METHOD AND APPARATUS FOR USING INCREMENTAL SEARCH SEQUENCE IN AUDIO ERROR CONCEALMENT

TECHNICAL FIELD

[0001] Various aspects of this disclosure generally relate to signal processing, and more particularly, to audio error concealment.

BACKGROUND

[0002] In a digital wireless audio application, namely wireless speaker or wireless headset, audio data is packetized and sent from a transmitter to a receiver in burst. Retry mechanism is implemented to improve the success rate of audio data transmission in such a digital wireless audio application. However, audio data packet (may also be referred to as audio frame) may be dropped if it fails to be transmitted after a specific number of retry. Listener may hear a popping or cracking noise when there is an audio signal gap due to packet lost (i.e., dropped packet).

[0003] FIG. 1 illustrates an example of an audio signal gap caused by packet lost. In the example, the original audio data transmitted by the transmitter is shown in diagram 100. The audio packets received by the receiver is shown in diagram 120. A packet 122 is lost during transmission. Such packet loss may cause popping or cracking noise, thus detrimentally affecting user experience.

SUMMARY

[0004] The following presents a simplified summary of one or more aspects in order to provide a basic understanding of such aspects. This summary is not an extensive overview of all contemplated aspects, and is intended to neither identify key or critical elements of all aspects nor delineate the scope of any or all aspects. Its sole purpose is to present some concepts of one or more aspects in a simplified form as a prelude to the more detailed description that is presented later.

[0005] In wireless audio streaming application, audio data has to reach the receiver in a given time period. However, the audio data might be interfered and not received by the

receiver. In that case, there will be an audio gap during receiver playback. In one aspect of this disclosure, a method to search for similar audio waveform from past history to reconstruct the missing audio data is provided. It may preserve the continuity of the audio waveform, resulting in better audio playback quality, less popping and cracking noise.

[0006] In an aspect of the disclosure, a method, a computer readable medium, and an apparatus for audio error concealment are provided. The apparatus may receive a plurality of audio frames. The apparatus may receive a first audio frame after the receiving of the plurality of audio frames. The apparatus may detect a second audio frame being lost in transmission. The second audio frame is transmitted after the plurality of audio frames and before the first audio frame. The apparatus may identify an audio segment within the plurality of audio frames that best matches a reference audio pattern of the plurality of audio frames. The identified audio segment may be received before the last audio frame of the plurality of audio frames. The apparatus may reconstruct the second audio frame based on audio data received subsequent to the identified audio segment.

[0007] To the accomplishment of the foregoing and related ends, the one or more aspects include the features hereinafter fully described and particularly pointed out in the claims. The following description and the annexed drawings set forth in detail certain illustrative features of the one or more aspects. These features are indicative, however, of but a few of the various ways in which the principles of various aspects may be employed, and this description is intended to include all such aspects and their equivalents.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] FIG. 1 illustrates an example of an audio signal gap caused by packet lost.

[0009] FIG. 2 illustrates an example of audio waveform in long-term and short-term views.

[0010] FIG. 3 is a diagram illustrating an example of audio data being transmitted in frames in wireless audio application.

[0011] FIG. 4 illustrates an example of an audio data FIFO queue.

[0012] FIG. 5 is a flowchart of a method of audio error concealment.

[0013] FIG. 6 is a diagram illustrating an example of selecting a reference audio pattern to search for the best matched audio data when an audio frame is lost.

[0014] FIG. 7 illustrates an example of reconstructing missing audio frame based on the best matched segment.

[0015] FIG. 8 is a diagram illustrating an example of the 2-stage search of the best matched audio segment.

[0016] FIG. 9 is a flowchart of the first stage of a method of audio error concealment.

[0017] FIG. 10 is a flowchart of the second stage of the method of audio error concealment.

[0018] FIG. 11 is a diagram illustrating an example of using an incremental search sequence to perform an incremental search on an audio buffer.

[0019] FIG. 12 is a flowchart of a method of audio error concealment.

[0020] FIG. 13 is a flowchart of a method of identifying audio segment within the plurality of audio frames that best matches the reference audio pattern of the plurality of audio frames.

[0021] FIG. 14 is a conceptual data flow diagram illustrating the data flow between different means/components in an exemplary apparatus.

[0022] FIG. 15 is a diagram illustrating an example of a hardware implementation for an apparatus employing a processing system.

