

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4230636号
(P4230636)

(45) 発行日 平成21年2月25日(2009.2.25)

(24) 登録日 平成20年12月12日(2008.12.12)

(51) Int.Cl.

H04N 7/26 (2006.01)
H03M 7/30 (2006.01)

F 1

H04N 7/13
H03M 7/30Z
Z

請求項の数 6 (全 14 頁)

(21) 出願番号 特願2000-54598 (P2000-54598)
 (22) 出願日 平成12年2月29日 (2000.2.29)
 (65) 公開番号 特開2001-245294 (P2001-245294A)
 (43) 公開日 平成13年9月7日 (2001.9.7)
 審査請求日 平成18年1月26日 (2006.1.26)

(73) 特許権者 000003078
 株式会社東芝
 東京都港区芝浦一丁目1番1号
 (74) 代理人 100058479
 弁理士 鈴江 武彦
 (74) 代理人 100084618
 弁理士 村松 貞男
 (74) 代理人 100092196
 弁理士 橋本 良郎
 (74) 代理人 100091351
 弁理士 河野 哲
 (74) 代理人 100088683
 弁理士 中村 誠
 (74) 代理人 100070437
 弁理士 河井 将次

最終頁に続く

(54) 【発明の名称】動画像再生方法および動画像再生装置

(57) 【特許請求の範囲】

【請求項 1】

符号化された動画像信号を復号および再生するための動画像再生方法であって、
 前記符号化された動画像信号を復号する復号処理を実行する復号ステップと、
 前記復号ステップによって復号された動画像信号に対してフィルタ処理を施すフィルタ
 処理ステップと、

前記フィルタ処理ステップによってフィルタ処理された動画像信号を再生出力するステ
 ップと、

前記動画像信号の再生処理の遅れを検出する遅れ検出ステップと、

前記検出された遅れが所定の第1の遅れ量よりも大きい場合、前記フィルタ処理ステッ
 プによるフィルタ処理の実行をスキップし、前記検出された遅れが前記所定の第1の遅
 量よりも大きい所定の第2の遅れ量よりも大きい場合、前記復号ステップによる復号処理
 の実行をスキップしてフレームスキップを行う再生制御ステップとを具備することを特徴
 とする動画像再生方法。

【請求項 2】

前記遅れ検出ステップは、前記再生出力ステップで再生中のフレームのタイムスタンプ
 の値と前記動画像信号の再生開始からの経過時間とを比較して、前記再生出力ステップに
 おける前記動画像信号の再生処理の遅れを検出することを特徴とする請求項1記載の動
 画像再生方法。

【請求項 3】

10

符号化された動画像信号を復号および再生するための動画像再生装置であって、
前記符号化された動画像信号を復号する復号処理を実行する復号手段と、
前記復号手段によって復号された動画像信号に対してフィルタ処理を施すフィルタ処理手段と、

前記フィルタ処理手段によってフィルタ処理された動画像信号を再生出力する再生出力手段と、

前記動画像信号の再生処理の遅れを検出する遅れ検出手段と、

前記検出された遅れが所定の第1の遅れ量よりも大きい場合、前記フィルタ処理手段によるフィルタ処理の実行をスキップし、前記検出された遅れが前記所定の第1の遅れ量よりも大きい所定の第2の遅れ量よりも大きい場合、前記復号手段による復号処理の実行をスキップしてフレームスキップを行う再生制御手段とを具備することを特徴とする動画像再生装置。

10

【請求項4】

前記遅れ検出手段は、前記再生出力手段で再生中のフレームのタイムスタンプの値と前記動画像信号の再生開始からの経過時間とを比較して、前記再生出力手段における前記動画像信号の再生処理の遅れを検出することを特徴とする請求項3記載の動画像再生装置。

【請求項5】

符号化された動画像信号を復号および再生するためのプログラムが格納されたコンピュータ読み取り可能な記録媒体であって、

前記符号化された動画像信号を復号する復号処理を実行する復号手順と、

20

前記復号ステップによって復号された動画像信号に対してフィルタ処理を施すフィルタ処理手順と、

前記フィルタ処理手順によってフィルタ処理された動画像信号を再生出力する手順と、
前記動画像信号の再生処理の遅れを検出する遅れ検出手順と、

前記検出された遅れが所定の第1の遅れ量よりも大きい場合、前記フィルタ処理手順によるフィルタ処理の実行をスキップし、前記検出された遅れが前記所定の第1の遅れ量よりも大きい所定の第2の遅れ量よりも大きい場合、前記復号手順による復号処理の実行をスキップしてフレームスキップを行う再生制御手順とをコンピュータに実行させるプログラムが格納された記録媒体。

30

【請求項6】

前記遅れ検出手順は、前記再生出力手順で再生中のフレームのタイムスタンプの値と前記動画像信号の再生開始からの経過時間とを比較して、前記再生出力手順における前記動画像信号の再生処理の遅れを検出することを特徴とする請求項5記載の記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は符号化された動画像信号を復号および再生するための動画像再生方法および動画再生装置に関し、特に滑らかな動画再生が行えるように改善された動画像再生方法および動画再生装置に関する。

