

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4875204号
(P4875204)

(45) 発行日 平成24年2月15日 (2012. 2. 15)

(24) 登録日 平成23年12月2日 (2011.12.2)

(51) Int.Cl.		F I		
G 1 0 L 19/00	(2006.01)	G 1 0 L 19/00	3 3 0 Z	
G 1 0 L 19/02	(2006.01)	G 1 0 L 19/02	1 8 0 A	
		G 1 0 L 19/02	1 7 0 Z	

請求項の数 10 (全 13 頁)

(21) 出願番号	特願2010-506343 (P2010-506343)	(73) 特許権者	502087507
(86) (22) 出願日	平成20年1月31日 (2008. 1. 31)		ソニー エリクソン モバイル コミュニ
(65) 公表番号	特表2010-525414 (P2010-525414A)		ケーションズ, エービー
(43) 公表日	平成22年7月22日 (2010. 7. 22)		スウェーデン国, 2 2 1 8 8 ルンド
(86) 国際出願番号	PCT/US2008/052581	(74) 代理人	100076428
(87) 国際公開番号	W02008/134103		弁理士 大塚 康徳
(87) 国際公開日	平成20年11月6日 (2008. 11. 6)	(74) 代理人	100112508
審査請求日	平成21年12月25日 (2009. 12. 25)		弁理士 高柳 司郎
(31) 優先権主張番号	11/741, 297	(74) 代理人	100115071
(32) 優先日	平成19年4月27日 (2007. 4. 27)		弁理士 大塚 康弘
(33) 優先権主張国	米国 (US)	(74) 代理人	100116894
			弁理士 木村 秀二

最終頁に続く

(54) 【発明の名称】 符号化オーディオデータを処理するための装置及び方法

(57) 【特許請求の範囲】

【請求項 1】

データストリーム(70)における複数の符号化オーディオフレーム(72)であって各フレームがヘッダ(80)を有する符号化オーディオフレーム(72)を復号化する方法であって、

同期語(62)と、有効な符号化オーディオフレーム(72)のヘッダ(80)フィールドに対する少なくとも1つの予測値に対応する1つ以上の付加ビット(64)とを含むマッチングパタン(60)を生成するステップと、

前記マッチングパタン(60)のインスタスがないかデータストリーム(70)の部分を検索することによりフレーム境界(74)を検出するステップと、

前記検出されたフレーム境界(74)に対応する前記データストリーム(70)における点から始まる1つ以上の符号化オーディオフレーム(72)を復号化するステップと、
を有することを特徴とする方法。

【請求項 2】

前記フレーム境界(74)を検出するステップは、前記マッチングパタン(60)の所定数のインスタスを検索するステップを含み、前記検出されたフレーム境界(74)は、前記所定数のインスタスのうちの最後のものに対応することを特徴とする請求項1に記載の方法。

【請求項 3】

停止信号を受信するステップを更に有し、

前記フレーム境界(74)を検出するステップは、前記停止信号が受信されるまで、前記マッチングパタン(60)のインスタスがないか前記データストリーム(70)の部分を検索するステップを含むことを特徴とする請求項1に記載の方法。

【請求項4】

前記符号化オーディオフレーム(72)は、Advanced Audio Codec Rawデータブロックを含み、前記フレームヘッダ(80)は、Audio Data Transport Stream (ADTS)ヘッダを含み、前記マッチングパタン(60)は、12ビット同期語(62)と、1ビットIDフィールド(84)、2ビットレイヤフィールド(86)、1ビット保護なしフィールド(88)の予測値に対応する付加ビット(64)とを含むことを特徴とする請求項1に記載の方法。

10

【請求項5】

オーディオ処理誤りを検出するステップと、
前記オーディオ処理誤りに対応する前記データストリーム(70)における誤り位置を特定するステップと、を更に有し、
前記マッチングパタン(60)のインスタスがないか前記データストリーム(70)の部分を検索するステップは、前記誤り位置から開始することを特徴とする請求項1に記載の方法。

【請求項6】

前記フレーム境界(74)を検出するステップは、CRC(cyclical redundancy checksum)ビットを評価して前記検出されたフレーム境界(74)が有効なヘッダ(80)に対応しているかを確認することにより、前記検出されたフレーム境界(74)が有効なヘッダ(80)に対応しているかを検証するステップを含むことを特徴とする請求項1に記載の方法。

20

【請求項7】

データストリーム(70)における符号化オーディオフレーム(72)を復号化するオーディオデコーダ(50)であって、
同期語(62)と、有効な符号化オーディオフレーム(72)のヘッダ(80)フィールドに対する少なくとも1つの予測値に対応する1つ以上の付加ビット(64)とを含むマッチングパタン(60)を生成するマッチングパタン生成部(54)と、
前記マッチングパタン(60)のインスタスがないかデータストリーム(70)の部分を検索してフレーム境界(74)を検出するフレーム境界検出部(56)と、
前記検出されたフレーム境界(74)に対応する前記データストリーム(70)における点から始まる1つ以上の符号化オーディオフレーム(72)を復号化するフレームデコーダ(58)と、
を備えることを特徴とするオーディオデコーダ(50)。

30

【請求項8】

前記フレーム境界検出部(56)は、前記マッチングパタン(60)の所定数のインスタスを検索するように構成され、前記検出されたフレーム境界(74)は、前記所定数のインスタスのうちの最後のものに対応することを特徴とする請求項7に記載のオーディオデコーダ(50)。