DETAILED DESCRIPTION

[0023] The detailed description set forth below in connection with the appended drawings is intended as a description of various configurations and is not intended to represent the only configurations in which the concepts described herein may be practiced. The detailed description includes specific details for the purpose of providing a thorough understanding of various concepts. However, it will be apparent to those skilled in the art that these concepts may be practiced without these specific details. In some instances, well known structures and components are shown in block diagram form in order to avoid obscuring such concepts.

[0024] Several aspects of audio error concealment will now be presented with reference to various apparatus and methods. These apparatus and methods will be described in the following detailed description and illustrated in the accompanying drawings by various blocks, components, circuits, processes, algorithms, etc. (collectively referred to as

"elements"). These elements may be implemented using electronic hardware, computer software, or any combination thereof. Whether such elements are implemented as hardware or software depends upon the particular application and design constraints imposed on the overall system.

[0025] By way of example, an element, or any portion of an element, or any combination of elements may be implemented as a "processing system" that includes one or more processors. Examples of processors include microprocessors, microcontrollers, graphics processing units (GPUs), central processing units (CPUs), application processors, digital signal processors (DSPs), reduced instruction set computing (RISC) processors, systems on a chip (SoC), baseband processors, field programmable gate arrays (FPGAs), programmable logic devices (PLDs), state machines, gated logic, discrete hardware circuits, and other suitable hardware configured to perform the various functionality described throughout this disclosure. One or more processors in the processing system may execute software. Software shall be construed broadly to mean instructions, instruction sets, code, code segments, program code, programs, subprograms, software components, applications, software applications, software packages, routines, subroutines, objects, executables, threads of execution, procedures, functions, etc., whether referred to as software, firmware, middleware, microcode, hardware description language, or otherwise.

[0026] Accordingly, in one or more example embodiments, the functions described may be implemented in hardware, software, or any combination thereof. If implemented in software, the functions may be stored on or encoded as one or more instructions or code on a computer-readable medium. Computer-readable media includes computer storage media. Storage media may be any available media that can be accessed by a computer. By way of example, and not limitation, such computer-readable media may include a random-access memory (RAM), a read-only memory (ROM), an electrically erasable programmable ROM (EEPROM), optical disk storage, magnetic disk storage, other magnetic storage devices, combinations of the aforementioned types of computer-readable media, or any other medium that can be used to store computer executable code in the form of instructions or data structures that can be accessed by a computer.

[0027] Some embodiments of the disclosure implement an efficient search method that makes use of incremental search sequence to find similar waveform from the past history and reconstruct/recreate the missing audio data using the short-term periodic properties of

audio signal. FIG. 2 illustrates an example of audio waveform in long-term and short-term views. The diagram 200 shows multi-tone frequency with random period when looking at long-term interval. The diagram 220 shows multi-tone frequency with periodic interval when looking at short-term interval. The audio signal in the diagram 220 may be a small segment of the audio signal in the diagram 200.

[0028] FIG. 3 is a diagram 300 illustrating an example of audio data being transmitted in frames in wireless audio application. In the example, a transmitter 302 sends audio data in frames (e.g., F(n-2), F(n-1), F(n), F(n+1), F(n+2), etc.) to a receiver 304. Since audio data is sequential in nature, the frames are transmitted according to the sequence of the audio data.

[0029] In some embodiments, a frame is a digital data transmission unit in wireless communication. In some embodiments, a frame may be a simple container for a single network packet. In some embodiments, a frame may be a repeating structure supporting time-division multiplexing. In some embodiments, a frame may have a pre-determined length in time.

[0030] In some embodiments, all audio frames may be tagged with a frame sequence number (FSN). In such embodiments, a missing frame may be detected by checking the sequence of received frame sequence numbers.

[0031] In some embodiments, received audio data may be stored in a first in first out (FIFO) queue for pattern matching purposes. FIG. 4 illustrates an example of an audio data FIFO queue 400. In the example, the FIFO queue 400 may have stored several audio data frames, e.g., from F(n-5) to F(n). A search window 402, e.g., from F(n-5) to F(n-1), may be selected to find similar waveform from recently received audio data frames.

[0032] FIG. 5 is a flowchart 500 of a method of audio error concealment. The method may be performed by an apparatus (e.g., apparatus 1402/1402' described below with reference to FIG. 14 or 15). In some embodiments, the apparatus may be a wireless speaker or a wireless headphone.

