

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2012-231303

(P2012-231303A)

(43) 公開日 平成24年11月22日(2012.11.22)

(51) Int.Cl.

HO4N 7/01 (2006.01)

F 1

HO4N 7/01

テーマコード(参考)

Z 5C063

審査請求 未請求 請求項の数 6 O L (全 20 頁)

(21) 出願番号

特願2011-98391 (P2011-98391)

(22) 出願日

平成23年4月26日 (2011.4.26)

(71) 出願人 000002185

ソニー株式会社

東京都港区港南1丁目7番1号

(74) 代理人 100082131

弁理士 稲本 義雄

(74) 代理人 100121131

弁理士 西川 孝

(72) 発明者 元山 琢人

東京都港区港南1丁目7番1号 ソニー株式会社内

F ターム(参考) 5C063 AC01 BA10 BA12 CA05 CA07

(54) 【発明の名称】 画像処理装置および方法、並びにプログラム

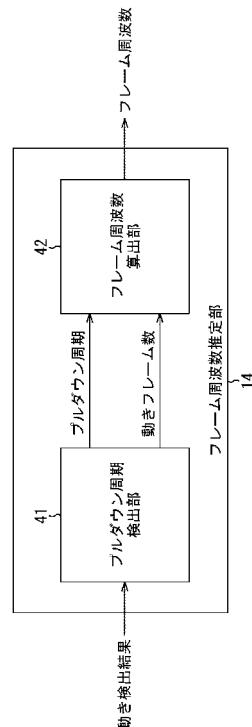
(57) 【要約】

【課題】任意のプルダウンパターンの入力画像について、プルダウンされる前のフレーム周波数を推定する。

【解決手段】プルダウン周期検出部は、プルダウンされた入力画像におけるプルダウンパターンを検出し、フレーム周波数算出部は、プルダウンパターン、および入力画像のフレーム周波数である第1のフレーム周波数に基づいて、入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出する。本技術は、所定のプルダウンパターンでプルダウン処理が行われた入力画像をフレームレート変換する画像処理装置に適用することができる。

【選択図】図7

図7



【特許請求の範囲】**【請求項 1】**

プルダウンされた入力画像におけるプルダウンパターンを検出するプルダウンパターン検出部と、

前記プルダウンパターン、および前記入力画像のフレーム周波数である第1のフレーム周波数に基づいて、前記入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出するフレーム周波数算出部と
を備える画像処理装置。

【請求項 2】

前記プルダウンパターン検出部は、前記入力画像のフレーム間の動きの有無のパターンに基づいて、前記プルダウンパターンが繰り返されるフレームの周期であるプルダウン周期を検出するとともに、前記プルダウン周期におけるフレーム間の動きの数を表す動きフレーム数をカウントし、

前記フレーム周波数算出部は、前記プルダウン周期、前記動きフレーム数、および前記第1のフレーム周波数に基づいて、前記第2のフレーム周波数を算出する

請求項1に記載の画像処理装置。

【請求項 3】

前記原画像のフレーム間の動きベクトルを検出する動きベクトル検出部と、

前記動きベクトル、前記第2のフレーム周波数、および、出力画像のフレーム周波数である第3のフレーム周波数に基づいて動き補償を行い、前記原画像の補間フレームを生成する動き補償部とをさらに備える

請求項1に記載の画像処理装置。

【請求項 4】

前記第2のフレーム周波数および前記第3のフレーム周波数に基づいて、前記原画像のフレーム間における前記補間フレームの時間的な位置を表す補間位相を求める補間位相決定部をさらに備え、

前記動き補償部は、前記原画像のフレーム間における前記補間位相で、前記補間フレームを生成する

請求項3に記載の画像処理装置。

【請求項 5】

プルダウンされた入力画像におけるプルダウンパターンを検出するプルダウンパターン検出ステップと、

前記プルダウンパターン、および前記入力画像のフレーム周波数である第1のフレーム周波数に基づいて、前記入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出するフレーム周波数算出ステップと

を含む画像処理方法。

【請求項 6】

プルダウンされた入力画像におけるプルダウンパターンを検出するプルダウンパターン検出ステップと、

前記プルダウンパターン、および前記入力画像のフレーム周波数である第1のフレーム周波数に基づいて、前記入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出するフレーム周波数算出ステップと