40

【請求項9】

前記フレーム境界検出部(56)は、停止信号を受信するように構成され、
前記フレーム境界検出部(56)は更に、前記停止信号が受信されるまで、前記マッチングパタン(60)のインスタスがないか前記データストリーム(70)の部分を検索するように構成されている
ことを特徴とする請求項7に記載のオーディオデコーダ(50)。

【請求項10】

オーディオ処理誤りを検出し、前記オーディオ処理誤りに対応する前記データストリームにおける誤り位置を特定する復号化誤り検出部を更に有し、
前記フレーム境界検出部(56)は、前記誤り位置から前記検索を開始する

50

ことを特徴とする請求項7に記載のオーディオデコーダ(50)。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、携帯音楽プレーヤその他のマルチメディア装置などに使用し得る、オーディオデコーダに関する。オーディオデコーダは、格納されたオーディオファイルまたはネットワークを介して提供されるデータストリームを復号化するために使用される。

【背景技術】

【0002】

オーディオを符号化するために様々な標準規格が知られている。また、符号化オーディオデータをデータストリーム(データファイルまたはネットワークを介して提供されるデータストリームを含む)に含めるために様々な標準規格も知られている。Audio Data Transport Stream (ADTS)方式はそれらの一例で、広く使われているAdvanced Audio Coding (AAC) 標準規格に従って符号化されたオーディオをデータストリームに含ませて送信するために使用される。

10

【0003】

ADTSその他の方式では、データストリームを、それぞれのフレームがヘッダを有するオーディオデータのフレームに組織化する。あるアプリケーションでは、符号化オーディオフレームの始まりを判定するためにデータストリームの部分をスキャンする必要がある。そのスキャンを容易にするために、フレームヘッダに同期語(syncword)と呼ばれるものを含めるのが一般的である。同期語は、一定の長さを持ち、一定の値であり、一般的には、例えばヘッダの最初などヘッダの決められた位置に置かれる。

20

【0004】

データストリームをスキャンして同期語の存在を検出することはフレームヘッダを特定するのに有効ではあるが、誤りを生じることもあり得る。実用的な理由から、一般に同期語は12ビットなど、比較的短いため、オーディオペイロードデータ中、すなわちフレームヘッダの外で、見かけ上の同期語が現れることがあり得る。それによって誤ったフレーム検出が行われてしまう。そのような検出誤りから回復するため様々な方法はあるが、検出誤りによって貴重な処理時間、処理サイクルが使われてしまう。

【0005】

したがって、検出誤りを低減して、データストリームにおけるフレーム境界を効果的に特定する方法が必要である。

30

【発明の概要】

【0006】

データストリームにおける複数の符号化オーディオフレームであって各フレームがヘッダを有する符号化オーディオフレームを復号化するオーディオデコーダが提供される。オーディオデコーダは、同期語と、有効な符号化オーディオフレームのヘッダフィールドに対する少なくとも1つの予測値に対応する1つ以上の付加ビットとを含むマッチングパターンを生成し、マッチングパターンのインスタンスがないかデータストリームの部分を検索してフレーム境界を検出し、検出されたフレーム境界に対応するデータストリームにおける点から始まる1つ以上の符号化オーディオフレームを復号化するための1つ以上の回路を含む。

40

実施形態において、オーディオデコーダは、同期語と、有効な符号化オーディオフレームのヘッダフィールドに対する少なくとも1つの予測値に対応する1つ以上の付加ビットとを含むマッチングパターンを生成するマッチングパターン生成部を含む。オーディオデコーダは更に、前記マッチングパターンのインスタンスがないかデータストリームの部分を検索してフレーム境界を検出するフレーム境界検出部と、前記検出されたフレーム境界に対応する前記データストリームにおける点から始まる1つ以上の符号化オーディオフレームを復号化するフレームデコーダとを含む。実施形態において、前記フレーム境界検出部は、前記マッチングパターンの所定数のインスタンスを検索するように構成され、前記検出され

50

たフレーム境界は、前記所定数のインスタンスのうちの最後のものに対応する。また、前記フレーム境界検出部は、停止信号を受信するように構成され、前記フレーム境界検出部は更に、前記停止信号を受信されるまで、前記マッチングパタンのインスタンスがないか前記データストリームの部分を検索するように構成される。また、実施形態において、前記フレーム境界検出部は更に、検出された前記マッチングパタンのインスタンスの数を示す、前記停止信号の生成に用いるためのフレーム検出信号を提供するように構成される。また、実施形態において、前記符号化フレームは、Audio Data Transport Stream (ADTS) ヘッダを含み、マッチングパタン生成部は、12ビット同期語と、1ビットIDフィールド、2ビットレイヤフィールド、1ビット保護なしフィールドの予測値に対応する付加ビットを含むマッチングパタンを生成するように構成される。

10

また、実施形態において、オーディオ処理誤りを検出し、前記オーディオ処理誤りに対応する前記データストリームにおける誤り位置を特定する復号化誤り検出部を更に有し、前記フレーム境界検出部は、前記誤り位置から前記検索を開始する。また、実施形態において、前記フレーム境界検出部は更に、前記検出されたフレーム境界が有効なヘッダに対応しているかを検証するように構成される。また、実施形態において、前記フレーム境界検出部は、前記データストリームにおけるCRC (cyclical redundancy checksum) ビットを評価して前記検出されたフレーム境界が有効なヘッダに対応しているかを確認することによって、前記検出されたフレーム境界が有効なヘッダに対応しているかを検証するように構成される。