40

【0002】

【従来の技術】

近年、コンピュータ技術の発達に伴い、デジタルビデオプレーヤ、セットトップボックス、デジタルTV、デジタルVCR、パソコンコンピュータ等のマルチメディア対応の各種電子機器が開発されている。この種の電子機器では、MPEG2/MPEG4などの動画像高能率符号化方式で符号化された動画データを復号・再生するためのデコーダが設かれている。

【0003】

この種のデコーダにおいては、動画像信号を復号した後に、その復号画像に対してポストフィルタと称される画質改善のためのフィルタ処理が施されるのが通常である。MPEG2/MPEG4などに代表されるように、多くの符号化方式では画像をブロック単位で処

50

理するため、その復号画像にはブロックノイズやエッジノイズといった劣化が生じる。ブロックノイズとは、本来の画像の絵柄にはないブロックパターン状の幾何学模様が見える歪みである。また、エッジノイズはエッジ付近に生じるリンクギング状の歪み（モスキートノイズともいう）である。

【0004】

このような符号化方式固有のノイズを除去するのが、ポストフィルタである。ポストフィルタは復号画像信号を平滑化して、ブロック間の境界部における高域成分を取り除く等の処理を行う。このようなポストフィルタを動画像復号器後段に配したものとしては、たとえば特開昭64-55987号公報が知られている。この特開昭64-55987号公報には、ブロック単位処理を基本とする動画像復号器から出力された復号動画像信号に対して、動ブロック領域と静ブロック領域の判別を行い、その結果に応じて平滑化フィルタにおける平滑化の度合いを切り替え、処理を効率化する技術が開示されている。10

【0005】

【発明が解決しようとする課題】

しかし、このようなポストフィルタによるフィルタ処理は比較的多くの演算量を必要とするため、フィルタ処理に時間がかかり、結果的に再生処理の遅れが引き起こされる場合がある。特に、マイクロプロセッサベースでソフトウェアデコードを行うシステムでは、動きの激しい場面においては復号処理そのものにプロセッサ資源が占有されてしまうので、それによってプロセッサ負荷が高くなり、再生処理の遅れが多々引き起こされることになる。再生処理の遅れは、音声と同期して動画像の復号・再生をソフトウェアによって行う場合に特に顕在化され、音声とのずれや、コマ落ちなどの問題が生じることになる。20

【0006】

すなわち、動画像の再生処理に遅れが生じた場合には、通常、その遅れを解消するために、いくつかのフレームに関する復号処理を省くというフレームスキップが行われる。これにより、コマ落ちが発生し、ギクシャクとした再生画像となってしまう。

【0007】

本発明は上述の事情に鑑みてなされたものであり、フレームスキップの発生を極力排除できるようにし、滑らかな動画再生を行うことが可能な動画再生方法および動画再生装置を提供することを目的とする。

【0008】

【課題を解決するための手段】

上述の課題を解決するため、本発明は、符号化された動画像信号を復号および再生するための動画像再生方法であって、前記符号化された動画像信号を復号する復号ステップと、前記復号ステップによって復号された動画像信号に対してフィルタ処理を施すフィルタ処理ステップと、前記フィルタ処理ステップによってフィルタ処理された動画像信号を再生出力するステップと、前記動画像信号の再生状況に基づき、前記フィルタ処理ステップによるフィルタ処理を制御する再生制御ステップとを具備することを特徴とする。30

【0009】

この動画再生方法においては、動画像信号の再生状況に基づいて、フィルタ処理の制御が行われ、現在の再生状況に最適なフィルタ処理を行うことができる。このように、画質調整のためのフィルタ処理の内容を最適化することにより、フレームスキップなどの処理を行わずとも、プロセッサの負荷状況に合わせて演算処理量を調整することができるので、滑らかな動画再生を実現することが可能となる。40

【0010】

動画像信号の再生状況を表す指標としては、再生処理の遅れの度合いを用いることが好ましい。再生処理の遅れが検出された場合には、フィルタ処理の実行をスキップしたり、フィルタ処理の種類を処理量の少ない簡便なものに切り換えるという制御を行うことにより、再生処理の遅れを回復することが可能となる。これにより、フレームスキップによるコマ落ちの発生を防止できる。よって、プロセッサ負荷が増大する動きの激しいシーンなどにおいても滑らかな動画再生を実現することができる。なお、フィルタ処理をスキップし50

たり、フィルタ処理の種類を切り換えることにより、画質改善効果については低下するものの、一般に、それによって視覚的に認識される画質低下の度合いは、コマ落ちの発生に比べ非常に小さい。

【0011】

また、再生処理の遅れ量を、状態変数などを用いて管理することにより、遅れの度合いに応じてフィルタ処理の種類を処理量の少ないものに段階的に切り替えていき、最終的にフィルタ処理の実行をスキップするという多段階制御を利用することにより、画質改善効果の低下を最小限に抑えることができる。

【0012】

また、上述のフィルタ処理の制御とフレームスキップ処理とを組み合わせて、再生処理の遅れの度合いがある一定の範囲内のうちにはフィルタ処理の制御のみを行い、遅れ量が所定の値に達した場合に初めて復号ステップの実行を所定フレーム数分スキップさせるように制御することにより、滑らかな動画再生と、破綻のない正常な再生処理を両立することができる。

