

ADAPTIVE GAIN CONTROL FOR DIGITAL AUDIO SAMPLES IN A MEDIA STREAM

CROSS-REFERENCES TO RELATED APPLICATIONS

[0001] This Application claims priority to U.S. Non-Provisional Application Serial No. 12/507,971, filed July 23, 2009.

TECHNICAL FIELD

[0002] Embodiments of the subject matter described herein relate generally to the processing of digital audio samples in a media stream. More particularly, embodiments of the subject matter relate to digitally adjusting gain of digital audio samples in a media stream such that the perceived volume is normalized during presentation.

BACKGROUND

[0003] Recently, consumers have expressed significant interest in “place-shifting” devices that allow viewing of television or other media content at locations other than their primary media presentation device. Place-shifting devices typically packetize media content that can be transmitted over a local or wide area network to a portable computer, mobile phone, personal digital assistant, remote television or other remote device capable of playing back the packetized media stream for the viewer. Place-shifting therefore allows consumers to view their media content from remote locations such as other rooms, hotels, offices, and/or any other locations where portable media player devices can gain access to a wireless or other communications network.

[0004] While place-shifting does greatly improve the convenience afforded to the end user, there remain some challenges related to the manner in which different media streams are presented at the end device. For instance, the digital audio samples in one media stream may be associated with a baseline or average presentation loudness or volume, while the digital audio samples in another media stream may be associated with a different baseline/average presentation loudness or volume. Thus, if the user switches between different media streams the perceived loudness may be inconsistent, and the user will therefore need to adjust the volume control on the presentation device.

[0005] Volume normalization techniques can be utilized to automatically adjust the volume perceived by the user. Some volume normalization techniques operate in the analog domain, and others operate in the digital domain. Digital volume normalization

techniques are best suited for place-shifting applications because the media streams are encoded and transmitted to the presentation device using data packets. Unfortunately, existing digital volume normalization techniques tend to be ineffective and/or they introduce audible artifacts that can be distracting to the user.

BRIEF SUMMARY

[0006] An adaptive gain control method for digital audio samples is provided. The method begins by initializing processing of a media stream. The method continues by adjusting gain of a first set of digital audio samples in the media stream using a fast gain adaptation scheme, resulting in a first group of gain-adjusted digital audio samples having normalized volume during presentation. Thereafter, the method adjusts gain of a second set of digital audio samples in the media stream using a steady state gain adaptation scheme that is different than the fast gain adaptation scheme, resulting in a second group of gain-adjusted digital audio samples having normalized volume during presentation.

[0007] Also provided is a computer program product, which is tangibly embodied in a computer-readable medium. The computer program product is operable to cause a digital media processing device to perform operations for a media stream. These operations include: calculating a loudness estimate for a current block of digital audio samples in the media stream; calculating a reference gain value for the current block of digital audio samples, the reference gain value being influenced by the loudness estimate; calculating a maximum gain value for the current block of digital audio samples; calculating an estimated gain value for the current block of digital audio samples, the estimated gain value being influenced by the reference gain value and the maximum gain value; and calculating a gain value for the current block of digital audio samples, the gain value being influenced by the estimated gain value, the maximum gain value, and a previous gain value for a previous block of digital audio samples in the media stream. The computer program product is also operable to cause the digital media processing device to modify the current block of digital audio samples by applying the gain value to the digital audio samples in the current block of digital audio samples. In certain embodiments the maximum gain value is influenced by dynamic range of the current block of digital audio samples.

[0008] A system for processing digital audio samples in a media stream is also provided. The system includes a first means for adjusting gain of a first block of digital audio samples in the media stream using a fast gain adaptation scheme, resulting in a first

block of gain-adjusted digital audio samples. The system also includes a second means for adjusting gain of a second block of digital audio samples in the media stream using a steady state gain adaptation scheme that is different than the fast gain adaptation scheme, resulting in a second block of gain-adjusted digital audio samples. The system also includes means for transmitting gain-adjusted digital audio samples to a remotely-located media player.

[0009] This summary is provided to introduce a selection of concepts in a simplified form that are further described below in the detailed description. This summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used as an aid in determining the scope of the claimed subject matter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] A more complete understanding of the subject matter may be derived by referring to the detailed description and claims when considered in conjunction with the following figures, wherein like reference numbers refer to similar elements throughout the figures.

[0011] FIG. 1 is a schematic representation of an embodiment of a media presentation system;

[0012] FIG. 2 is a schematic representation of an embodiment of a digital media processing device;

[0013] FIG. 3 is a flow chart that illustrates an embodiment of an adaptive digital gain control process; and

[0014] FIG. 4 is a flow chart that illustrates an embodiment of a digital gain calculation process.

DETAILED DESCRIPTION

[0015] The following detailed description is merely illustrative in nature and is not intended to limit the embodiments of the subject matter or the application and uses of such embodiments. As used herein, the word “exemplary” means “serving as an example, instance, or illustration.” Any implementation described herein as exemplary is not necessarily to be construed as preferred or advantageous over other implementations. Furthermore, there is no intention to be bound by any expressed or implied theory presented in the preceding technical field, background, brief summary or the following detailed description.

[0016] Techniques and technologies may be described herein in terms of functional and/or logical block components, and with reference to symbolic representations of operations, processing tasks, and functions that may be performed by various computing components or devices. Such operations, tasks, and functions are sometimes referred to as being computer-executed, computerized, software-implemented, or computer-implemented. In practice, one or more processor devices can carry out the described operations, tasks, and functions by manipulating electrical signals representing data bits at memory locations in the system memory, as well as other processing of signals. The memory locations where data bits are maintained are physical locations that have particular electrical, magnetic, optical, or organic properties corresponding to the data bits. It should be appreciated that the various block components shown in the figures may be realized by any number of hardware, software, and/or firmware components configured to perform the specified functions. For example, an embodiment of a system or a component may employ various integrated circuit components, e.g., memory elements, digital signal processing elements, logic elements, look-up tables, or the like, which may carry out a variety of functions under the control of one or more microprocessors or other control devices.