データストリームにおける複数の符号化オーディオフレームを復号化するための種々の方法も開示する。そのような方法の一例は、同期語と、有効な符号化オーディオフレームのヘッダフィールドに対する少なくとも1つの予測値に対応する1つ以上の付加ビットを含むマッチングパタンを生成するステップと、前記マッチングパタンのインスタンスがないかデータストリームの部分を検索することによりフレーム境界を検出するステップと、前記検出されたフレーム境界に対応する前記データストリームにおける点から始まる1つ以上の符号化オーディオフレームを復号化するステップとを有する。実施形態において、前記フレーム境界を検出するステップは、前記マッチングパタンの所定数のインスタンスを検索するステップを含み、前記検出されたフレーム境界は、前記所定数のインスタンスのうちの最後のものに対応する。実施形態において、上記方法は、停止信号を受信するステップを更に有し、前記フレーム境界を検出するステップは、前記停止信号を受信されるまで、前記マッチングパタンのインスタンスがないか前記データストリームの部分を検索するステップを含む。実施形態において、前記フレーム境界を検出するステップは、前記停止信号を受信される前に検出された前記マッチングパタンの最後のインスタンスに対応するフレーム境界を検出するステップを含む。実施形態において、上記方法は、検出された前記マッチングパタンのインスタンスの数を示す、前記停止信号の生成に用いるためのフレーム検出信号を提供するステップを更に有する。

20

30

方法の実施形態においては、上記した装置と同様に、前記符号化オーディオフレームは、Advanced Audio Codec Rawデータブロックを含む。実施形態において、前記フレームヘッダは、Audio Data Transport Stream (ADTS)ヘッダを含み、前記マッチングパタンは、12ビット同期語と、1ビットIDフィールド、2ビットレイヤフィールド、1ビット保護なしフィールドの予測値に対応する付加ビットを含む。

40

実施形態において、上記したいずれかの方法は、オーディオ処理誤りを検出するステップと、前記オーディオ処理誤りに対応する前記データストリームにおける誤り位置を特定するステップとを更に有し、前記マッチングパタンのインスタンスがないか前記データストリームの部分を検索するステップは、前記誤り位置から開始する。また、実施形態において、前記フレーム境界を検出するステップは、前記検出されたフレーム境界が有効なヘッダに対応しているかを検証するステップを含む。また、実施形態において、前記検出されたフレーム境界が有効なヘッダに対応しているかを検証するステップは、CRC (cyclical redundancy checksum) ビットを評価して前記検出されたフレーム境界が有効なヘッダに対応しているかを確認するステップを含む。

50

【図面の簡単な説明】

【0007】

【図1】符号化オーディオフレームを含むデータストリームを示す図である。

【図2】符号化オーディオフレームのヘッダ構造の一例を示す図。

【図3】本発明の実施形態で使用されるマッチングパタンの一例を示す図。

【図4】符号化オーディオフレームを処理する方法の一例を示す図。

【図5】オーディオフレームを処理するオーディオデコーダの一例を示すブロック図。

【発明を実施するための形態】

【0008】

本発明は、符号化オーディオデータを含むデータストリームであってデータストリームがフレームに組織化されるデータストリームを処理する方法を提供する。以下説明する方法で、フレーム境界の検出誤りを少なくし、オーディオデコーダにおける誤り回復の改良及びオーディオ処理機能の強化を可能にする。本発明は、ファイルに組織化され、不揮発性メモリに格納されるオーディオデータ、または、ネットワーク接続可能な装置で受信される、オーディオストリームまたはマルチメディアストリームにおけるオーディオデータに適用可能である。

10

【0009】

図1は、いくつかの符号化オーディオフレーム72を含むデータストリーム70を示す。符号化オーディオフレーム72はそれぞれヘッダ80を含む。ここで、ヘッダの始まりはフレーム境界74に対応する。

20

【0010】

データストリーム70は、MP3 (MPEG Layer 3)、Advanced Audio Coding (AAC) などの種々の公知のオーディオ符号化方式のうちの一つによって符号化されたオーディオデータを含むことができる。AACは、MPEG-2標準規格 (正式には ISO/IEC 13818-7:1997) の Part 7 として、また、MPEG-4 標準規格 (正式には ISO/IEC 14496-3:1999) の Part 3 として標準化されている。さまざまなオーディオ符号化方法が既に存在しており、将来も開発されていくであろうが、それらの各手法は、オーディオデータを圧縮し符号化するための様々な技術を含むことは、当業者は理解するであろう。AAC標準でいうと、それ自体で、“プロファイル (profile)” または “オブジェクトタイプ (object type)” に整理される多数の符号化方法を有している。