10

【0013】

【発明の実施の形態】

以下、図面を参照して本発明の実施形態を説明する。

図1には、本発明の一実施形態に係る動画再生装置の一例として、パーソナルコンピュータを用いた場合の構成が示されている。このパーソナルコンピュータはノートブックタイプの携帯型のコンピュータシステムであり、MPEG2/MPEG4等によって圧縮符号化されたオーディオ・ビデオストリームをソフトウェアによって復号・再生することができる。

20

【0014】

図1のコンピュータシステムにおいては、図示のように、CPU11、ホスト-PCIブリッジ12、主メモリ13、表示コントローラ14、サウンドコントローラ15、通信インターフェース16、I/Oコントーラ17、PCI-ISAブリッジ18、カメラ20、ハードディスクドライブ(HDD)21、DVDドライブ22などが設けられている。

【0015】

CPU11は本システム全体を制御するためのものであり、主メモリ13にロードされたオペレーティングシステムや他の各種プログラムを実行する。本実施形態においては、オーディオ・ビデオストリームの復号・再生のためのプログラムとして、動画再生ソフトウェア100が用いられる。この動画再生ソフトウェア100は、MPEG2、MPEG4、AVI、DVIなどの各種圧縮形式のAV(オーディオ・ビデオ)データに対応しており、AVデータの符号化ファイルを読み込み、それを復号・再生することができる。

30

【0016】

ホスト-PCIブリッジ12はCPUバス1とPCIバス2を接続するバスブリッジであり、ここには主メモリ13を制御するためのメモリコントロールロジックも内蔵されている。表示コントローラ14は本コンピュータシステムのディスプレイモニタとして使用されるLCDや外部CRTディスプレイを制御する。AVデータの再生時には、動画再生ソフトウェア100によって得られた復号画像信号が表示コントローラ14を通じてディスプレイモニタに表示されることになる。

40

【0017】

サウンドコントローラ15は音源として用いられるものであり、各種オーディオデータの入出力を行う。AVデータの再生時には、動画再生ソフトウェア100によって得られた復号音声信号がサウンドコントローラ15を通じてスピーカから再生、あるいはラインアウト端子から外部オーディオ機器に出力される。

【0018】

通信インターフェース16は、たとえばUSBやIEEE1394などのシリアルインターフェース規格で外部または内蔵のカメラ20と通信するためのものであり、カメラ20からビデオデータを取り込むことができる。カメラ20によって撮影された動画像信号は

50

、そのまま再生表示したり、あるいはMPEG2、MPEG4、AVI、DIVXなどの各種圧縮形式に変換した後にI/Oコントローラ17を介してHDD21、DVDドライブ22、メモリカード23などの各種記録メディアに記録することができる。

【0019】

PCI - ISAブリッジ18はPCIバス2とISAバス3との間を接続するバスブリッジであり、ここにはリアルタイムクロック(RTC)181などの各種システムデバイスが内蔵されている。リアルタイムクロック(RTC)181は時計モジュールであり、本実施形態では、このリアルタイムクロック(RTC)181から取得した時刻を用いてAVデータの再生処理の遅れを管理している。

【0020】

(動画再生ソフト)

ここで、図2および図3を参照して、動画再生ソフト100の基本機能について説明する。

【0021】

前述したように、動画再生ソフトウェア100は、HDD21、DVDドライブ22、またはメモリカード23などの蓄積メディアに記録されているAVデータの符号化ファイルを読み込み、それを復号・再生することができる。図2には、HDD21に記録されているAV符号化ファイル210を復号・再生する場合の様子が示されている。

【0022】

AV符号化ファイル210には、動画像信号とオーディオ信号をそれぞれデジタル圧縮符号化した後にそれらの符号化ビットストリームを多重化することによって生成されたものである。動画再生ソフト100は、HDD21からAV符号化ファイル210を読み取り、動画像信号とオーディオ信号とに分離した後にそれぞれを伸張するための復号処理を行い、そして動画像をディスプレイモニタに表示すると共に、オーディオ信号をスピーカなどから再生する。

【0023】

動画再生ソフトウェア100は、図3に示されているように、アプリケーションプログラム101と再生エンジン102とから構成されている。アプリケーションプログラム101は動画再生処理のためのユーザインターフェースと再生エンジン102を制御するためのインターフェースとを有しており、ユーザから指定されたAV符号化ファイル210の再生に必要な動作を、再生エンジン102に対して指示する。再生エンジン102は、オペレーティングシステム(OS)103上に実装されたマルチメディア処理用のプラットホームであり、ファイル入出力処理や動画／音声再生のためのレンダリング処理をはじめとする、動画・音声再生のために必要な処理を行う種々のプログラムモジュール(フィルタ)群から構成されている。これらモジュールはアプリケーションプログラム101からの指示により任意に組み合わせて用いることができる。

【0024】

(再生エンジン)

次に、再生エンジン102の機能構成を説明する。図4は、AV符号化ファイルの再生時におけるモジュール構成を示している。

【0025】

AV符号化ファイルに含まれる動画・オーディオの再生は、マネージャ301、ファイルリーダ302、デマルチブレクサ(DMUX)303、ビデオデコーダ304、ビデオレンダラ305、オーディオデコーダ306、オーディオレンダラ307を図示のようにリンクさせることによって実現される。