[0033] At 502, the apparatus may receive an audio frame. At 504, the apparatus may determine whether the received audio frame is continuous from a previously received audio frame. The previously received audio frame is received before the currently received audio frame, and there is no audio frame received between the previously received audio frame and the currently received audio frame. In some embodiments, the frame sequence number

of the frames may indicate whether the frames are continuous. If the received audio frame is continuous from the previously received audio frame, the apparatus may proceed to 506 and 512. If the received audio frame is not continuous from the previously received audio frame, a missing audio frame is detected and the apparatus may proceed to 508.

[0034] At 506, the apparatus may store the received audio frame in an FIFO queue. The FIFO queue may store historic audio data. In some embodiments, the historic audio data stored in the FIFO queue may be recently received audio frames.

[0035] At 508, the apparatus may look for audio data from the FIFO queue that best matches the missing audio frame. At 510, the apparatus may replace the missing audio frame with the best matched audio data. The best matched audio data may be placed before (in time) the currently received audio frame. At 512, the apparatus may play back the reconstructed audio data.

[0036] In some embodiments, when a frame is lost, the last segment (e.g., *n* ms) of the last received audio frame may be used as the reference audio pattern to search for the best matched audio data. FIG. 6 is a diagram 600 illustrating an example of selecting a reference audio pattern to search for the best matched audio data when an audio frame is lost. In the example, when a lost frame 602 is detected, the last segment of the last received audio frame is selected as the reference pattern 604. The reference pattern 604 may be used to search for the best matched audio data in FIFO 608.

[0037] In some embodiments, the best matched segment is determined by at least one of the following methods:

- The audio waveform difference between the reference segment r(n) and the candidate segment s(n) is the smallest

$$Y = \sum_0^n |r(i) - s(i)|;$$

- The correlation between the reference segment r(n) and the candidate segment s(n) is the largest

$$Y = \sum_0^n r(i) * s(i).$$

[0038] In some embodiments, a similarity value may be determined based on at least one of the difference or correlation between the reference segment r(n) and the candidate segment s(n), and the best matched segment may be determined based on all the similarity values. Once the best matched segment is obtained, the subsequent audio data of the best matched segment may be chosen as the replacement data for the missing frame.

[0039] FIG. 7 illustrates an example of reconstructing missing audio frame based on the best matched segment. The diagram 700 shows the waveform of the original received audio with a lost frame 702. The diagram 750 illustrates the waveform of the reconstructed received audio. In the example, the last segment 758 of the last audio frame received before the lost frame 702 is chosen as the reference pattern r(n). The reference pattern r(n) may be used to search for the best matched audio segment 754 in the search duration 752. Once the best matched audio segment 754 is identified, the subsequent audio data for the best matched segment 754 may be chosen as the replacement candidate 756 for the lost frame 702. In some embodiments, the replacement candidate 756 may have the same length (in time) as the lost frame 702. The replacement candidate 756 may be used as the replacement data 760 to replace the lost frame 702. As a result, the gap caused by the lost frame 702 may be mitigated. In some embodiments, the replacement data 760 may be a copy of the replacement candidate 756. In some embodiments, the replacement data 760 may be derived from the replacement candidate 756.

[0040] In some embodiments, in order to search for the best matched audio segment based on the reference pattern, a linear search may be performed. The linear search may search for the similar waveform from the first audio sample to the last audio sample in the audio buffer (e.g., a FIFO queue). The total number of operation is

$$T = n * m,$$

n = number of audio sample,

m = size of search window.

[0041] In some embodiments, a 2-stage search of the best matched audio segment may be implemented. The first stage may make use of incremental search sequence to perform an incremental search for the similar waveform (i.e., a coarse search). In some embodiments, an incremental search sequence may be:

a linear incremental sequence, such as 1, 1, 2, 3, 4, 5, 6, 7, 8,

a linear incremental repeat sequence, such as 1, 1, 2, 2, 3, 3, 4, 4, or

a Fibonacci sequence, such as 1, 1, 2, 3, 5, 8.