30

【0011】

AACで符号化されたような符号化オーディオデータは一般的に、データブロックの系列からなる。そのデータをカプセル化するために様々な方法が提案されている。その中で一番単純な方法は、オーディオデータがファイルとして整理され、完全なファイルとしてメモリに格納される状況において使用されることを想定している。そのような状況においては、オーディオのカプセル化は、単にデータファイルの最初に1つのヘッダを挿入するだけでよい。このヘッダは、オーディオデータのフォーマットを示すデータを、その他のデータとともに含むことができる。例えば、AACファイル作成するためには、Audio Data Interchange Format (ADIF) が、AACデータとともに使用される。ADIFヘッダは、ファイルのフォーマットを特定するためのフィールド、著作権の管理に関するデータ、オーディオデータの生成に使用されたオーディオ符号化方式に関する詳細のデータを含む。

40

【0012】

ネットワーク環境におけるオーディオストリーム又はマルチメディアストリームの転送などを扱うために、符号化オーディオデータのより複雑なカプセル化手法が開発されている。インターネットラジオやモバイル通信で実現されるネットワークストリーム環境においては、オーディオデコーダは、全てのオーディオデータにアクセスすることがいつでもできるわけではない。さらに、例えばビデオデータのように、データ転送のために、オーディオデータはマルチメディアと織り合わされることもあり得る。この場合に対応するべく、オーディオデータが図1に示される符号化オーディオフレーム72のようなフレームに組織化される、オーディオデータをカプセル化するさまざまな手法が提案されている。

50

AACデータとともに使用されるそのような方式の一例が、Audio Data Transport Stream (ADTS)フォーマットである。このフォーマットは、MPEG-2 Part 7、MPEG-4 Part 3において、AACとともに標準化されている。図1に示されるように、ADTSフォーマットデータは一般に、データストリーム70に整理され、そのデータストリームが符号化オーディオフレーム72に組織化され、それぞれの符号化オーディオフレーム72はヘッダ80を含む。

【0013】

ADTSを使うか使わないにかかわらず、データストリームは、符号化オーディオに加えて、例えばビデオデータなどの他のデータを含みうることは、当業者は認識するであろう。したがって、一連の符号化オーディオフレーム72としてフォーマットされたオーディオデータを使用する転送手法は、データストリーム70における他のデータと区別するのに有用である。よって、符号化オーディオフレーム72は、連続するブロックに組織化される必要はない。さらに、オーディオフレームを使用するADTS及びその他の転送方式は、データネットワークにおけるオーディオのストリーミングに関するアプリケーションに限定されない。ADTSのようなフレームベースの方式は、ADIFのような簡単なフォーマットよりもオーバーヘッドが増大するが、フレームベースのフォーマットは言うまでもなく、オーディオデータがファイルに組織化され、取得や再生のためメモリに格納される状況に好適である。従って、本明細書において、「データストリーム」とは、ファイルに組織化されてメモリに格納されるデータ、あるいは、オーディオデコーダがいつでもオーディオデータの全てにアクセスできるわけではないインターネットラジオなどのストリーミングアプリケーションにより転送されるデータをいう。

【0014】

図2は、データストリーム70における符号化オーディオフレーム72のヘッダ80の一例を示す図である。ヘッダ80は、フレームヘッダの存在を表すのに使用される固定長のビット列である同期語(syncword)82を含む。図2では、同期語82は12個の“1”からなるビット列からなり、フレームヘッダの最初に現れる。ADTSフォーマットでは図2に示されるようなヘッダが使用されるが、他のフォーマットにおいては、異なる長さ、異なるデータ、さらには、ヘッダ中の異なる位置に現れる同期語を有するヘッダが使用されうることはもちろんである。しかし、所与の転送フォーマットにおいては構成及び内容が一定であることが、同期語82の一貫性の特徴である。したがって、ADTSでフォーマットされたすべてのデータストリームにおいては、例えば、各ヘッダ80は同一の同期語82を有することになる。

【0015】

一方、ヘッダ内の他のフィールドは、データストリームによって変化しうる。例えば、図2のヘッダ80は1ビットのIDフィールドを含んでいる。ADTSでは、このフィールドは、データストリーム70がMPEG-2標準で符号化されたのか(IDフィールド=1)、MPEG-4標準で符号化されたのか(IDフィールド=0)を示している。そのため、このフィールドはデータストリームによって変わることがあり得る。また図2には、ADTSでは“00”固定されているレイヤフィールド86、保護なし(Protection Absent)フィールド88(ADTSでは、ヘッダがチェックサムを含むかどうかを表す1ビットのフィールド)、プロファイルフィールド90(ADTSでは、AAC符号化の中でどれが使われたかを表す2ビットのフィールド)も表示されている。ヘッダは最後に、ヘッダの安全性の検証に使用される、ADTSではオプションであるCRC(cyclical redundancy check)チェックサム・フィールド92を含む。

【0016】

図2はヘッダの一例の構成を示したものに過ぎない。様々な代案が可能だが、通常のヘッダ80は、同じタイプのデータストリームに対してすべての値が一定である同期語と、さらに、同じタイプの違ったデータストリーム70で変化する又は同じデータストリームの中でもヘッダによって変わる様々な他のフィールドとを備える。例えば、ADTSでは、IDフィールド84、レイヤフィールド86、保護なしフィールド88、プロファイルフィー

10

20

30

40

50

ルド90は、通常は1つのデータストリーム70の中では一定である。ただし、データストリームが変わると1つ以上のフィールドが変わることはあり得る。一方、CRCフィールド92は、同じデータストリーム中でもヘッダ80とヘッダ80との間で変わり得る。ヘッダの中で1つ以上のフィールドがデータストリームの中で一定であるため、多くの場合、任意のヘッダ80の同期語のみならず、有効なヘッダ80の内容について前もって得た知識から1つ以上のフィールドの値を予測することが可能である。