を含む処理をコンピュータに実行させるプログラム。

【発明の詳細な説明】**【技術分野】****【0001】**

本技術は、画像処理装置および方法、並びにプログラムに関し、特に、任意のプルダウンパターンの入力画像について、プルダウンされる前のフレーム周波数を推定することができるようとする画像処理装置および方法、並びにプログラムに関する。

【背景技術】

10

20

30

40

50

【0002】

近年、様々なフレーム周波数の画像信号が数多く存在している。

【0003】

例えば、映画のオリジナル画像のフレーム周波数は24Hzであるのに対し、コンピュータグラフィックスのオリジナル画像のフレーム周波数は30Hzである。また、国によっても、放送される画像のフレーム周波数が異なり、例えば、日本で放送される画像のフレーム周波数は60Hzであるのに対し、欧州で放送される画像のフレーム周波数は50Hzである。さらに、近年急増しつつあるインターネット上の動画コンテンツのフレーム周波数は、まさに多種多様である。

【0004】

10

このようにフレーム周波数が異なる画像をデジタルテレビジョン放送で放送する場合には、放送局側でフレーム周波数が統一されて、放送されている。例えば、フレーム周波数が24Hzである映画のオリジナル画像を60Hzのフレーム周波数で放送する場合、放送局側で3-2 プルダウン処理が行われ、フレーム周波数が30Hzである画像を60Hzのフレーム周波数で放送する場合、放送局側で2-2 プルダウン処理が行われる。

【0005】

20

ここで、3-2 プルダウン処理とは、図1に示されるように、例えば、フレーム周波数が24Hzである映画の1フレーム目の画像を、フレーム周波数が60Hzであるテレビジョン画像の1, 2, 3フィールド目に使用し、映画の2フレーム目の画像をテレビジョン画像の4, 5フィールド目に使用し、映画の3フレーム目の画像をテレビジョン画像の6, 7, 8フィールド目に使用し、映画の4フレーム目の画像をテレビジョン画像の9, 10フィールド目に使用し、この処理を順次繰り返すことにより、24Hzのフレーム周波数（フレームレート）を60Hzに変換する処理である。なお、プルダウン処理には、3-2 プルダウンや2-2 プルダウン以外のプルダウンパターンが存在する。

【0006】

しかしながら、このようにしてプルダウン処理が行われた画像がテレビジョン受像機等の画面に表示されると、例えば動きのあるシーンで、その動きが不自然に見える、いわゆるジャダーが、視聴者に知覚されてしまう。

【0007】

30

図2は、テレビジョン受像機等への入力画像としてのフレーム周波数が60Hzである3-2 プルダウン処理が行われた画像と、フレーム周波数が60Hzであるオリジナル画像のフレームの位相の比較を示している。図2において、上側には、3-2 プルダウン処理が行われた画像についての、時間（時刻）に対するフレームの位相が示されており、下側には、オリジナル画像についての、時間に対するフレームの位相が示されている。図2に示されるように、オリジナル画像については、時間に対して各フレームが順次出力されるのに対して、3-2 プルダウン処理が行われた画像については、時間に対して同じ位相のフレームが3枚、2枚の順番で出力されるようになる。すなわち、3-2 プルダウン処理が行われた画像は、画面に表示されると滑らかでない画像となってしまう。

【0008】

40

そこで、プルダウン処理が行われた画像をフレームレート変換する際に、3-2 プルダウンや2-2 プルダウン等の特定のプルダウンパターンを検出し、その検出結果に応じて画像を補正し、フレーム補間を行うようによることで、上述したジャダーを低減する手法が提案されている（例えば、特許文献1参照）。

【先行技術文献】**【特許文献】****【0009】****【特許文献1】特開2010-11108号公報****【発明の概要】****【発明が解決しようとする課題】****【0010】**

50

ところで、上述した手法では、3 - 2 プルダウンや2 - 2 プルダウン以外のプルダウンパターンを検出することができないので、プルダウンされる前のフレーム周波数を推定することができなかった。

【0011】

また、現存するプルダウンパターンを全てテーブルに保持することで、3 - 2 プルダウンや2 - 2 プルダウン以外のプルダウンパターンを検出することは可能になるが、全てのプルダウンパターンについて処理を行う必要があるため、制御が複雑になるとともに、コストがかかってしまう。さらに、現存しないプルダウンパターンには対応できないため、将来的な拡張性は低い。

【0012】

本技術は、このような状況に鑑みてなされたものであり、任意のプルダウンパターンの入力画像について、プルダウンされる前のフレーム周波数を推定することができるようとするものである。