[0017] When implemented in software or firmware, various elements of the systems described herein are essentially the code segments or instructions that perform the various tasks. The program or code segments can be stored in a processor-readable medium or transmitted by a computer data signal embodied in a carrier wave over a transmission medium or communication path. The “processor-readable medium” or “machine-readable medium” may include any medium that can store or transfer information. Examples of the processor-readable medium include an electronic circuit, a semiconductor memory device, a ROM, a flash memory, an erasable ROM (EROM), a floppy diskette, a CD-ROM, an optical disk, a hard disk, or the like. The computer data signal may include any signal that can propagate over a transmission medium such as electronic network channels, optical fibers, air, electromagnetic paths, or RF links. The code segments may be downloaded via computer networks such as the Internet, an intranet, a LAN, or the like.

[0018] According to various embodiments, the perceived presentation loudness (i.e., volume) of a media stream is normalized or leveled relative to a reference loudness, such that different media streams are presented at about the same average loudness for a constant volume setting at the presentation device. The volume normalization scheme is

carried out in the digital domain by modifying, adjusting, or otherwise altering the digital audio samples associated with the media streams. In certain embodiments, the digital audio samples are modified by a digital media processing device that encodes and transmits media streams (via a data communication network) to the user's media presentation device (e.g., a laptop computer, a cell phone, a remote set-top box, or the like). The digitally normalized audio samples are transmitted to the presentation device in the desired media stream, resulting in normalized presentation volume for different media streams. Notably, the presentation device itself need not be modified to support the digital volume normalization techniques described here because the digital audio samples arrive at the presentation device after application of digital gain adjustment.

[0019] Turning now to the figures and with initial reference to FIG. 1, an exemplary embodiment of a media presentation system 100 can be utilized to carry out place-shifting of digital media content that includes digital audio samples. This particular embodiment of the system 100 includes a digital media processing device (e.g., a place-shifting encoder system 102) that receives media content 122 from a content source 106, encodes the received content into a streaming format, and then transmits the encoded media stream 120 to a remotely-located digital media player (or other presentation device) 104 over a network 110. The media player 104 receives the encoded media stream 120, decodes the stream, and presents the decoded content to a viewer on a television or other display 108. Although not depicted in FIG. 1, the media player 104 includes or cooperates with at least one speaker, audio transducer, or other sound-generating element that supports the presentation of the audio portion of media streams. In various embodiments, a server 112 may also be provided to communicate with the encoder system 102 and/or the media player 104 via the network 110 to assist these devices in locating each other, maintaining security, providing or receiving content or information, and/or any other features as desired. This feature is not required in all embodiments, however, and the concepts described herein may be deployed in any data streaming application or environment, including place-shifting but also any other media or other data streaming situation.

[0020] The encoder system 102 is any component, hardware, software logic and/or the like capable of transmitting a packetized stream of media content over the network 110. In various embodiments, the encoder system 102 incorporates suitable encoder and/or transcoder (collectively "encoder") logic to convert audio/video or other media content 122 into a packetized format that can be transmitted over the network 110. The

media content 122 may be received in any format, and may be received from any internal or external content source 106 such as any sort of broadcast, cable or satellite television programming source, a “video-on-demand” or similar source, a digital video disk (DVD) or other removable media, a video camera, and/or the like. The encoder system 102 encodes the media content 122 to create the encoded media stream 120 in any manner. In various embodiments, the encoder system 102 contains a transmit buffer 105 that temporarily stores encoded data prior to transmission on the network 110.

[0021] In practice, an embodiment of the encoder system 102 may be implemented using any of the various SLINGBOX products available from Sling Media of Foster City, California, although other products could be used in other embodiments. Certain embodiments of the encoder system 102 are generally capable of receiving the media content 122 from an external content source 106 such as any sort of digital video recorder (DVR), set top box (STB), cable or satellite programming source, DVD player, and/or the like. In such embodiments, the encoder system 102 may additionally provide commands 124 to the content source 106 to produce the desired media content 122. Such commands 124 may be provided over any sort of wired or wireless interface, such as an infrared or other wireless transmitter that emulates remote control commands receivable by the content source 106. Other embodiments, however, particularly those that do not involve place-shifting, may modify or omit this feature entirely.

[0022] In other embodiments, the encoder system 102 may be integrated with any sort of content-receiving or other capabilities typically affiliated with the content source 106. The encoder system 102 may be a hybrid STB or other receiver, for example, that also provides transcoding and place-shifting features. Such a device may receive satellite, cable, broadcast and/or other signals that encode television programming or other content received from an antenna, modem, server and/or other source. A receiver of the encoder system 102 may further demodulate or otherwise decode the received signals to extract programming that can be locally viewed and/or place-shifted to the remotely-located media player 104 as appropriate. In this regard, the encoder system 102 may also include a content database stored on a hard disk drive, memory, or other storage medium to support a personal or digital video recorder (DVR) feature or other content library as appropriate. Hence, in some embodiments, the content source 106 and the encoder system 102 may be physically and/or logically contained within a common component, housing or chassis.