【0017】

データストリーム70を処理する場合には、フレームヘッダ80の始めに対応するフレーム境界74を検出する必要がある。データストリーム70は通常は直線的(すなわち、1ビットずつ、または、1ワードずつ)に処理されるが、データストリーム70の中に誤りのあるデータが存在すると、データストリーム70の処理される位置から次のヘッダ80の位置を特定する必要が生じる。さらに、オーディオ再生装置では、1つまたは複数の符号化オーディオフレーム72がスキップされる場合があるので、ヘッダの特定を繰り返し行うための、より複雑な機能が必要である。たとえば、早送り機能では、データストリーム70の任意の点でデータ処理を停止し、そのデータストリーム70における次の符号化オーディオフレーム72から再開する必要がある。そのような機能は、停止信号が送られるまで符号化オーディオフレーム72をスキップすることが必要になるであろう。代わりに、そのような機能は、所定の数の符号化フレームをスキップし、次に来る符号化オーディオフレーム72から再生(すなわち復号化)を再開することが必要になる。

【0018】

一般に、データストリーム70は連続的にスキャンされ同期語82に一致するビット列が検索される。そのため、次の符号化オーディオフレームに進み、単純に同期語82と一致するまで次々とデータストリーム70をスキャンし、一致したビットの位置から、符号化オーディオフレーム72の処理が始まる。

【0019】

しかし、同期語82が実用的な長さであるほど、同期語82とマッチするビット列がヘッダ80の同期語位置であると限らなくなってくる。一致するビット列が符号化オーディオデータの中でランダムな位置で現れることがあり得る。具体的な例としては、そのようなビット列のランダムな発生がASTSフォーマットされたデータの中でよく見られている。

【0020】

結果として、符号化オーディオの処理では、上述の従来技術に依存するどんなフレーム境界検出方法でも、受け入れがたい頻度の検出誤りが生じうる。そのような検出誤りを防止する方法においては、同期語に一致するビット列を検出してから、続くビットが通常のヘッダファイルに対応するかを確認するため解析することで、正しく解析されれば、次のオーディオデータの処理を進める。この解析は、ヘッダ80の安全性を検証するCRCチェックサム・フィールド92の評価を含み、これにより有効なヘッダ80が検出されたことを間接的に検証している。

【0021】

しかし、ヘッダを全て解析するには時間がかかる。処理サイクルが限定されている処理環境では、頻繁に起こるフレーム境界の検出誤りは、たとえその頻度が比較的低いとしても、望ましくない。

【0022】

図3は、本発明による実施形態で使用し得るマッチングパタン60を示している。マッチングパタン60は、ターゲットとするデータストリーム70の有効な符号化オーディオフレーム72にある同期語82と同一の同期語62を含む。そのマッチングパタン60には、データストリーム70の有効な符号化オーディオフレーム72のヘッダ80中で予測できる一つまたは複数のフィールドの値に対応する付加ビット64も含まれる。その付加ビット64の内容を以下で説明する。付加ビットは効果的に同期語を拡張するために使用できる。同期語の長さは検出誤り頻度に直接関係するので、同期語の拡張は、検出誤り頻度を下げることができる。

10

20

30

40

50

【 0 0 2 3 】

図4は、本発明による1つまたは複数の実施形態で、データストリーム70の符号化オーディオフレーム72を処理する方法を示している。データストリーム70に関して知られている情報に基づいて、最初に、マッチングパタン60を再生する(ブロック100)。特に、マッチングパタンは、ターゲットとするデータストリーム70の有効なヘッダ80で現れる同期語82に対応する同期語60を含む。例えば、ターゲットとするデータストリーム70がADTSフォーマット化されたデータであるとき、同期語62は12個の“1”ビットの列で構成される。

【 0 0 2 4 】

ブロック100で生成されたマッチングパタン60は、1つ以上の付加ビット64も含む。この付加ビット64は、特定のデータストリーム70の有効なヘッダ80に含まれている1つまたは複数のフィールドの予測値を含む。上述したように、同じ種類のデータストリーム70間で値が変化しても、特定のデータストリーム70中を考えると、ヘッダ80の特定のフィールドの値は一定である。それに応じて、ターゲットとするデータストリーム70の1つのヘッダ80に対してそのフィールドの値がわかれば、他のヘッダ80の中でも対応するフィールドの値が同じであることが予想できる。

【 0 0 2 5 】

また図2を参照すると、ADTSヘッダは、例えば、IDフィールド84、レイヤフィールド86、保護なしフィールド88、プロファイルフィールド88を含みうる。データストリームがADTSでフォーマットされている場合、そのフィールドの全てが、特定のデータストリーム70の中で、通常は一定である。逆に、CRCチェックサム・フィールド92の場合は、フォーマットされたヘッダとヘッダの間でフィールドが変化する。