【課題を解決するための手段】

【0013】

本技術の一側面の画像処理装置は、プルダウンされた入力画像におけるプルダウンパターンを検出するプルダウンパターン検出部と、前記プルダウンパターン、および前記入力画像のフレーム周波数である第1のフレーム周波数に基づいて、前記入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出するフレーム周波数算出部とを備える。

10

20

【0014】

前記プルダウンパターン検出部には、前記入力画像のフレーム間の動きの有無のパターンに基づいて、前記プルダウンパターンが繰り返されるフレームの周期であるプルダウン周期を検出させるとともに、前記プルダウン周期におけるフレーム間の動きの数を表す動きフレーム数をカウントさせ、前記フレーム周波数算出部には、前記プルダウン周期、前記動きフレーム数、および前記第1のフレーム周波数に基づいて、前記第2のフレーム周波数を算出させることができる。

【0015】

前記画像処理装置には、前記原画像のフレーム間の動きベクトルを検出する動きベクトル検出部と、前記動きベクトル、前記第2のフレーム周波数、および、出力画像のフレーム周波数である第3のフレーム周波数に基づいて動き補償を行い、前記原画像の補間フレームを生成する動き補償部とをさらに設けることができる。

30

【0016】

前記画像処理装置には、前記第2のフレーム周波数および前記第3のフレーム周波数に基づいて、前記原画像のフレーム間における前記補間フレームの時間的な位置を表す補間位相を求める補間位相決定部をさらに設け、前記動き補償部には、前記原画像のフレーム間における前記補間位相で、前記補間フレームを生成させることができる。

【0017】

本技術の一側面の画像処理方法は、プルダウンされた入力画像におけるプルダウンパターンを検出するプルダウンパターン検出ステップと、前記プルダウンパターン、および前記入力画像のフレーム周波数である第1のフレーム周波数に基づいて、前記入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出するフレーム周波数算出ステップとを含む。

40

【0018】

本技術の一側面のプログラムは、プルダウンされた入力画像におけるプルダウンパターンを検出するプルダウンパターン検出ステップと、前記プルダウンパターン、および前記入力画像のフレーム周波数である第1のフレーム周波数に基づいて、前記入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出するフレーム周波数算出ステップとを含む処理をコンピュータに実行させる。

【0019】

50

本技術の一側面においては、プルダウンされた入力画像におけるプルダウンパターンが検出され、プルダウンパターン、および入力画像のフレーム周波数である第1のフレーム周波数に基づいて、入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数が算出される。

【発明の効果】

【0020】

本技術の一側面によれば、任意のプルダウンパターンの入力画像について、プルダウンされる前のフレーム周波数を推定することが可能となる。

【図面の簡単な説明】

【0021】

【図1】3-2 プルダウン処理について説明する図である。

【図2】3-2 プルダウン処理が行われた画像とオリジナル画像のフレームの位相を比較する図である。

【図3】本技術を適用した画像処理装置の一実施の形態の機能構成例を示すブロック図である。

【図4】動き検出部の機能構成例を示すブロック図である。

【図5】画面の分割の例を示す図である。

【図6】動き検出部の他の機能構成例を示すブロック図である。

【図7】フレーム周波数推定部の機能構成例を示すブロック図である。

【図8】フレーム周波数推定処理について説明するフローチャートである。

【図9】プルダウン周期検出処理について説明するフローチャートである。

【図10】動き履歴の例を示す図である。

【図11】フレームレート変換処理について説明するフローチャートである。

【図12】補間位相について説明する図である。

【図13】動き補償処理の例について説明する図である。

【図14】コンピュータのハードウェアの構成例を示すブロック図である。

【発明を実施するための形態】

【0022】

以下、本技術の実施の形態について図を参照して説明する。なお、説明は以下の順序で行う。

1. 画像処理装置の機能構成
2. フレーム周波数推定処理
3. フレームレート変換処理

【0023】

<1. 画像処理装置の機能構成>

図3は、本技術を適用した画像処理装置の一実施の形態の構成を示している。

【0024】

図3の画像処理装置1は、プログレッシブ信号としての入力画像に対して、動きベクトルを用いたフレームレート変換処理を施し、フレームレート変換後の出力画像を、液晶ディスプレイ等の図示せぬ表示装置等に供給する。出力画像のフレームレートは、予め設定されていたり、ユーザによって設定されるようになされている。

