of the ML model (820):

- computing a training loss function based on at least one comparison result generated respectively for at least one scenario;
- updating the ML model on the basis of the training loss function.

7. The method according to claim 5 or 6, wherein the target audio signal (TS1, TS2, TS3, TS4) associated with a given microphone (M1, M2, M3, M4) includes, for each audio source that is not assigned to the given microphone, either silence or the audio content of the considered audio source at a reduced sound level.

8. The method according to any one of claims 5 to 7, wherein at least one target audio signal (TS1, TS2, TS3, TS4) is a dereverberated audio signal when at least one set of training audio signals is generated based on a room impulse response associated with the training spatial area.

9. The method according to any one of claims 5 to 8, wherein each target audio signal (TS1, TS2, TS3, TS4) associated with a considered training microphone (M1, M2, M3, M4) comprises the audio content emitted at the spatial position of each training audio source assigned to the considered training microphone and delayed by a time delay based on a time of flight between the relative spatial position of the considered assigned training audio source and the considered training microphone.

10. The method according to any one of claims 1 to 9 wherein, over the scenarios:

- a geometry of the training spatial area varies; and/or
- the number N of training audio sources varies; and/or
- the spatial position of at least one training microphone varies; and/or

- the spatial position of at least one training audio source varies; and/or
 - the number M of training microphones varies; and/or
 - at least one characteristic of the audio content emitted at a spatial position of at least one training audio source varies, the at least one characteristic being in a set of characteristics comprising: a volume of the audio content, a class of the audio content, a frequency content of the audio content, a sample rate of the audio content.
11. A method for separating audio sources, the method comprising:
- obtaining a trained ML model (910) adapted for audio sources separation using the method according to any one of claims 1 to 11;
 - obtaining a set of audio signals captured respectively by a plurality of microphones in a spatial area including audio sources configured to generate respective source audio contents (920);
 - processing, with the trained ML model, the set of audio signals to generate, for each or at least one considered microphone among the plurality of microphones, an associated processed audio signal (930), said associated processed audio signal comprising:
 - o the source audio content emitted by any audio source assigned to the considered microphone if any and separated from the other source audio content(s), wherein an audio source assigned to the considered microphone is an audio source for which the considered microphone is the closest microphone.
12. The method according to claim 11, wherein said associated processed audio signal comprises, for each audio source, if any, that is not assigned to the considered microphone, either silence or the audio content of the considered audio source at a reduced sound level.
13. The separating method according to claim 11 or 12, wherein the processing of the set of audio signals is carried out in real time while the relative spatial position of at least one audio source varies in the spatial area.
14. The separating method according to any one of claims 11 to 13, comprising outputting at least one among: the number of audio sources in each set of audio signals, the different classes of audio sources present in the audio signals, a list of audio sources assigned to a given microphone.
15. A processing device comprising means for performing the steps of a method of any one of the preceding claims.