【 0 0 2 6 】

したがって、オーディオデコーダは、ADTSフォーマットされたデータストリーム70で使用するためのマッチングパタン60を生成でき、そのマッチングパタンは、12ビットの同期語62と、付加ビット64としてIDフィールド84とレイヤフィールド86、保護なしフィールド88、プロファイルフィールド90のうちの1つまたは複数に対応する予想値を含む。以下の限定にはならない例では、結果として得られるマッチングパタン60の長さは18ビットである。代わりに、マッチングパタン60は12ビットの同期語62と、付加ビット64としてIDフィールド84とレイヤフィールド86と、保護なしフィールド88とだけに対応する予想値を含むこともあり得る。この場合には、マッチングパタン60の長さは16ビットで、つまり2バイト、である。この長さが本発明の実施形態ではより便利であり得る。

【 0 0 2 7 】

図4のブロック100は、マッチングパタン60を生成することを示している。マッチングパタン60は同期語62と付加ビット64の様々な組み合わせにより構成することができる。上述したように、その付加ビットはデータストリーム70の有効な符号化オーディオフレーム72のヘッダにある1つ以上のフィールドに対応する予想値から構成される。そのヘッダフィールドの値は、ターゲットとするデータストリーム70の過去の情報を用いて予想することができる。この過去の情報は、ターゲットとするデータストリーム70の1つのヘッダ80の内容を解析することで、または、別に提供されているターゲットとするデータストリーム70の情報を使うことで取得することができる。例えば、ストリーミングされる環境では、オーディオストリームを発生させているコンピュータサーバが、データストリーム70と別にオーディオストリームを記述するパラメータを提供することがあり得る。そのパラメータとしては、例えば、データストリーム70はMPEG-2標準に従ったAACエンコードされたデータである、あるいは、データストリーム70はヘッダ80にはCRCチェックサム・フィールド92を含まない、といった情報を提供するものがある。これらのパラメータがどのようにフォーマットされたかにかかわらず、以上により、最初にヘッダ80を復号化することなく、ヘッダ80の中での複数のヘッダフィールドの値を予想することができる。そのため、オーディオデコーダは、ヘッダ80をデコードする

10

20

30

40

50

ことで得られる情報、又は、別に提供される情報を使ってマッチングパターン60を生成し得る。

【0028】

図4は更に、マッチングパタンのインスタンス(instance)がないかをデータストリーム70の一部を検索することによりフレーム境界74を検出することを示している(ブロック102)。この検索は上述の同期語検索と同じように、つまりマッチングパターン60に一致するビット列を見つけるためデータストリームを連続的にスキャンすることで行うことができる。例えば、これは、データストリームを、マッチングパタンの長さと同じ長さのシフトレジスタに連続的に通すことで実行することができる。それぞれのサイクルでは、シフトレジスタの内容をマッチングパターン60と比較し、一致した場合にはフレーム境界を示すことになる。代わりに、データストリーム70の一部を、その一部の全ての可能な位置をマッチングパターンと比較するように設定された処理部がメモリに格納し、マッチを発見したとき、フレーム境界の検出を示す。上に挙げられた例はすべて、ただの例であって、それらの態様で本発明を限定するものではない。情報処理技術の分野における通常の知識を有する者であれば、データストリーム70の一部からマッチングパターン60のインスタンスを検索する様々な方法が可能であることが分かる。

10

【0029】

どんな場合でも、マッチングパターン60は同期語62より長いので、マッチングパターンとデータストリームの任意ビット列と一致する確率は、同期語だけの場合に比べて低い。マッチングパターン60に含まれる付加ビット64の数によるが、この方法により、検出誤り率を大幅に低減させることができる。例えば、符号化された音楽は一般にランダムであると仮定すると、長さ16ビットのマッチングパターンを使用することで検出誤り率を93%以上に低減させることができる。もちろん、実際の場合では、この改善の効果は変化するが、それでも、検出誤り率は、比較的少ない付加ビットを使った時にも、大幅に低減する。

20

【0030】

ブロック102で表示している検出ステップでは、オプションとしてデータストリーム70の中でマッチングパタンの複数のインスタンスを検出することもできる。この方法の一例として、データストリーム70の中で、マッチングパタンのインスタンスを所定の数だけ連続的に検査し、最終に検索したフレーム境界を検出する。例えば、この方法のアプリケーションはフレーム5個をスキップする必要がある場合が挙げられる。この場合には、検出ステップは、データストリーム70の中で、5個の連続したマッチングパターン60のインスタンスを検索することを含み、検出したフレーム境界はその5個の連続したインスタンスの最終のインスタンスに対応する。

30

【0031】

他の実施形態では、停止信号を受けるまで、データストリーム70がマッチングパターン60の複数のインスタンスが連続的に検査される。この実施形態では、検出されたフレーム境界74は停止信号を受ける前に最終的に検査されたマッチングパターン60のインスタンスに対応するものとすることができる。

【0032】

本発明のもう一つの実施形態としては、それぞれのマッチングパターン60のインスタンスの検出を、マッチングが得られたこと示す信号の発生トリガとし、この信号を停止信号として使用する。例えば、マッチングパターン60の複数のインスタンスのためにデータストリーム70が高速に検索される。各マッチングによって信号が発生するので、その発生した信号を数えることで、マッチが検出された回数を示すパラメータが得られる。たとえば、あるアプリケーションが60フレームをスキップすることが必要になると、60回のマッチがカウントされるまで検索を実行させ、60回のカウントを完了すると停止信号を発生することでスキップ処理を終了させる。この場合、検出したフレーム境界74は、この例では、停止信号を受ける前に検出されたマッチングパターン60の最終のインスタンスに対応する。