【0026】

まず、再生対象の符号化ファイルがファイルリーダ302によってリードされ、そこに含まれる符号化ビットストリームがデマルチブレクサ(DMUX)303に送られ、そこで動画データとオーディオデータとに分離される。

【0027】

10

20

30

40

50

動画データについては、デマルチプレクサ(D M U X) 3 0 3 内でフレーム単位に分離され、1フレームずつビデオデコーダ3 0 4 に送られる。ビデオデコーダ3 0 4 は動画データを復号・再生するためのモジュールであり、ここには画像伸張処理機能および画質改善のためのフィルタ処理機能などが含まれている。ビデオデコーダ3 0 4 によって得られた復号画像はビデオレンダラ3 0 5 を介して表示コントローラ1 4 に送られ、モニタ上に画面表示される。一方、オーディオデータはオーディオデコーダ3 0 6 によって復号された後に、オーディオレンダラ3 0 7 を介してサウンドコントローラ1 5 に送られ、スピーカなどから再生出力される。

【 0 0 2 8 】

マネージャ3 0 1 は動画の再生状況を管理するためのものであり、再生処理に遅れが生じたときにそれを最下流のビデオレンダラ3 0 5 に通知する(再生遅れ通知)。符号化された動画データには、図5に示すようにフレーム毎にタイムスタンプT S が埋め込まれているため、マネージャ3 0 1 は、再生開始からの経過時間と再生中のフレームのタイムスタンプとを比較することにより、再生処理速度を監視することができる。マネージャ3 0 1 からの再生処理の遅れ通知は、ビデオレンダラ3 0 5 から順に上流側のモジュールへと伝達される。つまり、ビデオデコーダ3 0 4 による復号・再生処理の状況が、ビデオレンダラ3 0 5 を介してビデオデコーダ3 0 4 にフィードバックされることになる。

【 0 0 2 9 】

(再生遅れ通知)

図6には、再生遅れ通知を発行するために必要なマネージャ3 0 1 の処理手順が示されている。マネージャ3 0 1 は、動画再生の開始時にR T C 1 8 1 からそのときの時刻(スタートタイム)を取得する(ステップS 1 1)。そして、定期的にR T C 1 8 1 から現在時刻(カレントタイム)を取得し(ステップS 1 2)、再生開始からの経過時間と再生中のフレームのタイムスタンプとを比較することによって、遅れ量を算出する(ステップS 1 3)。再生中フレームのタイムスタンプはビデオレンダラ3 0 5 から取得してもよいが、再生中フレームの何フレームか後のフレームのタイムスタンプをデマルチプレクサ(D M U X) 3 0 3 から取得し、それを再生中フレームのタイムスタンプの値に換算してもよい。A V I フォーマットの符号化データの場合には、たとえばM P E G 2 / M P E G 4 等で圧縮符号化されたストリームに対してタイムスタンプを含むヘッダが挿入されるので、M P E G 2 / M P E G 4 固有のタイムスタンプではなく、A V I フォーマットのタイムスタンプを利用することもできる。

【 0 0 3 0 】

算出された遅れ量は、再生速度などの値に換算されてビデオレンダラ3 0 5 に通知される(ステップS 1 4)。この場合、再生速度は1 0 0 %を正常速度として表し、遅れ量の度合いに応じて再生速度を示す%の値は低下する。

【 0 0 3 1 】

(ビデオデコーダ)

次に、図7を参照して、本実施形態の特徴とするビデオデコーダ3 0 4 の構成を説明する。

ビデオデコーダ3 0 4 は、前述したように再生エンジン1 0 2 内の1モジュールとして動作するものであり、図示のように、画像伸張部5 0 1 、画質改善部5 0 2 、画像出力部5 0 3 、遅れ検出部5 0 4 、状態変数5 0 5 、およびコントロール部5 0 6 から構成されている。符号化された動画像信号の復号・再生を行うための復号処理部は、画像伸張部5 0 1 、画質改善部5 0 2 、および画像出力部5 0 3 から構成されている。

【 0 0 3 2 】

画像伸張部5 0 1 は圧縮動画像を復号して伸張するためのものであり、M P E G 2 / M P E G 4 で圧縮された動画像信号については、可変長復号処理、逆量子化、逆D C T 、動き補償予測、差分復号画像への予測画像の加算などの処理を行う。画像伸張部5 0 1 によって1フレーム分の復号画像が得られると、それが画質改善部5 0 2 に送られる。

【 0 0 3 3 】

10

20

30

40

50

画質改善部 502 は復号画像の画質改善のためにブロックノイズ削減等のための平滑化フィルタ処理を行うためのものポストフィルタであり、フィルタの種類（IIR フィルタ、FIR フィルタ）の切り換え、およびフィルタパラメータの設定により、処理速度やフィルタ強度の異なる様々なフィルタ処理を復号画像信号に対して施すことができる。なお、ここで平滑化フィルタ処理は画質調整のための演算処理の一例であり、それ以外の様々な画質調整処理を含めて以下ではフィルタ処理と称することにする。