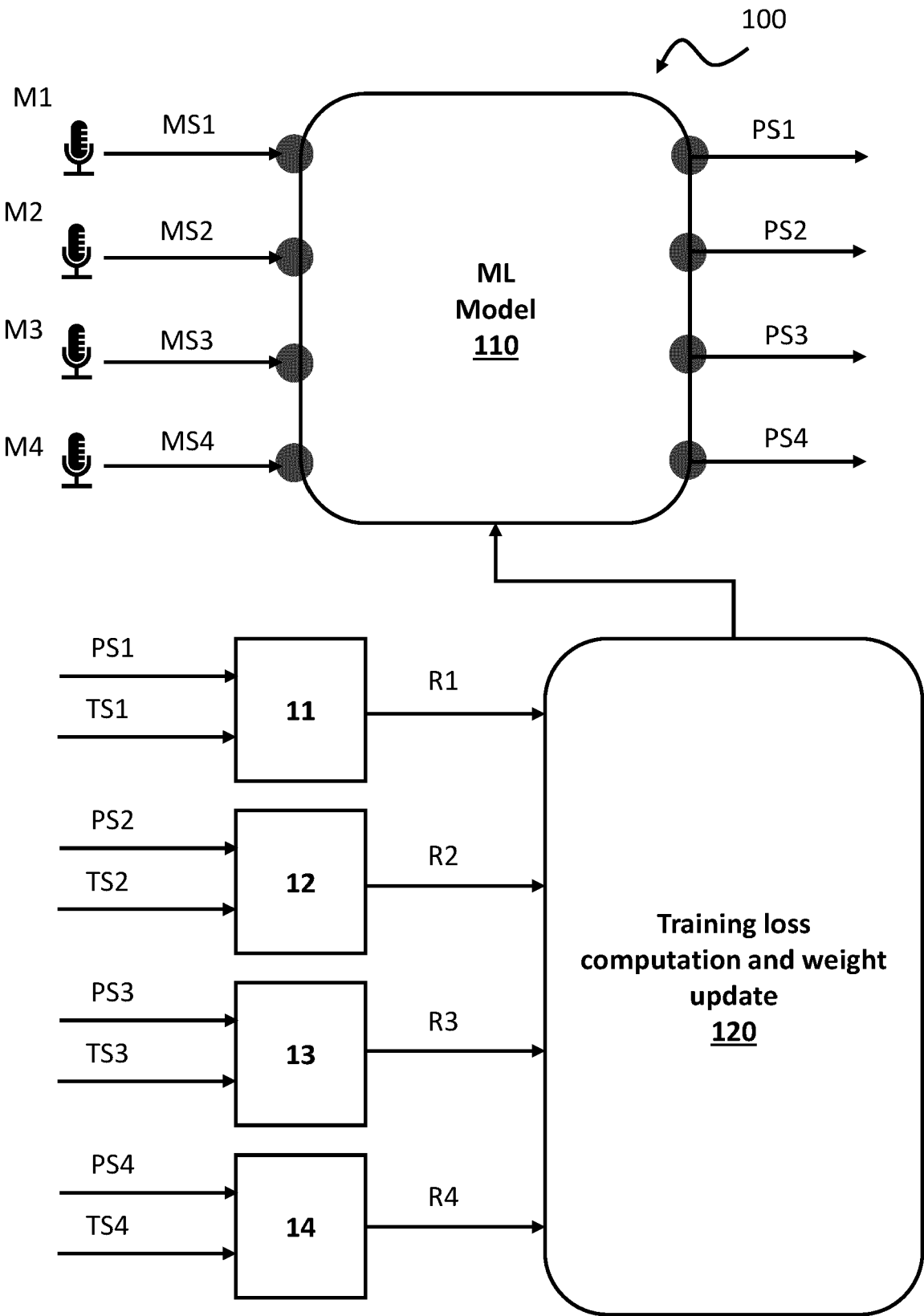


FIG. 1