【0025】

図3の画像処理装置1は、動き検出部11、フレームメモリ12、動きベクトル検出部13、フレーム周波数推定部14、フレーム制御部15、および動き補償部16から構成される。

【0026】

動き検出部11は、画像処理装置1に入力される入力画像のフレーム（以下、現フレームともいう）と、1フレーム前に入力された入力画像のフレーム（以下、前フレームともいう）との間の動きの有無を、1フレーム毎に検出する。動き検出部11は、検出した動きの有無を表す動き検出結果を、フレームメモリ12およびフレーム周波数推定部14に

10

20

30

40

50

供給する。

【0027】

[動き検出部の機能構成例]

ここで、図4を参照して、動き検出部11の詳細な機能構成例について説明する。

【0028】

図4の動き検出部11は、演算部21、絶対値算出部22、計測領域内総和計算部23、および閾値比較部24から構成される。

【0029】

演算部21は、現フレームの輝度値と、前フレームの輝度値との差分を画素毎に求め、絶対値算出部22に供給する。絶対値算出部22は、演算部21からのフレーム間の画素毎の輝度値の差分の絶対値(フレーム差分絶対値)を求め、計測領域内総和計算部23に供給する。

10

【0030】

計測領域内総和計算部23は、ユーザによって指定された計測領域内で、絶対値算出部22からの画素毎のフレーム差分絶対値の総和(フレーム差分絶対値和)を求め、閾値比較部24に供給する。閾値比較部24は、計測領域内総和計算部23からのフレーム差分絶対値和と、所定の閾値(動き閾値)とを比較する。動き閾値は、予め設定されていてよいし、ユーザにより設定されるようにしてもよい。

20

【0031】

閾値比較部24は、フレーム差分絶対値和が動き閾値より大きい場合、閾値比較部24は、現フレームと前フレームとの間に動きがあると判断し、その旨の動き検出結果を出力する。また、閾値比較部24は、フレーム差分絶対値和が動き閾値より小さい場合、現フレームと前フレームとの間に動きがないと判断し、その旨の動き検出結果を出力する。

【0032】

このようにして、動き検出部11は、フレーム間の動きの有無を表す動き検出結果を出力する。

【0033】

ところで、図4の動き検出部11においては、入力画像の一部のみに動きがあったり、2画面表示されるような入力画像の一方のみに動きがある等の場合には、入力画像全体における動きが相対的に小さくなってしまい、動きがあるにもかかわらず、動きがないと判断されてしまう恐れがある。

30

【0034】

そこで、図5に示されるように、入力画像を領域a乃至iの9領域に9分割し、それぞれの領域について、動きの有無が検出されるようにしてもよい。

【0035】

[動き検出部の他の機能構成例]

図6は、図5に示される、9分割された入力画像のそれぞれの領域について、動きの有無を検出するようにした動き検出部11の機能構成例を示している。

40

【0036】

図6の動き検出部11は、領域毎動き検出部31a乃至31i、およびOR演算部32から構成される。

【0037】

領域毎動き検出部31a乃至31iはそれぞれ、図4で説明した動き検出部11と同様に構成され、図5に示される入力画像の領域a乃至iについて動きの有無を検出する。すなわち、領域毎動き検出部31a乃至31iは、入力画像の領域a乃至iそれぞれについて、フレーム差分絶対値和を求めて、動き閾値と比較し、得られる動き検出結果をOR演算部32に供給する。

【0038】

OR演算部32は、領域毎動き検出部31a乃至31iそれからの動き検出結果に対してOR演算を行い、その結果を、入力画像(フレーム)全体の動き検出結果として出力す

50

る。すなわち、OR演算部32は、領域毎動き検出部31a乃至31iそれぞれからの動き検出結果において、動きがある旨の動き検出結果が1つでもある場合、入力画像全体について動きがある旨の動き検出結果を出力する。

【0039】

このような構成により、入力画像の一部のみに動きがあつたり、2画面表示されるような入力画像の一方のみに動きがあるような場合であつても、動きの有無が正しく判断されるようになる。