[0042] The second stage may make use of a linear search sequence to perform a linear search for the best matched waveform (i.e., a refined search). The incremental search starts from the oldest audio segment in the audio buffer and gradually proceeds to newer audio segments in the buffer. The search index of the incremental search is incremented based on

the incremental search sequence. For example, if the current number in the incremental search sequence is *n*, the search index will be incremented by *n*. Thus, the next audio segment to be searched in the buffer will be *n* audio segments after the currently searched audio segment. The linear search starts from the oldest audio segment in a search window and gradually proceeds to newer audio segments in the search window. For a linear search using a linear search sequence, the next audio segment to be searched will be the audio segment that is subsequently next to the currently searched audio segment in the search window. That is, in a linear search, the search index increments by one each time.

[0043] FIG. 8 is a diagram 800 illustrating an example of the 2-stage search of the best matched audio segment. In the example, at the first stage, an incremental search sequence 1, 1, 2, 3, 4, 5, 6, 7, 8 is used to search the buffer 802 for the best matched audio segment or sample. As a result of the incremental search, audio segment 804 is identified as the best matched audio segment at the first stage. The search index for the audio segment 804 corresponds to the number '6' in the incremental search sequence.

[0044] A search window 806 is identified based on the audio segment 804. The search window 806 may be centered at the audio segment 804. In some embodiments, the search window 806 may start from the search index corresponding to the previous number (i.e., '5') in the incremental search sequence to the search index corresponding to the next number (i.e., '7') in the incremental search sequence.

[0045] At the second stage, a linear search may be performed in the search window 806. That is, every audio segment in the search window 806 may be compared with the reference pattern to find the best matched audio segment.

[0046] FIG. 9 is a flowchart 900 of the first stage of a method of audio error concealment. The method may be performed by an apparatus (e.g., apparatus 1402/1402' described below with reference to FIG. 14 or 15). In some embodiments, the apparatus may be a wireless speaker or a wireless headphone. In some embodiments, the operations performed in the method may correspond to operations described above with reference to FIGS. 4-8.

[0047] At 902, the apparatus may get the next search index of an incremental search sequence, and set it as current search index. At 904, the apparatus may determine whether the end of the search buffer is reached based on the current search index. If the end of the

search buffer is reached, the apparatus may proceed to 910. If the end of the search buffer is not reached, the apparatus may proceed to 906.

[0048] At 906, the apparatus may determine the similarity between the reference pattern and the audio segment corresponding to the current search index. At 908, the apparatus may store the determined similarity value.

[0049] At 910, the apparatus may find the highest similarity value among all the stored similarity values. At 912, the apparatus may find the search index corresponding to the highest similarity value.

[0050] At 914, the apparatus may find the previous search index and the next search index in the incremental search before and after the search index of the highest similarity value to construct the smaller search window. The apparatus may then proceed to connection point B.

[0051] FIG. 10 is a flowchart 1000 of the second stage of the method of audio error concealment. The operations of the flowchart 1000 continues from the operations described above with reference to FIG. 9. The method may be performed by an apparatus (e.g., apparatus 1402/1402' described below with reference to FIG. 14 or 15). In some embodiments, the apparatus may be a wireless speaker or a wireless headphone. In some embodiments, the operations performed in the method may correspond to operations described above with reference to FIGS. 4-8.

[0052] Continuing from the connection point B, at 1002, the apparatus may get the next search index of a linear search, and set it as current search index. At 1004, the apparatus may determine whether the end of the search window is reached based on the current search index. If the end of the search window is reached, the apparatus may proceed to 1010. If the end of the search window is not reached, the apparatus may proceed to 1006. In some embodiments, the search window may be constructed by operations described above in 914 of FIG. 9.

[0053] At 1006, the apparatus may determine the similarity between the reference pattern and the audio segment corresponding to the current search index. At 1008, the apparatus may store the determined similarity value. At 1010, the apparatus may find the highest similarity value among all the stored similarity values.

[0054] At 1012, the apparatus may determine whether the highest similarity value is greater than a threshold. If the highest similarity value is greater than the threshold, the

apparatus may proceed to 1014. If the highest similarity value is not greater than the threshold, the apparatus may proceed to 1018. In some embodiments, the threshold may be a value to indicate the confidence level of the waveform found.

[0055] At 1014, the apparatus may find the search index corresponding to the highest similarity value. At 1016, the apparatus may replace the lost frame based on the search index of the highest similarity value. In some embodiments, the audio data immediately subsequent to the audio segment that has the highest similarity value may be chosen to replace/reconstruct the lost frame. At 1018, the apparatus may switch to the next search sequence for the first stage of the method.