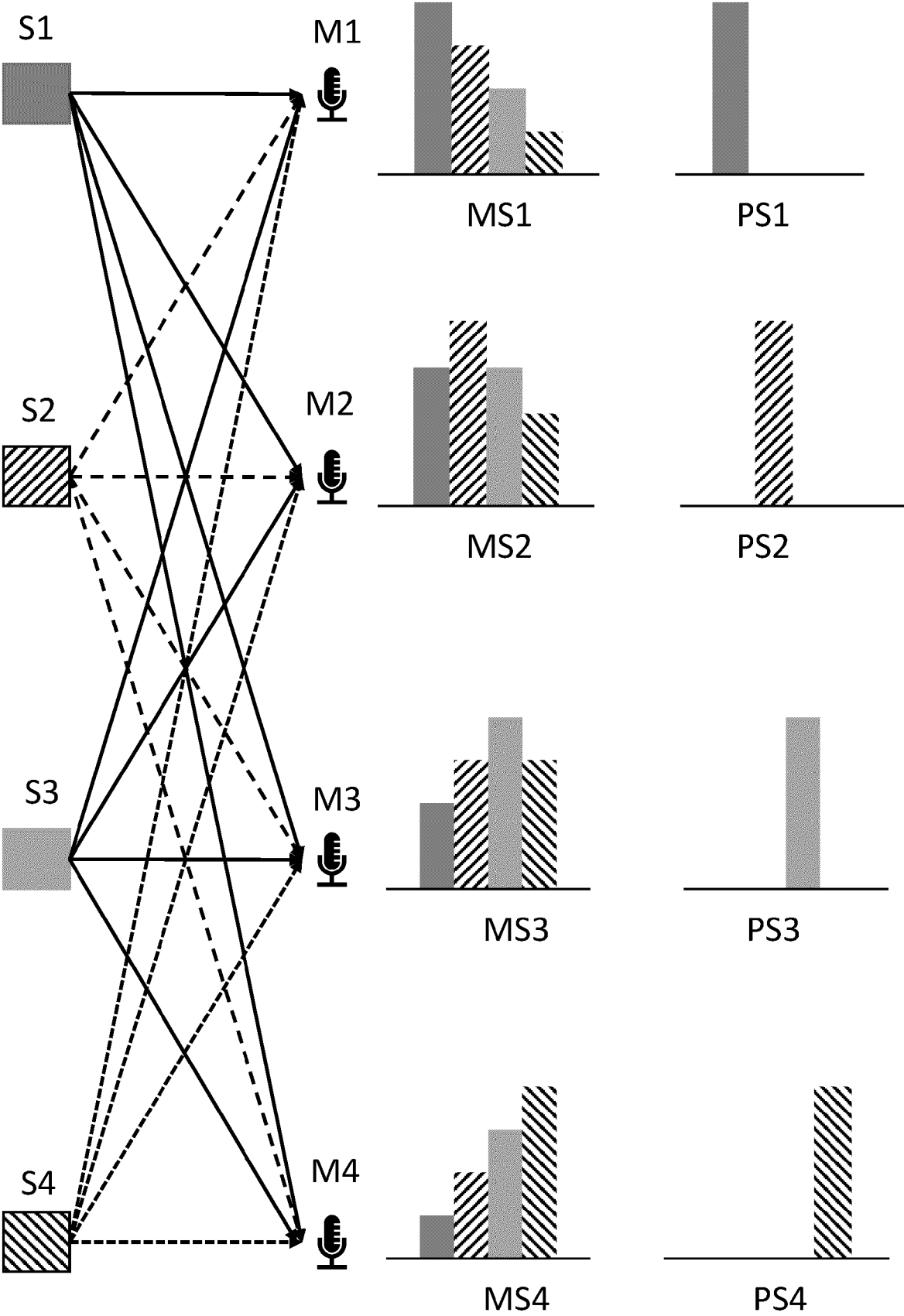
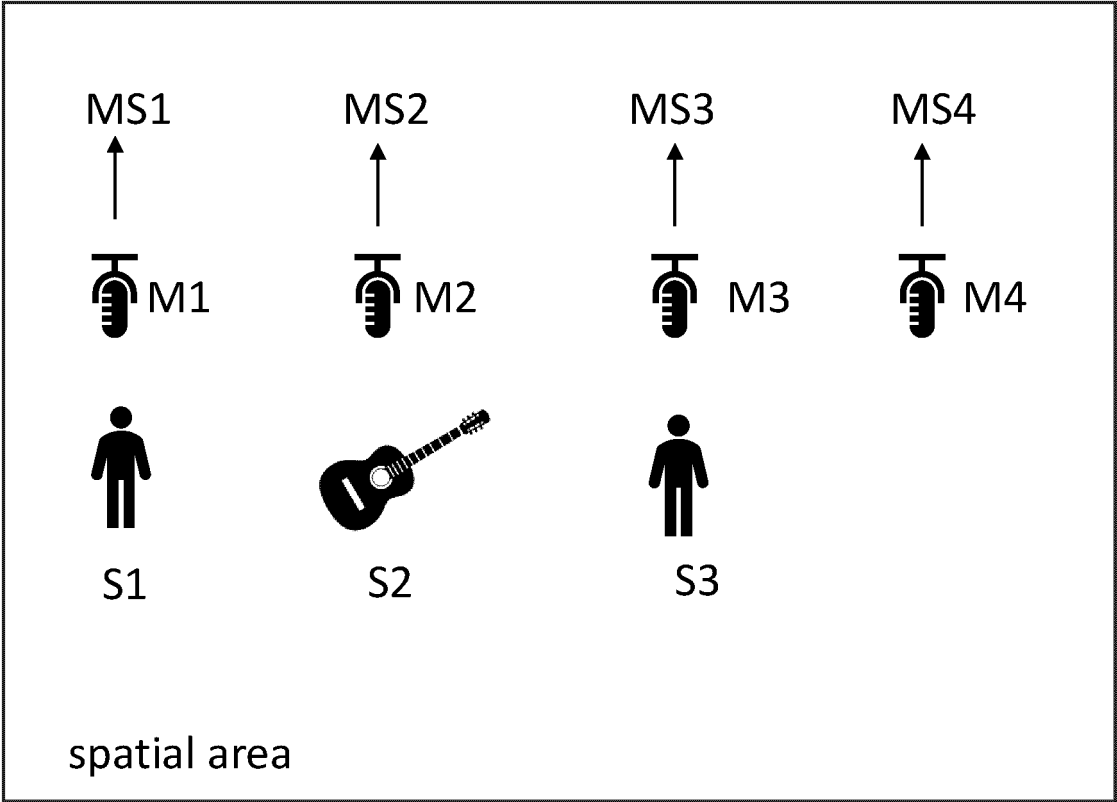