【0040】

なお、入力画像の分割方法は、図5に示されるものに限らず、また、動き検出部11の構成は、入力画像の分割方法に応じて、適宜変更されるようにもできる。

10

【0041】

さて、図3の説明に戻り、フレームメモリ12は、図示せぬフレームメモリコントローラを備えており、そのフレームメモリコントローラの制御の下、入力画像の各フレームや、動き検出部11からの動き検出結果、動きベクトル検出部13からの動きベクトルを記憶する。フレームメモリ12に記憶されている入力画像のフレームは、適宜、動き検出部11、動きベクトル検出部13、および動き補償部16に読み出され、フレームメモリ12に記憶されている動きベクトルは、適宜、動きベクトル検出部13および動き補償部16に読み出される。

【0042】

動きベクトル検出部13は、画像処理装置1に入力される現フレームと、フレームメモリ12に記憶されている前フレームとを用いて、入力画像の動きベクトルを1フレーム毎に検出し、フレームメモリ12に供給する。

20

【0043】

フレーム周波数推定部14は、動き検出部11からの動き検出結果を複数フレーム分保持し、入力画像がプルダウンされた画像である場合、その動き検出結果に基づいて、入力画像がプルダウンされる前のオリジナル画像のフレーム周波数を推定する。フレーム周波数推定部14は、推定したオリジナル画像のフレーム周波数をフレーム制御部15に供給する。

【0044】

[フレーム周波数推定部の機能構成例]

30

ここで、図7を参照して、フレーム周波数推定部14の詳細な機能構成例について説明する。

【0045】

図7のフレーム周波数推定部14は、プルダウン周期検出部41およびフレーム周波数算出部42から構成される。

【0046】

プルダウン周期検出部41は、入力画像が所定のプルダウンパターンでプルダウンされた画像である場合、動き検出部11からの動き検出結果に基づいて、そのプルダウンパターンを検出する。具体的には、プルダウン周期検出部41は、動き検出部11からの動き検出結果に基づいて、そのプルダウンパターンが繰り返されるフレームの周期であるプルダウン周期を検出する。また、プルダウン周期検出部41は、検出したプルダウン周期において、入力画像のフレーム間の動きの数を表す動きフレーム数をカウントする。さらに、プルダウン周期検出部41は、動き検出部11からの動き検出結果に基づいて、入力画像のフレーム周波数を特定する。プルダウン周期検出部41は、プルダウン周期および動きフレーム数を、入力画像のフレーム周波数とともにフレーム周波数算出部42に供給する。

40

【0047】

フレーム周波数算出部42は、プルダウン周期検出部41によって検出された入力画像のプルダウンパターンと、入力画像のフレーム周波数とに基づいて、入力画像がプルダウンされる前のオリジナル画像のフレーム周波数を算出する。具体的には、フレーム周波数

50

算出部 4 2 は、プルダウン周期検出部 4 1 からのプルダウン周期、動きフレーム数、および入力画像のフレーム周波数に基づいて、オリジナル画像のフレーム周波数を算出する。

【 0 0 4 8 】

図 3 の説明に戻り、フレーム制御部 1 5 は、フレーム周波数推定部 1 4 からのオリジナル画像のフレーム周波数と、予め設定されたか、またはユーザによって設定されている出力画像のフレーム周波数に基づいて、動き補償部 1 6 による動き補償処理に用いられるオリジナル画像の 2 フレーム（以下、ペアフレームともいう）を指定するフレーム制御信号を生成し、フレームメモリ 1 2 に供給する。また、フレーム制御部 1 5 は、動き補償部 1 6 による動き補償処理において生成される補間フレームの、オリジナル画像のペアフレーム間の時間的な位置を表す補間位相を求め、動き補償部 1 6 に供給する。このように、フレーム制御部 1 5 は、動き補償部 1 6 による動き補償処理を制御する。

10

【 0 0 4 9 】

動き補償部 1 6 は、フレームメモリ 1 2 に記憶されている入力画像のフレームのうちの、フレーム制御部 1 5 により生成されたフレーム制御信号で指定されるオリジナル画像のペアフレームと、ペアフレームに対応する入力画像についての動きベクトルに基づいて動き補償を行い、フレーム制御部 1 5 からの補間位相で、オリジナル画像の補間フレーム画像を生成する。

20

【 0 0 5 0 】

動き補償部 1 6 による動き補償処理の結果得られた画像は、フレームレート変換された出力画像として、液晶ディスプレイ等の図示せぬ表示装置に供給され、表示されるようになる。

20

【 0 0 5 1 】

< 2 . フレーム周波数推定処理 >

次に、図 8 を参照して、画像処理装置 1 によるフレーム周波数推定処理について説明する。図 8 のフレーム周波数推定処理は、フレーム周波数推定部 1 4 において、動き検出部 1 1 からの動き検出結果が所定数フレーム分保持されたときに開始される。