[0023] In still other embodiments, the encoder system 102 includes or is implemented as a software program, applet, or the like executing on a conventional computing system (e.g., a personal computer). In such embodiments, the encoder system 102 may encode, for example, some or all of a screen display typically provided to a user of the computing system for place-shifting to a remote location. One device capable of providing such functionality is the SlingProjector product available from Sling Media of Foster City, California, which executes on a conventional personal computer, although other products could be used as well.

[0024] The media player 104 is any device, component, module, hardware, software and/or the like capable of receiving the encoded media stream 120 from one or more encoder systems 102. In various embodiments, the media player 104 is personal computer (e.g., a “laptop” or similarly portable computer, although desktop-type computers could also be used), a mobile phone, a personal digital assistant, a personal media player, or the like. In many embodiments, the media player 104 is a general purpose computing device that includes a media player application in software or firmware that is capable of securely connecting to the encoder system 102, and is capable of receiving and presenting media content to the user of the device as appropriate. In other embodiments, however, the media player 104 is a standalone or other separate hardware device capable of receiving the encoded media stream 120 via any portion of the network 110 and decoding the encoded media stream 120 to provide an output signal 126 that is presented on the display 108. One example of a standalone media player 104 is the SLINGCATCHER product available from Sling Media of Foster City, California, although other products could be equivalently used.

[0025] The network 110 is any digital or other communications network capable of transmitting messages between senders (e.g., the encoder system 102) and receivers (e.g., the media player 104). In various embodiments, the network 110 includes any number of public or private data connections, links or networks supporting any number of communications protocols. The network 110 may include the Internet, for example, or any other network based upon TCP/IP or other conventional protocols. In various embodiments, the network 110 also incorporates a wireless and/or wired telephone network, such as a cellular communications network for communicating with mobile phones, personal digital assistants, and/or the like. The network 110 may also incorporate any sort of wireless or wired local area networks, such as one or more IEEE 802.3 and/or IEEE 802.11 networks.

[0026] The encoder system 102 and/or the media player 104 are therefore able to communicate in any manner with the network 110 (e.g., using any sort of data connections 128 and/or 125, respectively). Such communication may take place over a wide area link that includes the Internet and/or a telephone network, for example; in other embodiments, communications between the encoder system 102 and the media player 104 may take place over one or more wired or wireless local area links that are conceptually incorporated within the network 110. In various equivalent embodiments, the encoder system 102 and the media player 104 may be directly connected via any sort of cable (e.g., an Ethernet cable or the like) with little or no other network functionality provided.

[0027] Many different place-shifting scenarios could be formulated based upon available computing and communications resources, consumer demand and/or any other factors. In various embodiments, consumers may wish to place-shift content within a home, office or other structure, such as from the encoder system 102 to a desktop or portable computer located in another room. In such embodiments, the content stream will typically be provided over a wired or wireless local area network operating within the structure. In other embodiments, consumers may wish to place-shift content over a broadband or similar network connection from a primary location to a computer or other remote media player 104 located in a second home, office, hotel or other remote location. In still other embodiments, consumers may wish to place-shift content to a mobile phone, personal digital assistant, media player, video game player, automotive or other vehicle media player, and/or other device via a mobile link (e.g., a GSM/EDGE or CDMA/EVDO connection, any sort of 3G or subsequent telephone link, an IEEE 802.11 “Wi-Fi” link, and/or the like). Several examples of place-shifting applications available for various platforms are provided by Sling Media of Foster City, California, although the concepts described herein could be used in conjunction with products and services available from any source.

[0028] FIG. 2 is a schematic representation of an embodiment of a digital media processing device, such as the encoder system 102. Again, the encoder system 102 generally creates an encoded media stream 120 that is routable on the network 110 based upon the media content 122 received from the content source 106. In this regard, and with reference now to FIG. 2, the encoder system 102 typically includes an encoding module 202, a transmit buffer 105, and a network interface 206 in conjunction with appropriate control logic, which may be associated with a control module 205. In operation, the encoding module 202 typically receives the media content 122 from the

internal or external content source 106, encodes the data into the desired format for the encoded media stream 120, and stores the encoded data in the transmit buffer 105. The network interface 206 then retrieves the formatted data from the transmit buffer 105 for transmission on the network 110. The control module 205 suitably monitors and controls the encoding and network transmit processes carried out by the encoding module 202 and the network interface 206, respectively, and may perform other functions as well. The encoder system 102 may also have a command module 208 or other feature capable of generating and providing the commands 124 to the content source 106, as described above.

[0029] As noted above, creating the encoded media stream 120 typically involves encoding and/or transcoding the media content 122 received from the content source 106 into a suitable digital format that can be transmitted on the network 110. Generally, the encoded media stream 120 is placed into a standard or other known format (e.g., the WINDOWS MEDIA format available from the Microsoft Corporation of Redmond, Washington, the QUICKTIME format, the REALPLAYER format, an MPEG format, and/or the like) that can be transmitted on the network 110. This encoding may take place, for example, in any sort of encoding module 202 as appropriate. The encoding module 202 may be any sort of hardware (e.g., a digital signal processor or other integrated circuit used for media encoding), software (e.g., software or firmware programming used for media encoding that executes at the encoder system 102), or the like. The encoding module 202 is therefore any feature that receives the media content 122 from content source 106 (e.g., via any sort of hardware and/or software interface) and encodes or transcodes the received data into the desired format for transmission on the network 110. Although FIG. 2 shows a single encoding module 202, in practice the encoder system 102 may include any number of encoding modules 202. Different encoding modules 202 may be selected based upon the type of media player 104, network conditions, user preferences, and/or the like.