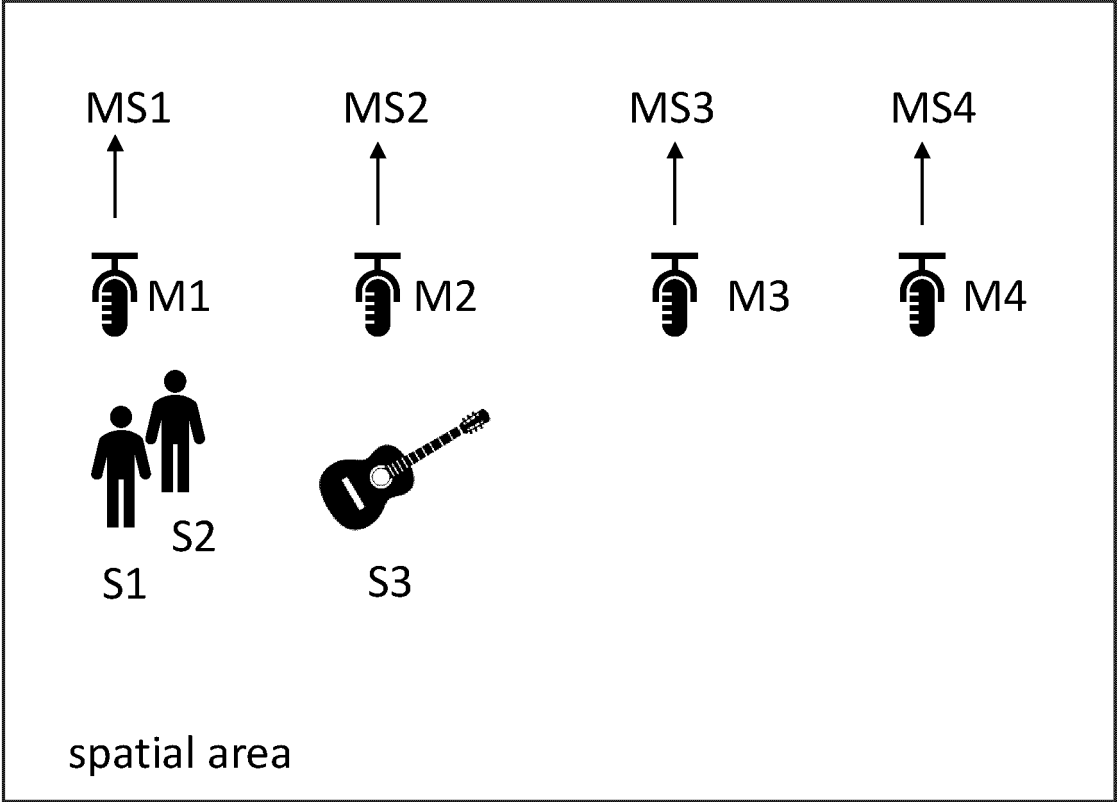
FIG. 2



Target or processed audio signals

TS1	TS2	TS3	TS4
S1	S2	S3	silence

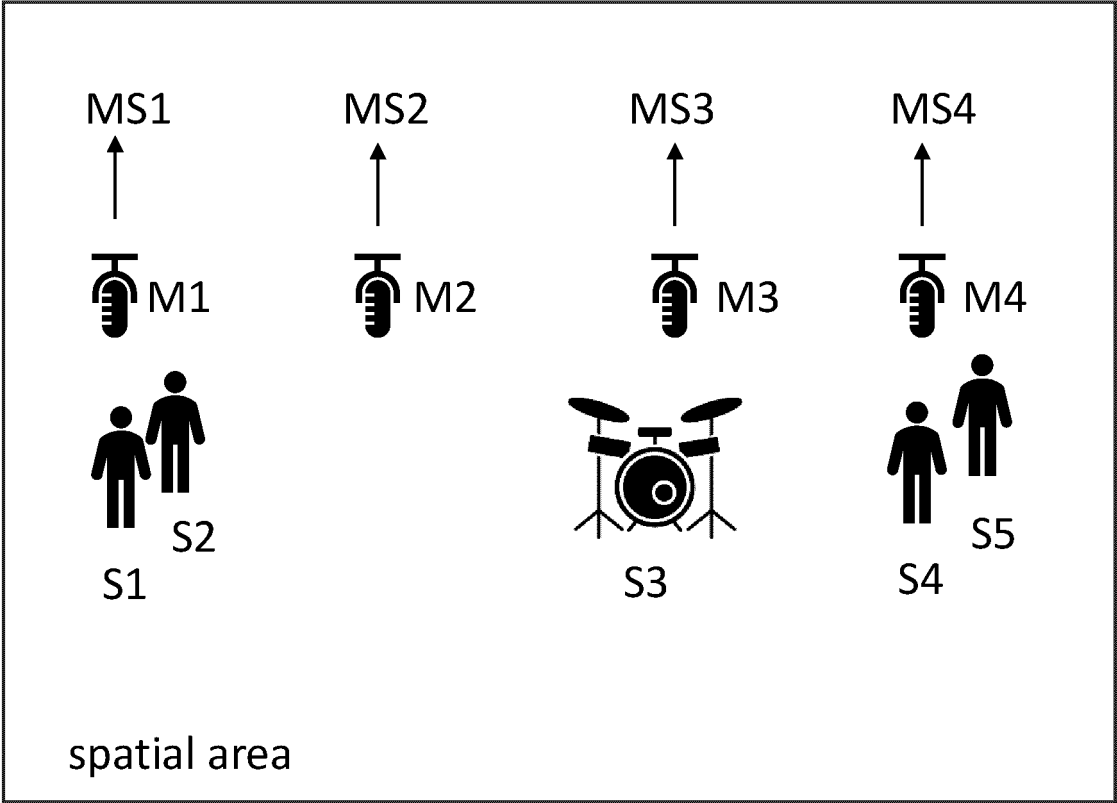
FIG. 3



Target or processed audio signals

TS1	TS2	TS3	TS4
S1+S2	S3	silence	silence

FIG. 4



Target or processed audio signals

TS1	TS2	TS3	TS4
S1+S2	silence	S3	S4+S5

FIG. 5

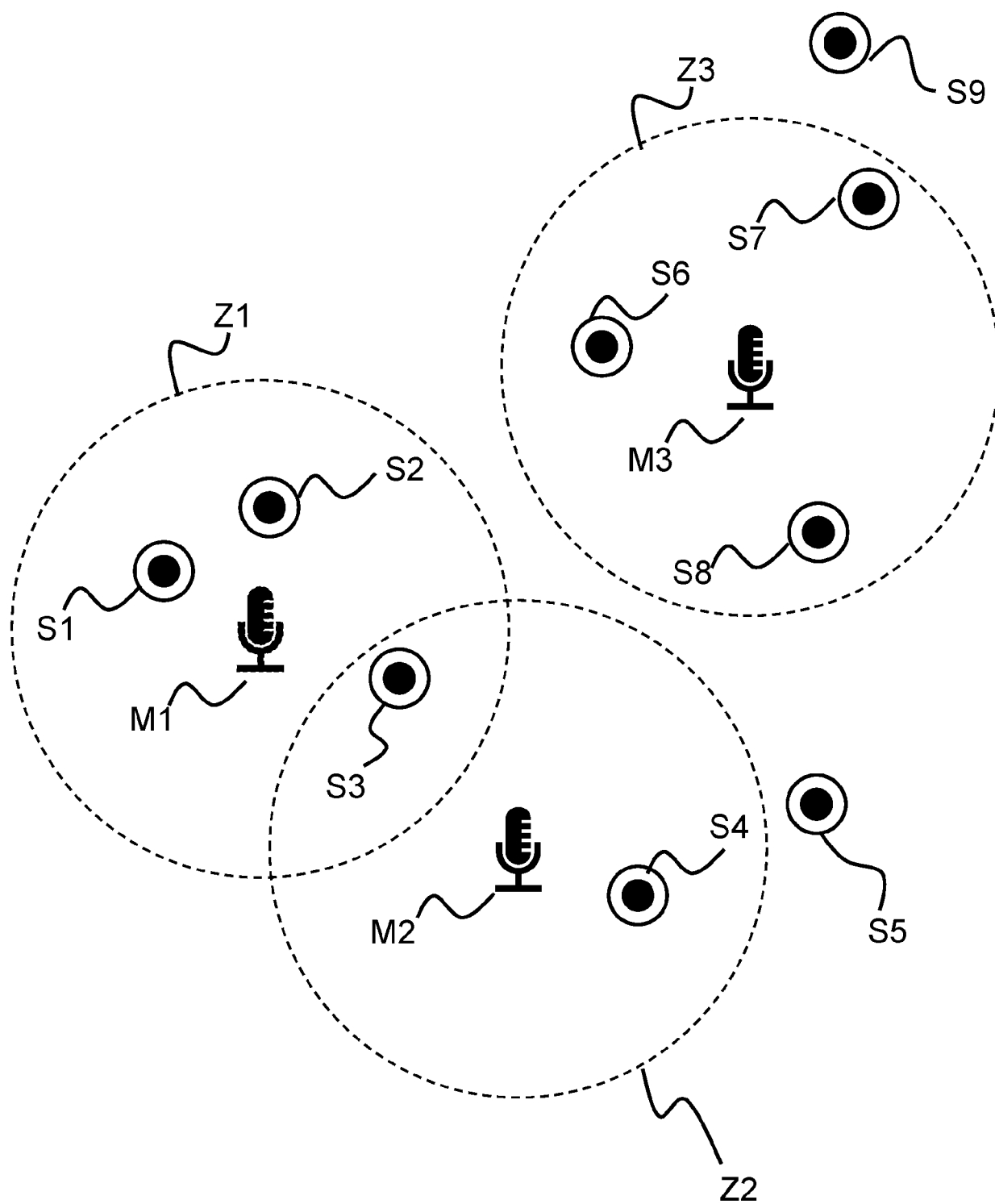