30

【 0 0 5 2 】

ステップ S 5 1 において、プルダウン周期検出部 4 1 は、プルダウン周期検出処理を実行し、入力画像のプルダウン周期を検出するとともに、入力画像の動きフレーム数をカウントする。

30

【 0 0 5 3 】

[プルダウン周期検出処理の例]

ここで、図 9 のフローチャートを参照して、プルダウン周期検出部 4 1 によるプルダウン周期検出処理について説明する。

40

【 0 0 5 4 】

ステップ S 8 1 において、プルダウン周期検出部 4 1 は、内部に保持している入力画像の所定数フレーム分の動き履歴としての動き検出結果 $his[0], \dots, his[MAXSIZE^2-1]$ を読み込む。

40

【 0 0 5 5 】

ここで、 $his[0]$ は、現フレームと前フレームについての動き検出結果を表し、 $his[k]$ は、現フレームより k フレーム前のフレームと、前フレームより k フレーム前のフレームについての動き検出結果を表す。また、 $MAXSIZE$ は、予めユーザによって設定される最大プルダウン周期を表す。なお、 $his[k]$ は、1 または 0 の値をとり、 $his[k] = 1$ は、動き検出結果として動きがあることを表し、 $his[k] = 0$ は、動き検出結果として動きがないことを表す。

【 0 0 5 6 】

図 1 0 は、動き履歴の例を示す図である。図 1 0 においては、入力画像として、一番右の「E」が記されたフレームが現フレームとなり、左にいくほど過去のフレームとなる。なお、図 1 0 の例において示される入力画像は、3 - 2 プルダウン処理された画像である

50

。

【 0 0 5 7 】

図 1 0においては、MAXSIZE = 6とされているので、ステップ S 8 1においては、動き履歴 $his[0], \dots, his[11]$ が読み出され、その値は、 $his[0]$ から順に、0, 0, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0 となっている。

【 0 0 5 8 】

ステップ S 8 2において、プルダウン周期検出部 4 1 は、最終的に検出すべきプルダウン周期 $cycleLength$ の値を 1 とする。

【 0 0 5 9 】

ステップ S 8 3において、プルダウン周期検出部 4 1 は、読み出した動き履歴 $his[0], \dots, his[MAXSIZE-1]$ の値が全て 0 であるか否かを判定する。図 1 0 の例においては、動き履歴 $his[0], \dots, his[MAXSIZE-1]$ の値が全て 0 ではないので、処理はステップ S 8 4 に進む。

【 0 0 6 0 】

ステップ S 8 4において、プルダウン周期検出部 4 1 は、読み出した動き履歴 $his[0], \dots, his[MAXSIZE-1]$ の値が全て 1 であるか否かを判定する。図 1 0 の例においては、動き履歴 $his[0], \dots, his[MAXSIZE-1]$ の値が全て 1 ではないので、処理はステップ S 8 5 に進む。

【 0 0 6 1 】

ステップ S 8 5において、プルダウン周期検出部 4 1 は、プルダウン周期の候補となるパラメータ（以下、候補プルダウン周期という） $tSize$ について、 $tSize = MAXSIZE$ とする。すなわち、図 1 0 の例においては、まず、 $tSize = 6$ とされる。

【 0 0 6 2 】

ステップ S 8 6において、プルダウン周期検出部 4 1 は、 $tSize > 1$ であるか否かを判定する。ステップ S 8 6において、 $tSize > 1$ であると判定されると、処理はステップ S 8 7 に進み、パラメータ i について、 $i = 0$ とする。

【 0 0 6 3 】

ステップ S 8 7において、プルダウン周期検出部 4 1 は、 $i < 2(MAXSIZE-tSize) + 1$ であるか否かを判定する。ここで、図 1 0 の例においては、 $2(MAXSIZE-tSize) + 1$ の値は 1 となり、 $i < 2(MAXSIZE-tSize) + 1$ であると判定されるので、処理はステップ S 8 9 に進む。