[0030] In various embodiments, the encoding module 202 may also apply other modifications, transforms, and/or filters to the received content before or during the encoding/transcoding process. Video signals, for example, may be resized, cropped and/or skewed. Similarly, the color, hue and/or saturation of the signal may be altered, and/or noise reduction or other filtering may be applied. Digital rights management encoding and/or decoding may also be applied in some embodiments, and/or other features may be applied as desired. Audio signals may be modified by adjusting

sampling rate, mono/stereo parameters, noise reduction, multi-channel sound parameters and/or the like. In this regard, digital audio samples in a media stream can be modified in accordance with the adaptive digital gain control techniques and methodologies described in more detail below. Such gain control techniques can be used to modify blocks of digital audio samples to normalize the volume or loudness perceived by the user during presentation of the media stream.

[0031] The network interface 206 refers to any hardware, software and/or firmware that allows the encoder system 102 to communicate on the network 110. In various embodiments, the network interface 206 includes suitable network stack programming and other features and/or conventional network interface (NIC) hardware such as any wired or wireless interface as desired.

[0032] In various embodiments, the control module 205 monitors and controls the encoding and transmit processes performed by the encoding module 202 and the network interface 206, respectively. To that end, the control module 205 is any hardware, software, firmware or combination thereof capable of performing such features. In various embodiments, the control module 205 further processes commands received from the remote media player via the network interface 206 (e.g., by sending the commands 124 to the content source 106 via the command module 208 or the like). The control module 205 may also transmit commands to the media player 104 via the network interface 206 and/or may control or otherwise effect any other operations of the encoder system 102. In various embodiments, the control module 205 implements the control features used to monitor and adjust the operation of the encoding module 202 and/or the network interface 206 to efficiently provide the media stream to the media player 104.

[0033] Certain embodiments of the encoding module 202 may include or execute one or more computer programs (e.g., software) that are tangibly embodied in appropriately configured computer-readable media 210. The computer program product is operable to cause the encoder system 102 to perform certain operations on media streams, as described in more detail below with reference to FIG. 3 and FIG. 4. For this embodiment, FIG. 2 depicts the computer-readable media 210 associated with the control module 205, although such an association need not be employed in all implementations. Indeed, the computer-readable media 210 may alternatively (or additionally) be utilized in connection with the encoding module 202 if so desired. As mentioned above, the computer-readable media 210 may include, without limitation: an electronic circuit, a semiconductor

memory device, a ROM, a flash memory, an erasable ROM (EROM), a floppy diskette, a CD-ROM, an optical disk, a hard disk, or the like.

[0034] FIG. 3 is a flow chart that illustrates an embodiment of an adaptive digital gain control process 300, and FIG. 4 is a flow chart that illustrates an embodiment of a digital gain calculation process 400. The various tasks performed in connection with an illustrated process may be performed by software, hardware, firmware, or any combination thereof. The operations and instructions associated with a described process may be executed by any processor and/or other processing features within the encoder system 102, and the particular means used to implement each of the various functions shown in the figures, then, could be any sort of processing hardware (such as the control module 205 of FIG. 2) executing software or processor-based logic in any format. It should be appreciated that a described process may include any number of additional or alternative tasks, or may omit one or more illustrated tasks. Moreover, the tasks shown in the figures need not be performed in the illustrated order, and a described process may be incorporated into a more comprehensive procedure or process having additional functionality not described in detail herein.

[0035] Referring now to FIG. 3, the adaptive digital gain control process 300 can be started when a new media stream has been designated for encoding and delivery to a presentation device. A media stream may include or otherwise be associated with a plurality of digital audio samples. As used here, a digital audio sample corresponds to a digital representation of an analog audio signal taken at a point in time (or over a relatively short period of time), as is well understood. The magnitude of the analog audio signal is converted into a digital representation, and the digital audio sample includes a number of bits that conveys that digital representation. In certain embodiments, one digital audio sample is represented by sixteen bits (although the number of bits in practice may be more or less than sixteen if so desired). In accordance with some embodiments, the sampling rate for digital audio samples is within the range of about 16,000 to 48,000 samples per second, and certain embodiments utilize a sampling rate of 32,000 samples per second.

[0036] As used here, a “block” of samples refers to a set, group, or other collection of samples contained in a media stream. The number of samples in a block can be arbitrarily defined, or the number may be selected for compatibility with certain data communication standards, streaming media standards, hardware requirements, and/or software requirements. In certain embodiments, one block of digital audio samples is

represented by 1024 consecutive samples (although the number of samples per block may be more or less than 1024 if so desired). Consequently, for a sampling rate of 32,000 samples per second, one block of digital audio samples represents about 32 ms of time.

[0037] The process 300 is able to respond in a real-time and dynamic manner to accommodate different media streams (the end user may switch from one media stream to another for presentation at the media player). In this regard, the process 300 begins by initializing processing of a media stream and entering a fast adaptation mode (task 302). The fast adaptation mode represents an initial training or learning period for the gain control technique described herein. In practice, the fast adaptation mode quickly adjusts the gain of the initial digital audio samples in the media stream so that the volume perceived by the user can be normalized (if needed) by at least some amount. Thereafter, the encoder system can transition to a steady state mode during which the gain control is adjusted in a more accurate and controlled manner.

[0038] For processing of digital audio samples during the fast adaptation mode, the encoder system utilizes a designated set of weighting factors; these weighting factors are used to calculate the gain to be applied to blocks of digital audio samples. The weighting factors control the extent to which the gain applied to adjacent blocks of digital audio samples can differ. The weighting factors and exemplary gain calculation methodologies are described in more detail below with reference to FIG. 4. In practice, the weighting factors are empirically determined values that are accessible by the encoder system. Although any number of weighting factors could be used, this particular embodiment uses two weighting factors, which are labeled w_1 and w_2 herein. Moreover, the values of w_1 and w_2 are variable for the fast gain adaptation scheme, and the values of w_1 and w_2 are fixed for the steady state gain adaptation scheme. For the embodiment described here, w_1 and w_2 are positive and satisfy the relationship $w_1 + w_2 = 1.0$ in both fast adaptation and steady state modes of operation.