40

50

【 0 0 3 3 】

フレーム境界 7 4 を検出してから、次の符号化オーディオフレーム 7 2 に進むことができる。本発明のある実施形態では、ブロック 1 0 4 で表示しているように、検出されたフレーム境界 7 4 に対応する符号化オーディオフレーム 7 2 のヘッダ 8 0 がオーディオデータを復号化する前に検証することができる。例えば、CRCチェックサム・フィールド 9 2 を計算することでヘッダ 8 0 を正しく受信したことを確認する。フレーム検出誤りの場合（本発明の実施形態では、発生する確率は低いものの発生する可能性はある）には、CRCチェックサム 9 2 はほぼ確実に不良となり、データが破損したこと、又は、間違っ

10

たフレーム境界 7 4 を検出していることが示される。したがって、CRCチェックサム・フィールド 9 2 の評価によって、検出したフレーム境界 7 4 は有効なヘッダ 8 0 に対応しているかを確認できる。

【 0 0 3 4 】

検出したフレーム境界 7 4 が有効なヘッダに対応することを検証できる他の方法も使用可能である。例えば、ヘッダ 8 0 はフレーム長さを示す情報を含んでいれば、処理部はデータストリームの先にあり得る次のフレームの同期語が正しい位置であることが確認できる。しかし、検出されたフレーム境界 7 4 が有効なヘッダに対応することを検証する処理は一般に、追加的な処理ステップを必要とする。したがって、本発明の教示による検出誤りの低減は、フレーム境界検出の検証に使われる処理ステップの低減にもつながる。

【 0 0 3 5 】

ブロック 1 0 6 で示すように、検出されたフレームヘッダが有効であれば、符号化オーディオフレーム 7 2 の復号化が、データストリームの検出されたフレーム境界に対応する点から始まる。符号化オーディオフレームの復号化は適用される符号化方法に従って行われる。したがって、例えば、AACで符号化されたオーディオフレーム 7 3 はAACデコーダを使って復号化される。

20

【 0 0 3 6 】

図 5 は、本発明の一つまたは複数の実施形態によるオーディオデコーダの一例を簡単化して示すブロック図である。オーディオデコーダ 5 0 は、最低限、制御論理部 5 2 と、マッチングパタン生成部 5 4 と、フレーム境界検出部 5 6 と、フレームデコーダ 5 8 とを備える。デコーダ 5 0 はメモリ 4 0 とインタフェースするように表示され、復号化オーディオを出力する。

30

【 0 0 3 7 】

制御論理部 5 2 は、オーディオデコーダ 5 0 の全体的な制御を行う。制御論理部 5 2 がオーディオ復号化処理の開始及び停止をトリガする。さらに、制御論理部 5 2 は、キーボード、タッチ画面などのユーザインタフェースのための論理部も含み、ユーザがオーディオデコーダ 5 0 を操作することを可能にする。

【 0 0 3 8 】

代わりに、又は、追加として、制御論理部 5 2 は他のソフトウェアまたはソフトウェアモジュールと通信するためのAPI(application programming interface)を実現する。

【 0 0 3 9 】

マッチングパタン生成部 5 4 は、上述したように、ターゲットとするデータストリーム 7 0 に対して使用するマッチングパタン 6 0 を生成する。そのため、マッチングパタン生成部 5 4 には、データストリームで使われている同期語 8 2 の情報を含むターゲットとするデータストリーム 7 0 についての情報が提供される。追加として、マッチングパタン生成部 5 4 には、ターゲットとするデータストリーム 7 0 中で、最低限一つの有効なヘッダフィールドの予測値の情報が提供される。上述したように、その情報は、ターゲットとするデータストリームの中で1つのヘッダ 8 0 を読むことで、又は、別に提供されるデータストリームに関する情報から参照することで取得することができる。どの場合でも、マッチングパタン生成部 5 4 は、同期語 6 2（同期語 8 2 とまったく同じである）と有効なヘッダの1つまたは複数のヘッダフィールドに対応する予測値とを含むマッチングパタン 6 0 を生成する。

40

50

【0040】

マッチングパタン60のインスタンスをデータストリームの一部の中から検索するために、フレーム境界検出部56によってマッチングパタン60が使用される。マッチングパタン60のそれぞれのインスタンスは一般に、フレーム境界74に対応する。本発明のある実施形態では、フレーム境界検出部56は、マッチングパタンの最初のインスタンスで検索を停止し、それに対応するフレーム境界を取得する。他の実施形態では、フレーム境界検出部56は、データストリーム70を、制御論理部52から停止信号を発生するまで検索し続けるように構成され、それにより、マッチングパタン60の複数のインスタンスを検索する。この例の場合、検出されたフレーム境界74は停止信号を受ける前に検索された最終のマッチングパタン60のインスタンスに対応するフレーム境界になる。

10

【0041】

もう一つの方法として、上述したように、フレーム境界検出部56は、データストリーム70を、所定の回数のマッチングパタンのインスタンスを検索するまで検索し続けるように構成され、それによって、得られる検出されたフレーム境界74は最終に検索されたマッチングパタンのインスタンスに対応するものになる。