[0056] In some embodiments, for an incremental search using a Fibonacci sequence 1, 1, 2, 3, 5, 8, the total number of operations

$$T = \left(\frac{6}{20}\right) n * m,$$

n = number of audio sample,

m = size of search window.

[0057] FIG. 11 is a diagram 1100 illustrating an example of using an incremental search sequence to perform an incremental search on an audio buffer 1102. In some embodiments, the incremental search in this example may be performed at the first stage of the 2-stage search described above with references to FIGS. 8 and 9.

[0058] In the example, the Fibonacci sequence is used to perform the incremental search. Each number in the Fibonacci sequence corresponds to an audio segment in the audio buffer 1102. For example, the first '1' in the sequence corresponds to S[1]; the second '1' in the sequence corresponds to S[2]; the '2' in the sequence corresponds to S[4], which is two segments after S[2]; the '3' in the sequence corresponds to S[7], which is three segments after S[4], and so on.

[0059] For each iteration of the incremental search, the current number in the search sequence serves as the starting point of the audio buffer 1102. For example, when the current number is the first '1' in the sequence, the reference audio segment 1104 may be compared with the audio data started from S[1] to determine a similarity value between them; when the current number is the '2' in the sequence, the reference audio segment 1104 may be compared with the audio data started from S[4] to determine a similarity value between them, and so on.

[0060] In some embodiments, the number of operations is reduced by a ratio of $\left(1 - \frac{6}{20}\right)$ if the Fibonacci sequence described above is used in the incremental search. By performing audio error concealment, the continuity of audio waveform is preserved, and the popping/cracking noise from audio playback is reduced.

[0061] FIG. 12 is a flowchart 1200 of a method of audio error concealment. The method may be performed by an apparatus (e.g., apparatus 1402/1402' described below with reference to FIG. 14 or 15). In some embodiments, the apparatus may be a wireless speaker or a wireless headphone. In some embodiments, the operations performed in the method may correspond to operations described above with reference to FIGS. 4-11.

[0062] At 1202, the apparatus may receive a plurality of audio frames. In some embodiments, the plurality of audio frames may be the audio frames in the search window 402 described above in FIG. 4 or the audio data in the FIFO 608 described above in FIG. 6. In some embodiments, an audio frame may include amplitude information of an audio signal over a predetermined period of time. In some embodiments, each audio frame may be assigned a frame sequence number to indicate the order of transmission.

[0063] At 1204, the apparatus may receive a first audio frame as the next audio frame received after the plurality of audio frames. At 1206, the apparatus may detect a second audio frame being lost in transmission. The second audio frame is transmitted after the plurality of audio frames and before the first audio frame.

[0064] At 1208, the apparatus may identify an audio segment within the plurality of audio frames that best matches a reference audio pattern of the plurality of audio frames. The identified audio segment may be received before the last audio frame of the plurality of audio frames. In some embodiments, the reference audio pattern may be the last audio segment of the last audio frame of the plurality of audio frames.

[0065] In some embodiments, to identify the audio segment, the apparatus may, for each candidate audio segment, perform at least one of: comparing audio waveform of the candidate audio segment and audio waveform of the reference audio pattern; or determining correlation between the candidate audio segment and the reference audio pattern. In some embodiments, the identified audio segment may have at least one of: the smallest audio waveform difference with the reference audio pattern; or the largest correlation with the reference audio pattern.

[0066] At 1210, the apparatus may reconstruct the second audio frame based on the audio data received subsequent to the identified audio segment. In some embodiments, the apparatus may further play back the reconstructed second audio frame after the plurality of audio frames and before the first audio frame.

[0067] FIG. 13 is a flowchart 1300 of a method of identifying audio segment within the plurality of audio frames that best matches the reference audio pattern of the plurality of audio frames. The method may be performed by an apparatus (e.g., apparatus 1402/1402' described below with reference to FIG. 14 or 15). In some embodiments, the apparatus may be a wireless speaker or a wireless headphone. In some embodiments, the operations performed in the method may correspond to the operations described above with reference to 1208 in FIG. 12.

[0068] At 1302, the apparatus may perform an incremental search of the audio segment within the plurality of audio frames based on an incremental search sequence. In some embodiments, the incremental search sequence may be one of a linear incremental sequence, a linear incremental repeat sequence, or a Fibonacci sequence.

[0069] At 1304, the apparatus may identify a search window within the plurality of audio frames based on the search result of the incremental search.