FIG. 6

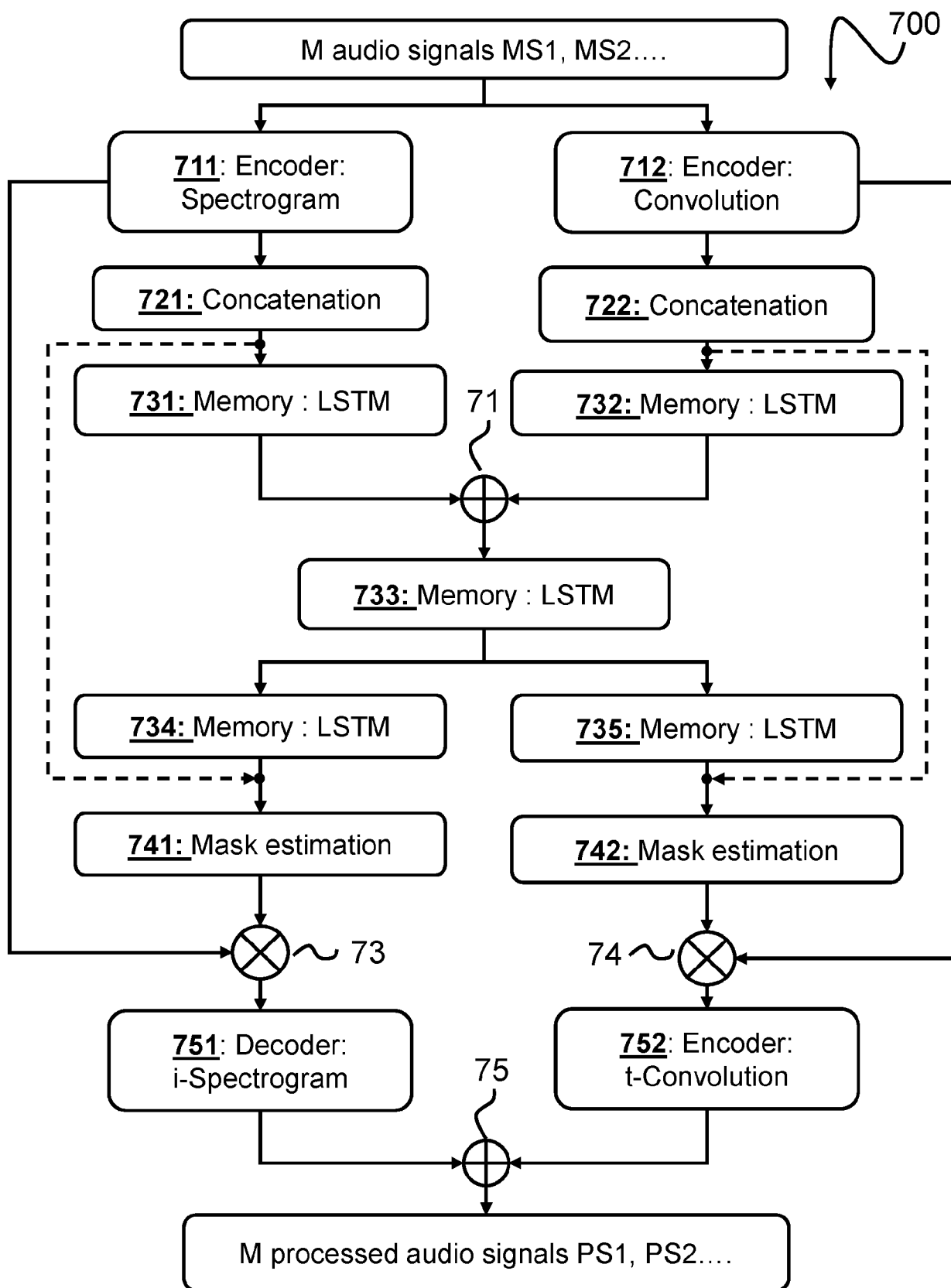


FIG. 7

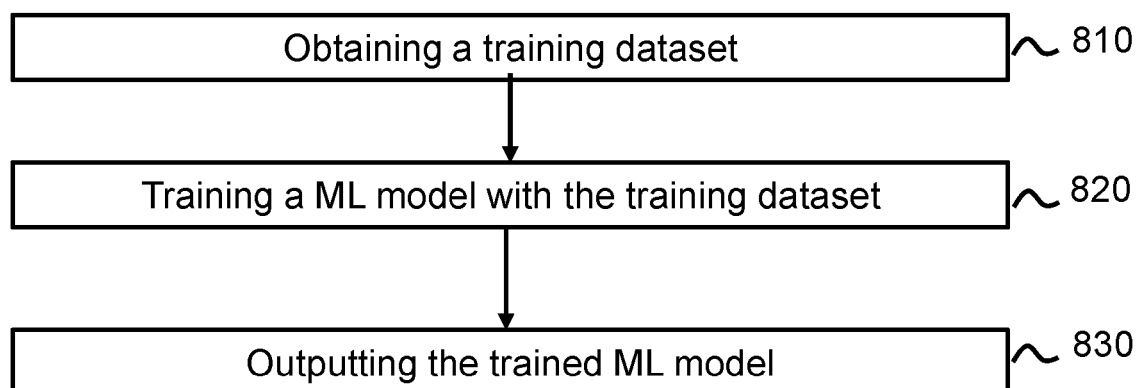


FIG. 8

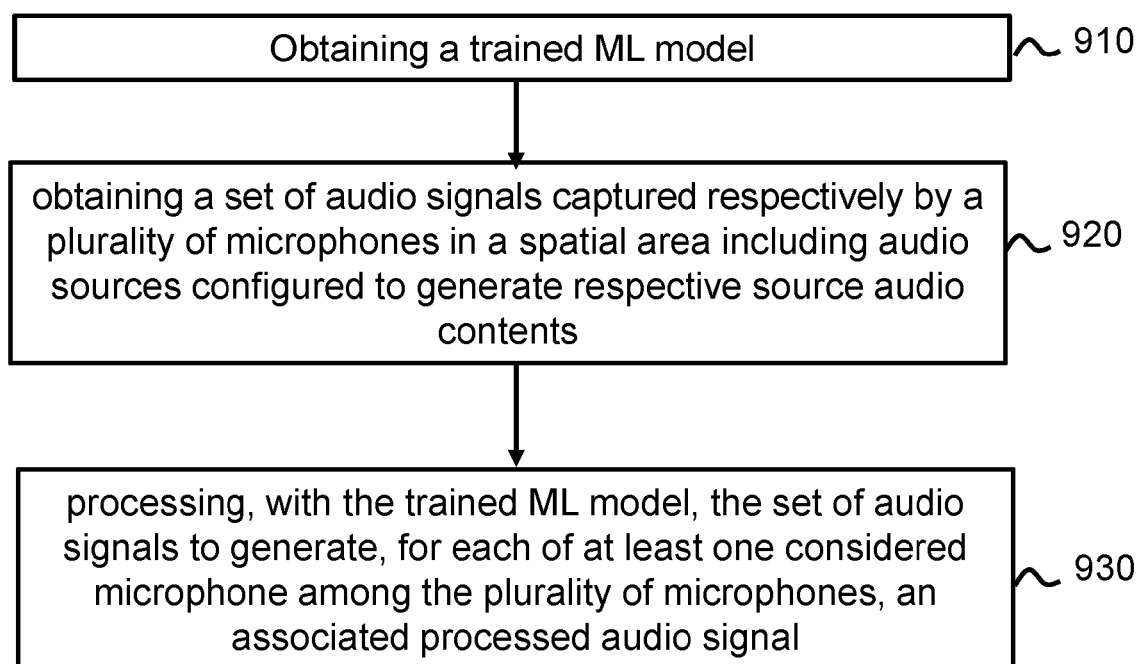


FIG. 9

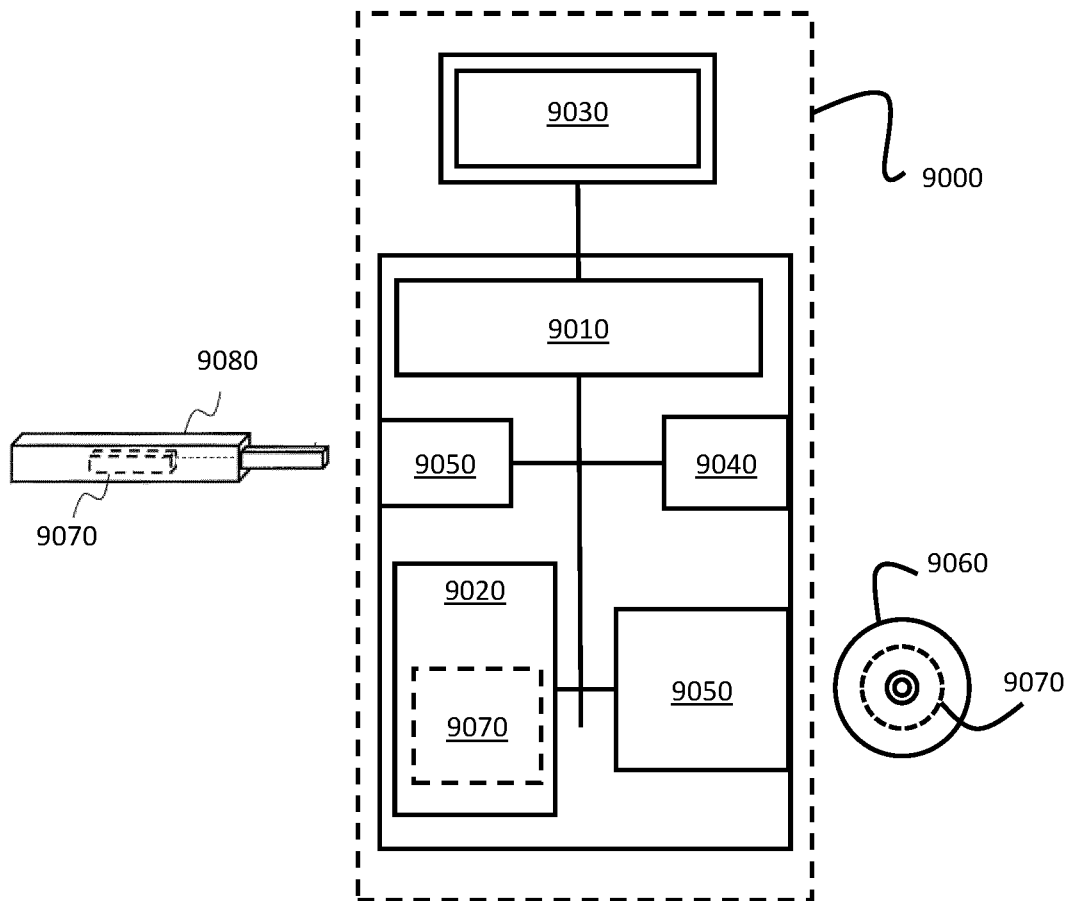


FIG. 10



EUROPEAN SEARCH REPORT

Application Number

EP 25 17 0686

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