[0039] In certain situations, the value of w_1 for the fast adaptation mode is less than or equal to the value of w_1 for the steady state mode, and the value of w_2 for the fast adaptation mode is greater than or equal to the value of w_2 for the steady state mode. In certain situations, $w_1 > w_2$ for both the fast adaptation mode and the steady state mode. In certain situations, the difference $w_1 - w_2$ for the fast adaptation mode is less than the difference $w_1 - w_2$ for the steady state mode. As one specific non-limiting example, in

the fast adaptation mode, $w_1 = 0.7$ and $w_2 = 0.3$, and in the steady state mode, $w_1 = 0.9$ and $w_2 = 0.1$.

[0040] For certain embodiments, the value of w_1 in the fast adaptation mode is always less than the value of w_1 in the steady state mode (although at the transition between modes the value might be the same), the value of w_2 in the fast adaptation mode is always greater than the value of w_2 in the steady state mode (although at the transition between modes the value might be the same), and the value of w_1 is always greater than the value of w_2 (for both modes). Moreover, the values of w_1 and w_2 need not remain constant during the fast adaptation mode. Indeed, these values can be adjusted during the fast adaptation mode to arrive at the values to be used during the steady state mode. In a typical implementation, w_1 increases linearly from one value (for example, 0.7) to another value (for example, 0.9), and w_2 decreases linearly from one value (for example, 0.3) to another value (for example, 0.1) during the fast adaptation mode.

[0041] For example, suppose that the process 300 remains in the fast adaptation mode for a hundred blocks of audio samples. When the process 300 enters the fast adaptation mode, the values of w_1 and w_2 are initialized to 0.7 and 0.3, respectively. After the gain adjustment of the first block of audio samples, w_1 is increased by $\frac{0.9 - 0.7}{100}$ and w_2 is decreased by $\frac{0.3 - 0.1}{100}$. This modification of w_1 and w_2 happens after gain adjustment of

each block of digital audio samples as long as the process 300 remains in the fast adaptation mode. By the time the process 300 reaches the end of the fast adaptation mode, the linear adjustments of w_1 and w_2 result in values of 0.9 and 0.1, respectively. Thereafter, the process 300 enters the steady state mode with these values. In the steady state adaptation mode, w_1 and w_2 do not undergo any further changes and they remain constant at their respective values (0.9 and 0.1 for this example). Later at some point in time, when the system switches back to the fast adaptation mode, w_1 and w_2 are again reset to their initial values (0.7 and 0.3 for this example) and linear adjustment occurs as explained above.

[0042] Referring back to FIG. 3, the process 300 can compute or access the gain weighting factors for the fast adaptation mode (task 304), and obtain the next block of digital audio samples for processing (task 306) during the fast adaptation mode. The process 300 then continues by calculating a gain value for the current block and adjusting the gain of the digital audio samples in the current block in accordance with the

calculated gain value (task 308). Such adjustment or modification of the digital audio samples results in gain-adjusted digital audio samples having normalized volume (loudness) during presentation at the destination media player. During task 308, the gain value for the current block is calculated using the currently applicable weighting factors (w_1 and w_2) corresponding to the fast adaptation mode. In this regard, the gain of the digital audio samples is adjusted using a fast gain adaptation scheme for as long as the process 300 remains in the fast adaptation mode. For the embodiments described here, the gain value represents a multiplicative gain that is used as a multiplier for the original non-adjusted digital audio sample value. Thus, a gain value of one represents no change and the original digital audio sample value will remain unchanged with a gain value of one.

[0043] The process 300 transmits the gain-adjusted digital audio samples to the remotely-located digital media player (task 310) in an ongoing manner. In certain embodiments, task 310 transmits the gain-adjusted digital audio samples in blocks, as is well understood. Moreover, the gain-adjusted digital audio samples will typically be transmitted in a media stream that also includes or otherwise conveys video content. Upon receipt, the media player simply decodes and presents the media stream to the user as usual. The media player need not perform any additional or special processing to implement the volume normalizing technique described here because the digital audio samples received by the media player are already gain-adjusted.

[0044] As mentioned previously, the fast adaptation mode is utilized as a brief training or learning period for new media streams. Accordingly, the encoder system may determine, detect, or otherwise be instructed to switch from the fast adaptation mode to the steady state mode (query task 312). If the process 300 detects a mode switching condition, then it can enter the steady state mode (task 314). Otherwise, the process 300 can return to task 304 to compute or access the newly adjusted values of w_1 and w_2 , obtain the next block of audio samples for processing, and continue as described previously. The mode switching condition can be associated with one or any number of appropriate metrics, measures, or parameters. For example, the fast adaptation mode may remain active for a predetermined time period after initializing the processing of the current media stream, for a predetermined time period after the system enters the fast adaptation mode, for a predetermined time period after the weighting factors are computed or retrieved in task 304, etc. In typical implementations, the fast adaptation mode lasts for about four to eight seconds. As another example, the fast adaptation mode may remain

active for a predetermined number of blocks (or samples) after initializing the processing of the current media stream, for a predetermined number of blocks (or samples) after the system enters the fast adaptation mode, for a predetermined number of blocks (or samples) after the weighting factors are retrieved in task 304, etc.