[0070] At 1306, the apparatus may perform a linear search within the search window to identify the audio segment. In some embodiments, during the incremental search or the linear search, the apparatus may, for each candidate audio segment, perform at least one of: comparing audio waveform of the candidate audio segment and audio waveform of the reference audio pattern; or determining correlation between the candidate audio segment and the reference audio pattern. In some embodiments, the identified audio segment may have at least one of: the smallest audio waveform difference with the reference audio pattern; or the largest correlation with the reference audio pattern.

[0071] FIG. 14 is a conceptual data flow diagram 1400 illustrating the data flow between different means/components in an exemplary apparatus 1402. In some embodiments, the apparatus 1402 may be a wireless speaker or a wireless headphone. The apparatus 1402 may include a reception component 1404 that receives audio frames from a device 1450 that includes a transmitter. In one embodiment, the reception component 1404 may perform the operations described above with reference to 1202 or 1204 in FIG. 12.

[0072] The apparatus 1402 may include a transmission component 1410 that transmits acknowledgments (ACK) regarding the received audio frames to the device 1450. The reception component 1404 and the transmission component 1410 may collaborate to coordinate the communication of the apparatus 1402.

[0073] The apparatus 1402 may include a loss detection component 1406 that is configured to detect a missing frame within the received audio frames. In one embodiment, the loss detection component 1406 may perform the operations described above with reference to 1206 in FIG. 12.

[0074] The apparatus 1402 may include a match identification component 1408 that is configured to identify audio data that best matches the missing frame. In one embodiment, the match identification component 1408 may perform the operations described above with reference to 1208 in FIG. 12, or FIG. 13.

[0075] The apparatus 1402 may include a reconstruction component 1412 that is configured to reconstruct the missing frame based on the identified best match audio data. In one embodiment, the reconstruction component 1412 may perform the operations described above with reference to 1210 in FIG. 12.

[0076] The apparatus 1402 may include additional components that perform each of the blocks of the algorithm in the aforementioned flowcharts of FIGS. 5, 9, 10, 12, and 13. As such, each block in the aforementioned flowcharts of FIGS. 5, 9, 10, 12, and 13 may be performed by a component and the apparatus may include one or more of those components. The components may be one or more hardware components specifically configured to carry out the stated processes/algorithm, implemented by a processor configured to perform the stated processes/algorithm, stored within a computer-readable medium for implementation by a processor, or some combination thereof.

[0077] FIG. 15 is a diagram 1500 illustrating an example of a hardware implementation for an apparatus 1402' employing a processing system 1514. In one embodiment, the apparatus 1402' may be the apparatus 1402 described above with reference to FIG. 14. The processing system 1514 may be implemented with a bus architecture, represented generally by the bus 1524. The bus 1524 may include any number of interconnecting buses and bridges depending on the specific application of the processing system 1514 and the overall design constraints. The bus 1524 links together various circuits including one or more processors and/or hardware components, represented by the processor 1504, the components

1404, 1406, 1408, 1410, 1412 and the computer-readable medium / memory 1506. The bus 1524 may also link various other circuits such as timing sources, peripherals, voltage regulators, and power management circuits, which are well known in the art, and therefore, will not be described any further.

[0078] The processing system 1514 may be coupled to a transceiver 1510. The transceiver 1510 is coupled to one or more antennas 1520. The transceiver 1510 provides a means for communicating with various other apparatus over a transmission medium. The transceiver 1510 receives a signal from the one or more antennas 1520, extracts information from the received signal, and provides the extracted information to the processing system 1514, specifically the reception component 1404. In addition, the transceiver 1510 receives information from the processing system 1514, specifically the transmission component 1410, and based on the received information, generates a signal to be applied to the one or more antennas 1520.

[0079] The processing system 1514 includes a processor 1504 coupled to a computer-readable medium / memory 1506. The processor 1504 is responsible for general processing, including the analyzation of data gathered by the apparatus itself through its own sensors and the execution of software stored on the computer-readable medium / memory 1506. The software, when executed by the processor 1504, causes the processing system 1514 to perform the various functions described *supra* for any particular apparatus. The computer-readable medium / memory 1506 may also be used for storing data that is manipulated by the processor 1504 when executing software. The processing system 1514 further includes at least one of the components 1404, 1406, 1408, 1410, 1412. The components may be software components running in the processor 1504, resident/stored in the computer readable medium / memory 1506, one or more hardware components coupled to the processor 1504, or some combination thereof.