【0042】

どんな場合でも、フレーム境界検出部56は、検出されたフレーム境界74に関する情報はフレームデコーダ58に転送する。フレームデコーダ58は、適切な復号化アルゴリズムを使用し、1つまたは複数の符号化オーディオフレームを復号化する。フレームデコーダ58は、オーディオアプリケーションで使用するため、及び/又は、アナログオーディオに変換するために、例えばPCM(pulse code modulation)オーディオストリームのような非圧縮のオーディオストリームになる復号化オーディオ出力を発生する。

20

【0043】

オーディオデコーダ50はデータストリーム70にアクセスするため、メモリとインタフェースすることがあり得る。データストリーム70はファイルとして組織化し、メモリ40に格納することが可能で、この場合、メモリ40はランダムアクセスメモリ(RAM)、又はフラッシュ、磁気ディスク記憶装置のような不揮発性メモリであり得る。

【0044】

データストリーム70はネットワーク上でストリーミングオーディオ、またはマルチメディア情報源から生成されうる。メモリ40は、データストリーム70の一部をバッファリングするランダムアクセスメモリ(RAM)である。

30

【0045】

制御論理部52、マッチングパタン生成部54、フレーム境界検出部56、フレームデコーダ58は、デジタル論理ハードウェア、又は、ソフトウェアを実行するマイクロプロセッサ、または、その両方の組み合わせで実現することができる。どんなブロックでも専用のプロセッサで実現することができるし、いくつかのブロックを1つのプロセッサで実現することもできる。フレームデコーダ58は特に、専用のDSP(digital signal processor)で実現することができ、他の部分は一般的に全体として、または部分として、一般使用のためのマイクロプロセッサまたはDSPを使って実現することができる。追加として、どんなブロックの機能でも、発明の本質から離れることなく、1つまたは複数のプロセッサまたはハードウェアブロックの間で分けることがあり得る。

40

【0046】

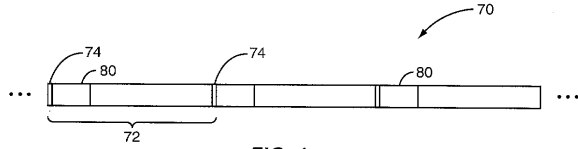
当業者であれば、本発明によれば、オーディオデコーダで使用される符号化オーディオストリームのフレーム境界を、高速かつ効果的に検出する方法及び装置が広く提供されることが理解されよう。

【0047】

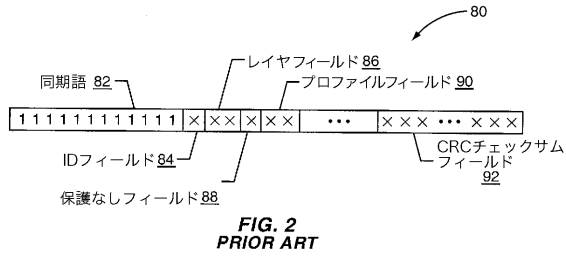
本発明は、発明の本質から離れることがなく、ここに記載されていない方法で実施することが可能である。そのため、本発明は上述の特徴及び利点又は図面に限定されない。本発明は、以下の特許請求の範囲及びそれに対して法律的に均等な範囲によってのみ限定される。

50

【図1】



【図2】



【図3】

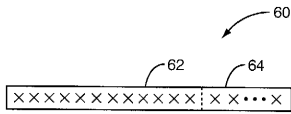


FIG. 3

【図4】

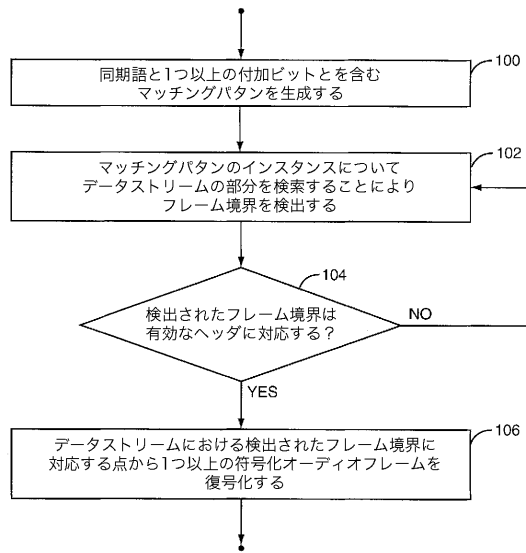


FIG. 4

【図5】

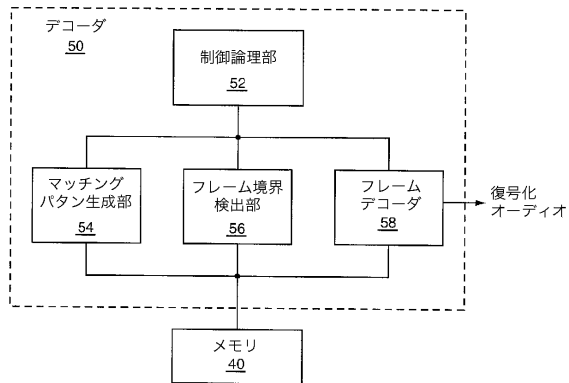


FIG. 5

フロントページの続き

(72)発明者 メッツ, ルディ ハンター
アメリカ合衆国 ノースカロライナ州 27713, ダラム, シルバー パイン コート 2
08

審査官 井上 健一

(56)参考文献 特開2008-84382(JP,A)
特表2002-530931(JP,A)

(58)調査した分野(Int.Cl., DB名)
G10L 19/00-19/14