【0034】

この画質改善部 502 によるフィルタ処理の内容はコントロール部 506 によって制御される。

【0035】

画像出力部 503 は、画質改善部 502 によって処理された復号画像を前述のビデオレンダラ 305 に出力する。遅れ検出部 504 はビデオレンダラ 305 を通じてマネージャ 301 から伝達される遅れ通知に従って動画像信号の再生処理の遅れを検出するためのものであり、状態変数 505 を用いて再生処理の遅れの度合いを現在の再生状況として管理する。

10

【0036】

コントロール部 506 は、画像伸張部 501、画質改善部 502、および画像出力部 503 から構成される復号処理部の動作を制御するためのものであり、画質改善部 502 によるフィルタ処理の最適化制御を行う。具体的には、コントロール部 506 は、画像伸張部 501 による復号処理のスキップ（フレームスキップ）を極力排除した状態で、再生処理の遅れを回復させるための制御を行うために、状態変数 505 の値に応じてフィルタ処理の実行をスキップしたり、あるいはフィルタ処理の種類を処理量の少ない簡便なものに切り換えるなどの遅れ回復制御を行う。以下、遅れ回復制御処理の具体的な手順について説明する。

20

【0037】

（遅れ回復制御 #1）

まず、図 8 および図 9 を参照して、遅れ回復制御処理の第 1 の例について説明する。本例では、2 値の状態変数 505（“0”または“1”）によって遅れ回復が必要な遅れであるか否かを管理し、遅れ回復が必要な遅れである場合には（状態変数 = “1”）、画質改善のためのフィルタ処理をスキップさせるという処理が行われる。

30

【0038】

図 8 は遅れ検出部 504 による遅れ検出処理を示したフローチャートである。図 8 に示されているように、遅れ検出部 504 は、ビデオレンダラ 305 を通じてマネージャ 301 からの再生遅れ通知を受けると、まず、その再生遅れ通知によって遅れ量として指定される再生速度（%）の値に基づき、再生処理の遅れがあらかじめ決められた所定量よりも大きいか否かを判定する（ステップ S311）。再生処理の遅れが所定量以下であれば（ステップ S311 の NO）、なにもせずに処理を終了する。一方、再生処理の遅れが所定量よりも大きい場合には（ステップ S311 の YES）、状態変数 505 の現在の値をチェックする（ステップ S312）。

40

【0039】

状態変数 = “0”であれば（ステップ S312 の YES）、遅れ検出部 504 は、遅れ回復処理が必要であることをコントロール部 506 に通知するために、状態変数の値を“1”に変更し（ステップ S313）、状態変数がすでに“1”であれば（ステップ S312 の NO）、なにもせずに処理を終了する。

【0040】

図 9 はコントロール部 506 によって各フレーム毎に行われる復号制御処理の手順を示すフローチャートである。まず、画像伸張部 501 によって該当するフレームの復号処理が実行される（ステップ S351）。1 フレーム分の復号処理が終了した時点で、コントロール部 506 は、状態変数 505 をチェックする（ステップ S352）。状態変数が“0”であれば（ステップ S352 の NO）、コントロール部 506 は、通常通り、復号され

50

た1フレーム画像を画質改善部502に渡し、画質改善のための平滑化フィルタ処理を実行させる(ステップS353)。そして、フィルタ処理を受けた画像信号が画像出力部503から出力される(ステップS354)。一方、状態変数が“1”であれば(ステップS352のYES)、コントロール部506は、画質改善部502による画質改善のためのフィルタ処理をスキップさせ、フィルタ処理を実行せずに、復号画像をそのまま画像出力部503から出力させる(ステップS354)。

【0041】

このように、CPU負荷が高く、再生処理速度に遅れが生じた場合でも、復号処理自体は間引くことなく、フィルタ処理の最適化制御を行うことにより、こま落ちのない滑らかな動画再生が可能となる。

10

【0042】

(遅れ回復制御#2)

次に、図10および図11を参照して、遅れ回復制御処理の第2の例について説明する。本例では、“0”, “1”, “2”, “3”的4値の状態変数505によって再生処理の遅れ量の度合いを管理し、それに応じて段階的にフィルタ処理の内容を変更するという処理が行われる。

20

【0043】

図10は遅れ検出部504による遅れ検出処理を示したフローチャートである。図10に示されているように、遅れ検出部504は、ビデオレンダラ305を通じてマネージャ301からの再生遅れ通知を受けると、まず、その再生遅れ通知によって遅れ量として指定される再生速度(%)の値に基づき、再生処理の遅れがあらかじめ決められた所定量よりも大きいか否かを判定する(ステップS411)。再生処理の遅れが所定量以下であれば(ステップS411のNO)、なにもせずに処理を終了する。一方、再生処理の遅れが所定量よりも大きい場合には(ステップS411のYES)、状態変数505の現在の値をチェックする(ステップS412)。

【0044】

状態変数が最高値“3”よりも小さい場合には(ステップS412のYES)、遅れ検出部504は、遅れ量の度合いを示す状態変数の値を+1増加させ(ステップS413)、状態変数がすでに最高値“3”であれば(ステップS412のNO)、なにもせずに処理を終了する。