[0080] In the following, various aspects of this disclosure will be illustrated:

[0081] Example 1 is a method or apparatus for audio error concealment. The apparatus may receive a plurality of audio frames. The apparatus may receive a first audio frame immediately after receiving the plurality of audio frames. The apparatus may detect a second audio frame being lost in transmission. The second audio frame is transmitted after the plurality of audio frames and before the first audio frame. The apparatus may identify an audio segment within the plurality of audio frames that best matches a reference audio

pattern of the plurality of audio frames. The identified audio segment may be received before the last audio frame of the plurality of audio frames. The apparatus may reconstruct the second audio frame based on audio data received immediately subsequent to the identified audio segment.

[0082] In Example 2, the subject matter of Example 1 may optionally include that, to identify the audio segment, the apparatus may perform an incremental search of the audio segment within the plurality of audio frames based on an incremental search sequence.

[0083] In Example 3, the subject matter of Example 2 may optionally include that the incremental search sequence may be one of a linear incremental sequence, a linear incremental repeat sequence, or a Fibonacci sequence.

[0084] In Example 4, the subject matter of Example 2 or 3 may optionally include that, to identify the audio segment, the apparatus may further: identify a search window within the plurality of audio frames based on a search result of the incremental search; and perform a linear search within the search window to identify the audio segment.

[0085] In Example 5, the subject matter of any one of Examples 1 to 4 may optionally include that an audio frame may include amplitude information of an audio signal over a predetermined period of time.

[0086] In Example 6, the subject matter of any one of Examples 1 to 5 may optionally include that each audio frame may be assigned a frame sequence number to indicate an order of transmission.

[0087] In Example 7, the subject matter of any one of Examples 1 to 6 may optionally include that, to identify the audio segment, the apparatus may, for each candidate audio segment, perform at least one of: comparing audio waveform of the candidate audio segment and audio waveform of the reference audio pattern; or determining correlation between the candidate audio segment and the reference audio pattern.

[0088] In Example 8, the subject matter of Example 7 may optionally include that the identified audio segment may have at least one of: a smallest audio waveform difference with the reference audio pattern; or a largest correlation with the reference audio pattern.

[0089] In Example 9, the subject matter of any one of Examples 1 to 8 may optionally include that the apparatus may play back the reconstructed second audio frame after the plurality of audio frames and before the first audio frame.

[0090] In Example 10, the subject matter of any one of Examples 1 to 9 may optionally include that the reference audio pattern is the last audio segment of the last audio frame of the plurality of audio frames.

[0091] It is understood that the specific order or hierarchy of blocks in the processes / flowcharts disclosed is an illustration of exemplary approaches. Based upon design preferences, it is understood that the specific order or hierarchy of blocks in the processes / flowcharts may be rearranged. Further, some blocks may be combined or omitted. The accompanying method claims present elements of the various blocks in a sample order, and are not meant to be limited to the specific order or hierarchy presented.

[0092] The previous description is provided to enable any person skilled in the art to practice the various aspects described herein. Various modifications to these aspects will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other aspects. Thus, the claims are not intended to be limited to the aspects shown herein, but is to be accorded the full scope consistent with the language claims, wherein reference to an element in the singular is not intended to mean "one and only one" unless specifically so stated, but rather "one or more." The word "exemplary" is used herein to mean "serving as an example, instance, or illustration." Any aspect described herein as "exemplary" is not necessarily to be construed as preferred or advantageous over other aspects. Unless specifically stated otherwise, the term "some" refers to one or more. Combinations such as "at least one of A, B, or C," "one or more of A, B, or C," "at least one of A, B, and C," "one or more of A, B, and C," and "A, B, C, or any combination thereof" include any combination of A, B, and/or C, and may include multiples of A, multiples of B, or multiples of C. Specifically, combinations such as "at least one of A, B, or C," "one or more of A, B, or C," "at least one of A, B, and C," "one or more of A, B, and C," and "A, B, C, or any combination thereof" may be A only, B only, C only, A and B, A and C, B and C, or A and B and C, where any such combinations may contain one or more member or members of A, B, or C. All structural and functional equivalents to the elements of the various aspects described throughout this disclosure that are known or later come to be known to those of ordinary skill in the art are expressly incorporated herein by reference and are intended to be encompassed by the claims. Moreover, nothing disclosed herein is intended to be dedicated to the public regardless of whether such disclosure is explicitly recited in the claims. The words "module," "mechanism," "element," "device," and the like

may not be a substitute for the word “means.” As such, no claim element is to be construed as a means plus function unless the element is expressly recited using the phrase “means for.”