【 0 0 6 4 】

ステップ S 8 9において、プルダウン周期検出部 4 1 は、候補プルダウン周期 $tSize$ 分の動き履歴をテンプレートとして設定する。具体的には、1つめのテンプレート $template1$ として動き履歴 $\{ his[i], his[i+1], \dots, his[i+tSize-1] \}$ を設定し、2つめのテンプレート $template2$ として動き履歴 $\{ his[i+tSize], his[i+tSize+1], \dots, his[i+2*tSize-1] \}$ を設定する。すなわち、図 1 0においては、まず、 $template1$ として動き履歴 $\{ his[0], his[1], \dots, his[5] \}$ が設定され、 $template2$ として動き履歴 $\{ his[6], his[7], \dots, his[11] \}$ が設定される。

【 0 0 6 5 】

そして、ステップ S 9 0において、プルダウン周期検出部 4 1 は、 $template1$ と $template2$ は一致するか否かを判定する。図 1 0 の例では、 $template1 = \{ 0, 0, 1, 0, 1, 0 \}$ 、 $template2 = \{ 0, 1, 0, 1, 0, 0 \}$ でそれぞれ一致しないので、処理はステップ S 9 1 に進む。

【 0 0 6 6 】

ステップ S 9 1において、プルダウン周期検出部 4 1 は、候補プルダウン周期 $tSize$ について、 $tSize = tSize - 1$ とし、処理はステップ S 8 6 に戻る。すなわち、図 1 0 の例では、 $tSize = 5$ とされ、2回目のステップ S 8 6 以降の処理が実行される。

【 0 0 6 7 】

図 1 0 の例では、2回目のステップ S 8 9 において、 $template1$ として動き履歴 $\{ his[0,$

10

20

30

40

50

], his[1], …, his[4]} が設定され、template2として動き履歴 {his[5], his[7], …, his[9]} が設定される。そして、2回目のステップ S 9 0においては、template1 = {0, 0, 1, 0, 1}, template2 = {0, 0, 1, 0, 1} でそれぞれ一致し、処理はステップ S 9 2に進むようになる。

【0 0 6 8】

ステップ S 9 2において、プルダウン周期検出部 4 1は、パラメータ iについて、 $i = i + 1$ とし、処理はステップ S 8 8に戻る。すなわち、図 1 0 の例では、 $i = 1$ とされ、3回目のステップ S 8 8以降の処理が実行される。

【0 0 6 9】

図 1 0 の例では、3回目のステップ S 8 9において、template1として動き履歴 {his[1], his[2], …, his[5]} が設定され、template2として動き履歴 {his[6], his[7], …, his[10]} が設定される。そして、3回目のステップ S 9 0においては、template1 = {0, 1, 0, 1, 0}, template2 = {0, 1, 0, 1, 0} でそれぞれ一致し、処理は2回目のステップ S 9 2に進む。10

【0 0 7 0】

図 1 0 の例では、2回目のステップ S 9 2においては、 $i = 2$ とされ、4回目のステップ S 8 8以降の処理が実行される。

【0 0 7 1】

すなわち、4回目のステップ S 8 9において、template1として動き履歴 {his[2], his[3], …, his[6]} が設定され、template2として動き履歴 {his[7], his[8], …, his[11]} が設定される。そして、4回目のステップ S 9 0においては、template1 = {1, 0, 1, 0, 0}, template2 = {1, 0, 1, 0, 0} でそれぞれ一致し、処理は3回目のステップ S 9 2に進む。20

【0 0 7 2】

図 1 0 の例では、3回目のステップ S 9 2においては、 $i = 3$ とされ、5回目のステップ S 8 8以降の処理が実行される。このとき、 $2(\text{MAXSIZE}-\text{tSize}) + 1$ の値は 3 となるので、5回目のステップ S 8 8においては、 $i < 2(\text{MAXSIZE}-\text{tSize}) + 1$ でないと判定され、処理はステップ S 9 3に進む。

【0 0 7 3】

ステップ S 9 3において、プルダウン周期検出部 4 1は、cycleLength = tSizeとする。すなわち、上述してきた図 1 0 の例では、cycleLength = 5 となり、処理はステップ S 9 1に戻る。図 1 0 の例の場合、ステップ S 9 1においてtSize = 4とされて、ステップ S 8 6以降の処理が繰り返され、最終的にtSize = 1となった後、ステップ S 8 6においてtSize > 1でないと判定されて、処理はステップ S 9 4に進むようになる。30

【0 0 7 4】

ステップ S 9 4において、プルダウン周期検出部 4 1は、{his[0], …, his[cycleLength-1]} 内で動きのあるフレーム数(動きフレーム数) motionNumをカウントする。すなわち、図 1 0 の例では、{his[0], …, his[4]} = {0, 0, 1, 0, 1}における「1」の数がカウントされ、motionNum = 2となる。