30

【0045】

図11はコントロール部506によって各フレーム毎に行われる復号制御処理の手順を示すフローチャートである。まず、画像伸張部501によって該当するフレームの復号処理が実行される(ステップS451)。1フレーム分の復号処理が終了した時点で、コントロール部506は、状態変数505をチェックし(ステップS452)、その状態変数の値に応じてフィルタの内容を切り換える。本例では、演算処理量の異なる3つのフィルタ処理A, B, Cがあらかじめ用意されている。フィルタ処理Aはもっとも画質改善に効果のあるフィルタ処理であるが、演算処理に時間を要する。次に、画質改善効果が期待できるのがフィルタ処理Bで、最後がフィルタ処理Cである。CPU11の負荷はフィルタ処理A, B, Cの順で少なくなる。

40

【0046】

状態変数が“0”であれば、コントロール部506は、復号された1フレーム画像を画質改善部502に渡し、フィルタ処理Aを実行させる(ステップS353)。そして、フィルタ処理を受けた画像信号が画像出力部503から出力される(ステップS456)。同様に、状態変数が“1”であればフィルタ処理Bを、状態変数が“2”であればフィルタ処理Cを実行させる(ステップS454, S455)。一方、状態変数が最高値“3”であれば、コントロール部506は、画質改善部502による画質改善のためのフィルタ処理をスキップさせ、フィルタ処理を実行せずに、復号画像をそのまま画像出力部503から出力させる(ステップS456)。

【0047】

50

このように、遅れの度合いに応じて徐々にフィルタ処理の内容を粗くするという制御を行うことにより、急激な画質変化をもたらすことなく、こま落ちのない滑らかな動画再生を行うことが可能となる。

【0048】

(遅れ回復制御 #3)

次に、図12および図13を参照して、遅れ回復制御処理の第3の例について説明する。本例は、“0”, “1”, “2”の3値の状態変数S05によって再生処理の遅れ量の度合いを管理し、フィルタ処理の実行の有無の切り替えと、フレームスキップとを組み合わせて行う場合の例である。

【0049】

図12は遅れ検出部S04による遅れ検出処理を示したフローチャートである。図12に示されているように、遅れ検出部S04は、ビデオレンダラ305を通じてマネージャ301からの再生遅れ通知を受けると、まず、その再生遅れ通知によって遅れ量として指定される再生速度(%)の値に基づき、再生処理の遅れがあらかじめ決められた所定量よりも大きいか否かを判定する(ステップS511)。再生処理の遅れが所定量以下であれば(ステップS511のNO)、なにもせずに処理を終了する。一方、再生処理の遅れが所定量よりも大きい場合には(ステップS511のYES)、状態変数S05の現在の値をチェックする(ステップS512)。

【0050】

状態変数が最高値“2”よりも小さい場合には(ステップS512のYES)、遅れ検出部S04は、遅れ量の度合いを示す状態変数の値を+1増加させ(ステップS513)、状態変数がすでに最高値“2”であれば(ステップS512のNO)、なにもせずに処理を終了する。

【0051】

図13はコントロール部S06によって各フレーム毎に行われる復号制御処理の手順を示すフローチャートである。コントロール部S06は、まず、画像伸張部S01によるフレームの復号処理の実行に先立ち、状態変数S05をチェックする(ステップS551)。状態変数S05が最高値“2”であれば(ステップS511のYES)、復号処理、フィルタ処理、画像出力処理をすべてスキップし、これによりフレームスキップを行う。

【0052】

一方、状態変数S05が“1”以下であれば(ステップS511のNO)、コントロール部S06は、画像伸張部S01に該当するフレームの復号処理を実行させた後に(ステップS552)、状態変数S05が“0”であるか“1”であるかを判断する(ステップS553)。状態変数が“0”であれば(ステップS553のNO)、コントロール部S06は、通常通り、復号された1フレーム画像を画質改善部S02に渡し、画質改善のための平滑化フィルタ処理を実行させる(ステップS554)。そして、フィルタ処理を受けた画像信号が画像出力部S03から出力される(ステップS555)。状態変数が“1”であれば(ステップS553のYES)、コントロール部S06は、画質改善部S02による画質改善のためのフィルタ処理をスキップさせ、フィルタ処理を実行せずに、復号画像をそのまま画像出力部S03から出力させる(ステップS555)。

【0053】

このように、再生遅れ量がある一定の範囲内にある場合にはフィルタ処理のスキップのみを行い、ある一定の範囲以上になったときに初めてフレームスキップを実行するという制御を行うことにより、滑らかな動画再生と、破綻のない再生処理との両立を図ることが可能となる。なお、この制御は、前述の第2の例においても適用することができる。この場合、再生遅れ量がある一定の範囲内にある場合にはフィルタ処理の段階的な制御が行われ、ある一定の範囲以上になったときにフレームスキップが実行されることになる。

【0054】

以上のように、本実施形態においては、CPU負荷が大きくなり再生処理に遅れが生じた場合にはフィルタ処理の内容を制御することによってCPU負荷を低減するという制御を

10

20

30

40

50

行っているので、CPUの負荷状況に合わせて画質調整処理の内容を最適化できるようになり、ソフトウェアデコードによって音声と同期して動画の復号・再生を行う場合においても、フレームスキップが発生する頻度を極力抑えることが可能となる。