CLAIMS

WHAT IS CLAIMED IS:

1. A method of audio error concealment, the method comprising:

receiving a plurality of audio frames;

receiving a first audio frame after the receiving of the plurality of audio frames;

detecting a second audio frame being lost in transmission, the second audio frame being transmitted after the plurality of audio frames and before the first audio frame;

identifying an audio segment within the plurality of audio frames that best matches a reference audio pattern of the plurality of audio frames, the identified audio segment being received before the last audio frame of the plurality of audio frames; and

reconstructing the second audio frame based on audio data received subsequent to the identified audio segment.

2. The method of claim 1, wherein the identifying of the audio segment comprises:

performing an incremental search of the audio segment within the plurality of audio frames based on an incremental search sequence.

3. The method of claim 2, wherein the incremental search sequence is one of a linear incremental sequence, a linear incremental repeat sequence, or a Fibonacci sequence.

4. The method of claim 2, wherein the identifying of the audio segment further comprises:

identifying a search window within the plurality of audio frames based on a search result of the incremental search; and

performing a linear search within the search window to identify the audio segment.

5. The method of claim 1, wherein an audio frame comprises amplitude information of an audio signal over a predetermined period of time.

6. The method of claim 1, wherein each audio frame is assigned a frame sequence number to indicate an order of transmission.

7. The method of claim 1, wherein the identifying of the audio segment comprises, for each candidate audio segment, performing at least one of:

comparing audio waveform of the candidate audio segment and audio waveform of the reference audio pattern; or

determining correlation between the candidate audio segment and the reference audio pattern.

8. The method of claim 7, wherein the identified audio segment has at least one of:

a smallest audio waveform difference with the reference audio pattern; or

a largest correlation with the reference audio pattern.

9. The method of claim 1, further comprising playing back the reconstructed second audio frame after the plurality of audio frames and before the first audio frame.

10. The method of claim 1, wherein the reference audio pattern is a last audio segment of the last audio frame of the plurality of audio frames.

11. An apparatus for audio error concealment, the apparatus comprising:

a memory; and

at least one processor coupled to the memory and configured to:

receive a plurality of audio frames;

receive a first audio frame after the receiving of the plurality of audio frames;

detect a second audio frame being lost in transmission, the second audio frame being transmitted after the plurality of audio frames and before the first audio frame;

identify an audio segment within the plurality of audio frames that best matches a reference audio pattern of the plurality of audio frames, the identified audio segment being received before the last audio frame of the plurality of audio frames; and

reconstruct the second audio frame based on audio data received subsequent to the identified audio segment.

12. The apparatus of claim 11, wherein, to identify the audio segment, the at least one processor is configured to:

perform an incremental search of the audio segment within the plurality of audio frames based on an incremental search sequence.

13. The apparatus of claim 12, wherein the incremental search sequence is one of a linear incremental sequence, a linear incremental repeat sequence, or a Fibonacci sequence.

14. The apparatus of claim 12, wherein, to identify the audio segment, the at least one processor is further configured to:

identify a search window within the plurality of audio frames based on a search result of the incremental search; and

perform a linear search within the search window to identify the audio segment.

15. The apparatus of claim 11, wherein an audio frame comprises amplitude information of an audio signal over a predetermined period of time.

16. The apparatus of claim 11, wherein each audio frame is assigned a frame sequence number to indicate an order of transmission.

17. The apparatus of claim 11, wherein, to identify the audio segment, the at least one processor is configured to, for each candidate audio segment, perform at least one of:

comparing audio waveform of the candidate audio segment and audio waveform of the reference audio pattern; or

determining correlation between the candidate audio segment and the reference audio pattern.

18. The apparatus of claim 17, wherein the identified audio segment has at least one of:

a smallest audio waveform difference with the reference audio pattern; or

a largest correlation with the reference audio pattern.

19. The apparatus of claim 11, wherein the at least one processor is further configured to play back the reconstructed second audio frame after the plurality of audio frames and before the first audio frame.

20. The apparatus of claim 11, wherein the reference audio pattern is a last audio segment of the last audio frame of the plurality of audio frames.