[0045] Assuming that it is time for the encoder system to switch modes, the process will enter and initiate the steady state mode. The steady state mode represents a “long term” and relatively stable period for the gain control technique described herein. In practice, the steady state mode takes over after the fast adaptation mode has made its initial gain adjustments. During the steady state mode, the gain of the digital audio samples is adjusted in an ongoing and accurate manner so that the volume perceived by the user remains normalized (if needed) relative to the reference volume level.

[0046] For the steady state mode, the process 300 retrieves or accesses the gain weighting factors for the steady state mode (task 316), which are different than the gain weighting factors used during the fast adaptation mode. The process 300 also obtains the next block of digital audio samples for processing (task 318) during the steady state mode. The process 300 then continues by calculating the gain value for the current block and adjusting the gain of the digital audio samples in the current block in accordance with the calculated gain value (task 320). During task 320, the gain value for the current block is calculated using the gain weighting factors (w_1 and w_2) corresponding to the steady state mode. Therefore, the gain of the digital audio samples is adjusted using a steady state gain adaptation scheme for as long as the process 300 remains in the steady state mode, where the steady state gain adaptation scheme is different than the fast gain adaptation scheme. The process 300 transmits the gain-adjusted digital audio samples to the remotely-located digital media player (task 322) as described above for task 310. Again, the media player need not perform any additional or special processing to implement the volume normalizing technique described here because the digital audio samples received by the media player are already gain-adjusted.

[0047] As mentioned previously, the process 300 can be repeated for each new media stream. Thus, the encoder system may determine, detect, or otherwise be instructed to switch from the current audio stream to a new audio stream (query task 324). If the process 300 detects a new media or audio stream, then it can initialize the processing of the new media stream and again enter the fast adaptation mode (task 302). Otherwise, the process 300 can return to task 318, obtain the next block of audio samples for processing in the steady state mode, and continue as described previously.

[0048] Although the embodiment described here uses two modes (fast adaptation and steady state), an adaptive digital gain control technique could instead employ only one mode, or it could employ more than two different modes. The use of two different modes strikes a good balance between audio quality, effectiveness, and normalization speed.

[0049] Referring now to FIG. 4, the digital gain calculation process 400 can be utilized by the encoder system during the adaptive digital gain control process 300. The process 400 is performed during both the fast adaptation mode and the steady state mode (with different values for the weighting factors w_1 and w_2 , as explained previously). The process 400 may begin (task 402) with the first block of digital audio samples (where k indicates the block number), and by obtaining that block of digital audio samples in the media stream for processing (task 404). The process 400 considers the digital audio samples in blocks because a single audio sample conveys no inherent loudness or volume information by itself. For this embodiment, the process 400 calculates a loudness estimate (L_k) for the current block of digital audio samples (task 406), where the k th block includes N samples: $\{a_1, a_2, a_3, \dots, a_N\}$. Although other estimating methodologies could be employed, the embodiment described here calculates the loudness estimate in

accordance with the expression $L_k = \sum_{i=1}^{i=N} |a_i|$, where L_k is the loudness estimate, a_i

represents the digital audio samples, and the current block includes N digital audio samples. The absolute value of each audio sample is taken because any given audio sample may be positive or negative, depending upon its intended sound pressure direction relative to the listener's eardrums. Thus, the process 400 calculates the loudness estimate as a sum of the "magnitudes" of the audio samples contained in the current block.

[0050] The calculated loudness estimate can then be compared to a silence threshold value (query task 408). The silence threshold value may be empirically determined and defined such that it is low enough to serve as an accurate threshold and high enough to contemplate bit errors, artifacts, inconsistencies in the original audio data, and "non-zero" audio samples that cannot be detected as sound. If the calculated loudness estimate is less than the silence threshold, then the process 400 can apply a multiplicative gain of one (or any baseline value, which may but need not be approximately equal to one) to the current block of digital audio samples (task 410). In other words, the process 400 assumes that the gain value (g_k) for the current block will be equal to one. Thus, if the process 400 determines that the current block represents silence, then there is no need to apply any

gain, and the remainder of the process 400 can be bypassed. As explained above with reference to FIG. 3 and the process 300, while operating in the fast adaptation mode, the values of w_1 and w_2 are linearly updated on a block-by-block basis (task 412). In this regard, task 412 leads back to task 404 so that the process 400 can obtain the next block for processing. If in the steady state mode, then task 412 would be bypassed.

[0051] If query task 408 determines that the loudness estimate is not less than the silence threshold, then the process 400 may continue by determining or calculating a reference gain value (g_r) that is influenced by the loudness estimate (task 414). More specifically, the reference gain value is based upon the loudness estimate and a reference loudness value. Although other methodologies could be employed, the embodiment described here calculates the reference gain value in accordance with the expression

$$g_r = \frac{L_{ref}}{L_k}, \text{ where } g_r \text{ is the reference gain value, and } L_{ref} \text{ is the reference loudness value.}$$

The reference loudness value is a constant value that represents, corresponds to, or otherwise indicates the desired normalized volume. Ideally and theoretically, therefore, gain-adjusted digital audio samples will be characterized by an adjusted loudness that corresponds to the reference loudness value. The reference gain value is used later in the process 400.

[0052] The process 400 may also calculate a maximum gain value (g_m) for the current block of digital audio samples (task 416). Although other methodologies could be employed, the embodiment described here calculates the maximum gain value in

accordance with the expression $g_m = \frac{2^{n-1}}{a_{max}}$, where n is the number of bits per digital

audio sample, and where a_{max} is the maximum absolute sample value in the current block of digital audio samples. The process 400 determines the maximum allowable gain value for the current block in this manner to prevent bit overflow in the digital audio samples of the block. For example, if the current block includes a digital audio sample that has a relatively high value that approaches the maximum sample value, then very little multiplicative gain can be applied to that sample without causing overflow. On the other hand, if all of the samples in the block have relatively low values, then a higher amount of multiplicative gain can be applied to the block.