A	RAJESH R ET AL: "Neural Networks for Interference Reduction in Multi-Track Recordings", 2023 IEEE WORKSHOP ON APPLICATIONS OF SIGNAL PROCESSING TO AUDIO AND ACOUSTICS (WASPAA), IEEE, 22 October 2023 (2023-10-22), pages 1-5, XP034423596, DOI: 10.1109/WASPAA58266.2023.10248133 [retrieved on 2023-09-15] * the whole document * -----	1-15	INV. G10L21/0272 ADD. G10L21/0208
A	TAHERIAN HASSAN ET AL: "Multi-Channel Talker-Independent Speaker Separation Through Location-Based Training", ARXIV:1806.04885V2,, vol. 30, 25 August 2022 (2022-08-25), pages 2791-2800, XP011919095, DOI: 10.1109/TASLP.2022.3202129 [retrieved on 2022-08-26] * the whole document * -----	1-15	TECHNICAL FIELDS SEARCHED (IPC)
A	US 2016/358107 A1 (KOKKINIS ELIAS [GR] ET AL) 8 December 2016 (2016-12-08) * the whole document * -----	1-15	G10L
The present search report has been drawn up for all claims			
Place of search		Date of completion of the search	Examiner
The Hague		21 August 2025	De Meuleneire, M
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			

EPO FORM 1503 03.82 (P24C01)

21-08-2025

EPO FORM P0459

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