【0055】

また、本実施形態では、再生中のフレームのタイムスタンプと再生開始からの経過時間との比較によって再生遅れ通知を生成する場合を例示したが、たとえばレート制御されたバッファの占有量などを用いて再生遅れの度合いを検出し、それに応じてフィルタ処理の内容を制御することも可能である。

【0056】

また、本実施形態では、状態変数を低下させる処理については特に説明しなかったが、状態変数の値は再生遅れが少なくなったときなどに低下させればよいことはもちろんである。
10

【0057】

さらに、本実施形態のソフトウェアデコードの方法は、その手順を含むコンピュータプログラムをコンピュータ読み取り可能な記録媒体を通じて通常のコンピュータに導入するだけで、容易に実現することができる。また、コンピュータのみならず、ゲーム機や、デジタルTV、セットトップボックスなどにも適用することができる。

【0058】

【発明の効果】

以上説明したように、本発明によれば、復号画像の画質改善ためのフィルタ処理を制御することにより、フレームスキップの発生を極力排除できるようになり、滑らかな動画再生を行うことが可能となる。
20

【図面の簡単な説明】

【図1】本発明の一実施形態に係るコンピュータシステムの構成を示すブロック図。

【図2】同実施形態で用いられる動画再生ソフトの基本機能を説明するための図。

【図3】同実施形態で用いられる動画再生ソフトの構造を示す図。

【図4】同実施形態で用いられる動画再生ソフトの再生エンジン部の機能構成を示す図。

【図5】同実施形態における動画像信号のストリーム構造の一例を示す図。

【図6】同実施形態で用いられる再生遅れ通知処理の手順を示すフローチャート。

【図7】同実施形態の再生エンジン部に設けられるビデオデコーダの機能構成を示す図。
30

【図8】同実施形態で用いられる遅れ検出処理の第1の例を示すフローチャート。

【図9】同実施形態で用いられる復号制御処理の第1の例を示すフローチャート。

【図10】同実施形態で用いられる遅れ検出処理の第2の例を示すフローチャート。

【図11】同実施形態で用いられる復号制御処理の第2の例を示すフローチャート。

【図12】同実施形態で用いられる遅れ検出処理の第3の例を示すフローチャート。

【図13】同実施形態で用いられる復号制御処理の第3の例を示すフローチャート。

【符号の説明】

1 1 ... CPU

1 0 0 ... 動画再生ソフト

1 0 1 ... 動画再生用アプリケーションプログラム

40

1 0 2 ... 再生エンジン

3 0 1 ... マネージャ

3 0 2 ... ファイルリーダ

3 0 3 ... デマルチプレクサ (DMUX)