[0053] This particular embodiment also determines or calculates an estimated gain value (g_e) for the current block of digital audio samples (task 418), where the estimated

gain value is influenced by the reference gain value and/or by the maximum gain value. More specifically, the estimated gain value is based upon the reference gain value and the maximum gain value. Although other techniques could be employed, the embodiment described here calculates the estimated gain value in accordance with the expression $g_e = \min(g_r, g_m)$, where g_e is the estimated gain value. Thus, the estimated gain value will be equal to either the reference gain value or the maximum gain value, whichever one is lower (or equal to both if they are the same).

[0054] Next, the process 400 calculates the gain value (g_k) to be applied to the current block of digital audio samples (task 420). Although other methodologies could be employed, the embodiment described here calculates the estimated gain value in accordance with the expression $g_k = \min(w_1 \times g_{k-1} + w_2 \times g_e, g_m)$, where g_k is the computed gain value, and w_1 and w_2 are the weighting factors, which were described previously. The expression for g_k includes a minimum operator that selects one of two values, whichever is lower (or selects either value if they are both the same). The first value is defined by the term $w_1 \times g_{k-1} + w_2 \times g_e$, and the second value is the maximum gain value (g_m). As indicated by this expression, the gain value will be influenced by the estimated gain value, the maximum gain value, and a previous gain value (g_{k-1}) for a previous block of digital audio samples in the media stream. For the reasons described above, this computed gain value will also be influenced by the reference gain value calculated during task 414. For this particular embodiment, the gain value for the current block (g_k) is determined in response to the gain value for the immediately preceding block (g_{k-1}). Alternatively (or additionally), g_k could be calculated by considering other gain values for blocks prior to the immediately preceding block. This reliance on a previous gain value prevents large block-to-block variations in the applied gain. In practice, g_k will have a value that ranges from 1.0 to about 8.0, although values that exceed 8.0 might be realized in certain embodiments.

[0055] The computed gain value for the current block of digital audio samples can then be applied to the samples in the block (task 422). In practice, task 422 modifies, adjusts, or otherwise changes the current block of digital audio samples. More specifically, task 422 modifies each sample in the current block by multiplying the original sample value by g_k , resulting in a gain-adjusted sample value. As mentioned above with reference to the process 300, the gain-adjusted sample values are sent to the remotely-located media player, which can then present the media stream with a

normalized loudness. FIG. 4 depicts task 422 leading back to task 412 as an indication of the potentially ongoing block-by-block nature of the process 400. As described previously, the process 400 can be initially performed for the fast adaptation mode (using the linearly adjusted set of weighting factors) and then repeated for the steady state mode (using a fixed set of weighting factors).

[0056] While at least one exemplary embodiment has been presented in the foregoing detailed description, it should be appreciated that a vast number of variations exist. It should also be appreciated that the exemplary embodiment or embodiments described herein are not intended to limit the scope, applicability, or configuration of the claimed subject matter in any way. Rather, the foregoing detailed description will provide those skilled in the art with a convenient road map for implementing the described embodiment or embodiments. It should be understood that various changes can be made in the function and arrangement of elements without departing from the scope defined by the claims, which includes known equivalents and foreseeable equivalents at the time of filing this patent application.

CLAIMS

What is claimed is:

1. An adaptive gain control method for digital audio samples, the method comprising:
 - initializing processing of a media stream;
 - adjusting gain of a first set of digital audio samples in the media stream using a fast gain adaptation scheme, resulting in a first group of gain-adjusted digital audio samples having normalized volume during presentation; and
 - thereafter, adjusting gain of a second set of digital audio samples in the media stream using a steady state gain adaptation scheme that is different than the fast gain adaptation scheme, resulting in a second group of gain-adjusted digital audio samples having normalized volume during presentation.
2. The method of claim 1, wherein adjusting gain of the first set of digital audio samples is performed for a predetermined time period after initializing processing of the media stream.
3. The method of claim 1, wherein adjusting gain of the first set of digital audio samples is performed for a predetermined number of blocks of digital audio samples after initializing processing of the media stream.
4. The method of claim 3, wherein:
 - one digital audio sample is represented by sixteen bits; and
 - one block of digital audio samples is represented by 1024 digital audio samples.
5. The method of claim 1, further comprising, after adjusting gain of the second set of digital audio samples:
 - initializing processing of a second media stream;
 - adjusting gain of a third set of digital audio samples in the second media stream using the fast gain adaptation scheme, resulting in a third group of gain-adjusted digital audio samples having normalized presentation volume; and

thereafter, adjusting gain of a fourth set of digital audio samples in the second media stream using the steady state gain adaptation scheme, resulting in a fourth group of gain-adjusted digital audio samples having normalized presentation volume.

6. The method of claim 1, further comprising:
calculating a loudness estimate for a block of digital audio samples in the media stream;
comparing the loudness estimate to a silence threshold value; and
applying a multiplicative gain of one to the block of digital audio samples when the loudness estimate is less than the silence threshold value.
7. The method of claim 1, further comprising the step of calculating a gain value for a current block of digital audio samples in the media stream, wherein the gain value is influenced by a previous gain value for a previous block of digital audio samples in the media stream.
8. The method of claim 1, further comprising:
calculating a loudness estimate for a block of digital audio samples in the media stream;
determining a reference gain value based upon the loudness estimate and a reference loudness value; and
calculating a gain value for the block of digital audio samples, wherein the gain value is influenced by the reference gain value.
9. The method of claim 1, further comprising:
calculating a maximum gain value for a block of digital audio samples in the media stream; and
calculating a gain value for the block of digital audio samples, wherein the gain value is influenced by the maximum gain value.
10. The method of claim 1, wherein adjusting gain of the first set of digital audio samples and adjusting gain of the second set of digital audio samples each comprises:
obtaining a current block of digital audio samples in the media stream for processing;

calculating a loudness estimate (L_k) for the current block of digital audio samples in accordance with the expression $L_k = \sum_{i=1}^{i=N} |a_i|$, where a_i represents the digital audio samples, and where the current block of digital audio samples includes N digital audio samples;

calculating a reference gain value (g_r) for the current block of digital audio samples in accordance with the expression $g_r = \frac{L_{ref}}{L_k}$, where L_{ref} is a reference loudness value;

calculating a maximum gain value (g_m) for the current block of digital audio samples in accordance with the expression $g_m = \frac{2^{n-1}}{a_{max}}$, where n is the number of bits per digital audio sample, and where a_{max} is the maximum absolute sample value in the current block of digital audio samples;

calculating an estimated gain value (g_e) for the current block of digital audio samples in accordance with the expression $g_e = \min(g_r, g_m)$; and

calculating a gain value for the current block of digital audio samples in accordance with the expression $g_k = \min(w_1 g_{k-1} + w_2 g_e, g_m)$, wherein w_1 and w_2 are weighting factors, the values of w_1 and w_2 are variable for the fast gain adaptation scheme, and the values of w_1 and w_2 are fixed for the steady state gain adaptation scheme.

11. The method of claim 10, wherein;

w_1 and w_2 are weighting factors;

$w_1 + w_2 = 1.0$;

$w_1 > 0.0$; and

$w_2 > 0.0$.

12. The method of claim 1, further comprising:

transmitting the first group of gain-adjusted digital audio samples to a remotely-located digital media player; and

transmitting the second group of gain-adjusted digital audio samples to the remotely-located digital media player.

13. A computer program product, tangibly embodied in a computer-readable medium, the computer program product being operable to cause a digital media processing device to perform operations for a media stream, comprising:

calculating a loudness estimate for a current block of digital audio samples in the media stream;

calculating a reference gain value for the current block of digital audio samples, the reference gain value being influenced by the loudness estimate;

calculating a maximum gain value for the current block of digital audio samples;

calculating an estimated gain value for the current block of digital audio samples, the estimated gain value being influenced by the reference gain value and the maximum gain value;

calculating a gain value for the current block of digital audio samples, the gain value being influenced by the estimated gain value, the maximum gain value, and a previous gain value for a previous block of digital audio samples in the media stream; and

modifying the current block of digital audio samples by applying the gain value to the digital audio samples in the current block of digital audio samples.

14. The computer program product of claim 13, wherein the computer program product is operable to cause the digital media processing device to perform further operations, comprising:

comparing the loudness estimate to a silence threshold value; and

applying a multiplicative gain value of one to the current block of digital audio samples when the loudness estimate is less than the silence threshold value.

15. The computer program product of claim 13, wherein calculating the loudness estimate is performed in accordance with the expression $L_k = \sum_{i=1}^{i=N} |a_i|$, where L_k is the loudness estimate, a_i represents the digital audio samples, and the current block includes N digital audio samples.

16. The computer program product of claim 15, wherein calculating the reference gain value is performed in accordance with the expression $g_r = \frac{L_{ref}}{L_k}$, where g_r is the reference gain value, and L_{ref} is a reference loudness value.

17. The computer program product of claim 16, wherein:
calculating the maximum gain value is performed in accordance with the expression $g_m = \frac{2^{n-1}}{a_{max}}$, where g_m is the maximum gain value, n is the number of bits per digital audio sample, and a_{max} is the maximum absolute sample value in the current block of digital audio samples; and
calculating the estimated gain value is performed in accordance with the expression $g_e = \min(g_r, g_m)$, where g_e is the estimated gain value.

18. The computer program product of claim 17, wherein calculating the gain value is performed in accordance with the expression $g_k = \min(w_1 g_{k-1} + w_2 g_e, g_m)$, where g_k is the gain value, and w_1 and w_2 are weighting factors.

19. A system for processing digital audio samples in a media stream, the system comprising:

a first means for adjusting gain of a first block of digital audio samples in the media stream using a fast gain adaptation scheme, resulting in a first block of gain-adjusted digital audio samples;

a second means for adjusting gain of a second block of digital audio samples in the media stream using a steady state gain adaptation scheme that is different than the fast gain adaptation scheme, resulting in a second block of gain-adjusted digital audio samples; and

means for transmitting gain-adjusted digital audio samples to a remotely-located media player.

20. The system of claim 19, wherein each of the first means for adjusting and the second means for adjusting is configured to:

calculate a loudness estimate for a current block of digital audio samples in the media stream;

calculate a reference gain value for the current block of digital audio samples, the reference gain value being influenced by the loudness estimate;

calculate a maximum gain value for the current block of digital audio samples;

calculate an estimated gain value for the current block of digital audio samples, the estimated gain value being influenced by the reference gain value and the maximum gain value;

calculate a gain value for the current block of digital audio samples, the gain value being influenced by the estimated gain value, the maximum gain value, and a previous gain value for a previous block of digital audio samples in the media stream; and

modify the current block of digital audio samples by applying the gain value to the digital audio samples in the current block of digital audio samples.