3 0 4 ... ビデオデコーダ

3 0 5 ... ビデオレンダラ

3 0 6 ... オーディオデコーダ

3 0 7 ... オーディオレンダラ

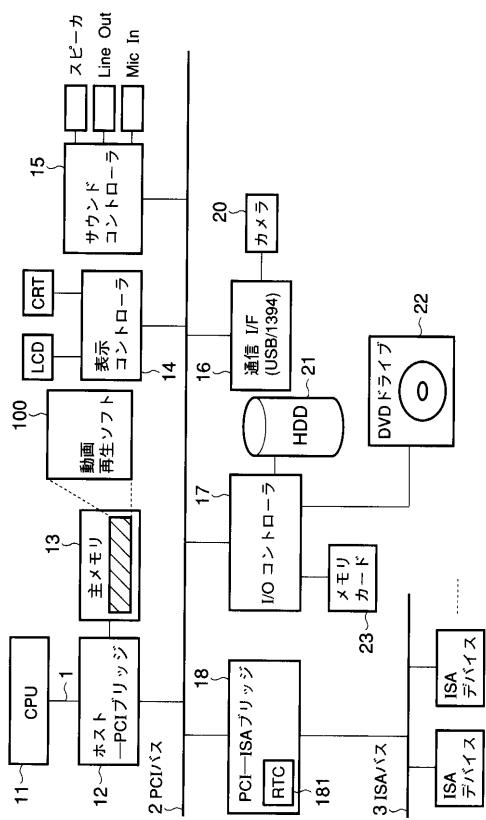
5 0 1 ... 画像伸張部

5 0 2 ... 画質改善部

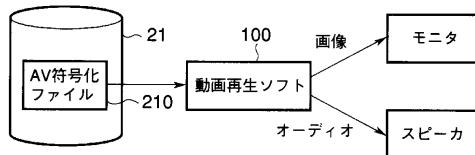
50

- 5 0 3 ... 画像出力部
 5 0 4 ... 遅れ検出部
 5 0 5 ... 状態変数
 5 0 6 ... コントロール部

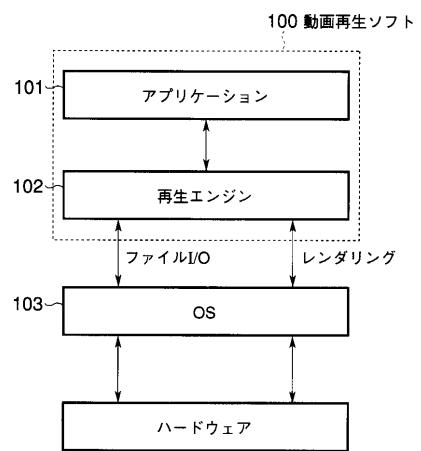
【図 1】



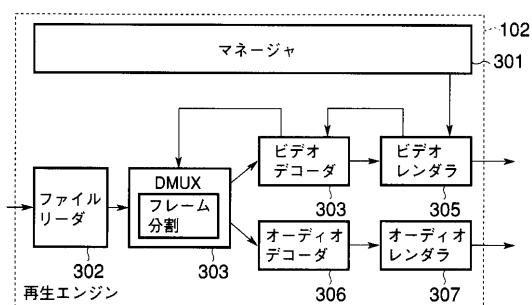
【図 2】



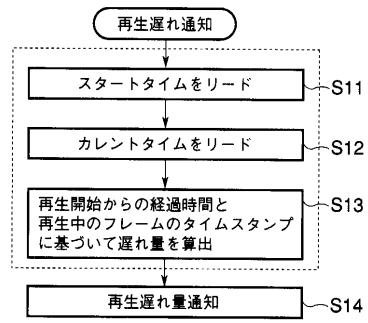
【図 3】



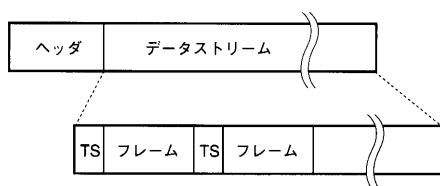
【図4】



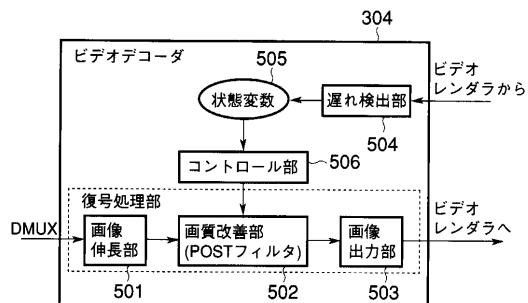
【図6】



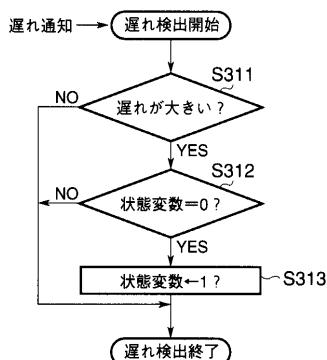
【図5】



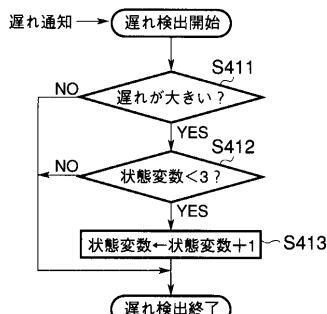
【図7】



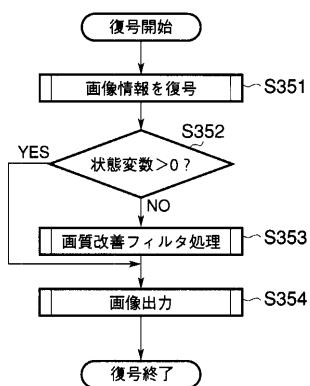
【図8】



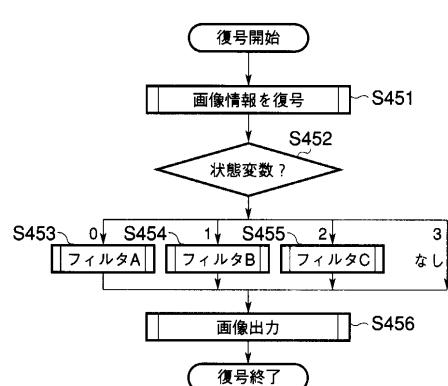
【図10】



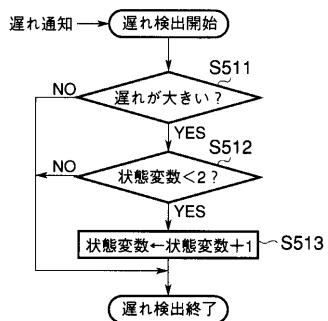
【図9】



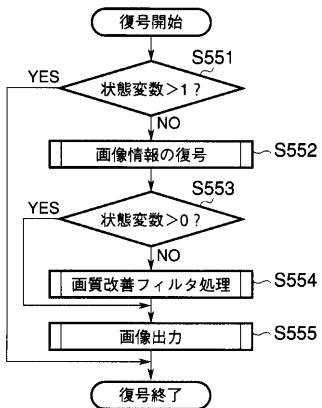
【図11】



【図12】



【図13】



フロントページの続き

(72)発明者 梅田 茂之

東京都青梅市末広町2丁目9番地 株式会社東芝青梅工場内

審査官 菅原 道晴

(56)参考文献 特開平10-210483(JP,A)

特開平11-122624(JP,A)

特開平07-231449(JP,A)

特開平08-149415(JP,A)

(58)調査した分野(Int.Cl., DB名)

H04N 7/26-7/68