【0 0 7 5】

そして、ステップ S 9 5において、プルダウン周期検出部 4 1は、プルダウン周期cycleLengthと動きフレーム数 motionNumを出力し、処理は終了する。すなわち、図 1 0 の例では、プルダウン周期として cycleLength = 5、動きフレーム数として motionNum = 2 が出力される。このように、3 - 2 プルダウン処理された入力画像については、プルダウン周期が 5 とされ、動きフレーム数が 2 とされる。40

【0 0 7 6】

一方、ステップ S 8 3において、動き履歴 his[0], …, his[MAXSIZE]の値が全て 0 であると判定された場合、すなわち、入力画像に動きが全くない場合には、処理はステップ S 9 4, S 9 5に進み、cycleLength = 1, motionNum = 0 とされる。

【0 0 7 7】

また、ステップS 8 4において、動き履歴 $his[0], \dots, his[MAXSIZE]$ の値が全て1であると判定された場合、すなわち、入力画像が、所定のプルダウンパターンでプルダウン処理された画像ではなく、いわゆるビデオ素材（オリジナル画像）である場合には、処理はステップS 9 4, S 9 5に進み、 $cycleLength = 1$, $motionNum = 1$ とされる。

【0078】

このように、ユーザによって指定された最大プルダウン周期MAXSIZEを候補プルダウン周期の初期値として、候補プルダウン周期分の動き履歴を1つめのテンプレートtemplate1とし、そのテンプレートをさらに過去に1候補プルダウン周期分ずらした動き履歴を2つめのテンプレートtemplate2として、2つのテンプレートが比較される。

【0079】

また、候補プルダウン周期分のテンプレートの比較の後は、候補プルダウン周期を小さくしていく（ステップS 9 1の処理）、同様のテンプレート比較処理が繰り返され、最終的に、2つのテンプレートが一致した最小の候補プルダウン周期が出力されるようになる。これにより、入力画像が2-2プルダウン処理された画像である場合、動き履歴としては $1, 0, 1, 0, 1, 0, \dots$ となり、候補プルダウン周期としては、 $\dots, 6, 4, 2$ が挙げられるようになるが、最小の候補プルダウン周期として2が出力されるようになる。

【0080】

なお、上述した処理においては、プルダウン周期が小さくなるほど、2つのテンプレートの比較の回数が増えてしまうが、演算量を削減するために、1つの候補プルダウン周期でテンプレートが比較される回数に制限を設けるようにしてもよい。

20

【0081】

さて、図8のフローチャートに戻り、ステップS 5 1において得られたプルダウン周期 $cycleLength$ と動きフレーム数 $motionNum$ は、フレーム周波数算出部4 2に供給される。このとき、プルダウン周期検出部4 1は、動き検出部1 1からの動き検出結果に基づいて、入力画像のフレーム周波数を特定し、フレーム周波数算出部4 2に供給する。

【0082】

そして、ステップS 5 2において、フレーム周波数算出部4 2は、プルダウン周期検出部4 1からのプルダウン周期 $cycleLength$ および動きフレーム数 $motionNum$ と、入力画像のフレーム周波数に基づいて、オリジナル画像のフレーム周波数を算出する。ここで、入力画像のフレーム周波数を f_{in} とすると、プルダウン前のオリジナル画像のフレーム周波数 f_{org} は、以下の式（1）で与えられる。

【0083】

$$f_{org} = f_{in} \times (motionNum / cycleLength) \quad \dots (1)$$

【0084】

例えば、入力画像が、3-2プルダウン処理された、フレーム周波数 $f_{in} = 60Hz$ の画像である場合、図10を参照して説明したように、プルダウン周期 $cycleLength = 5$ 、動きフレーム数 $motionNum = 2$ であるので、式（1）によれば、プルダウン前のオリジナル画像のフレーム周波数 f_{org} は、 $f_{org} = 60 \times (2 / 5) = 24Hz$ と与えられる。このようにして算出（推定）されたオリジナル画像のフレーム周波数は、フレーム制御部1 5に供給される。

30

【0085】

以上の処理によれば、任意のプルダウンパターンでプルダウン処理された入力画像について、入力画像がプルダウンされる前のオリジナル画像のフレーム周波数を推定することが可能となる。

【0086】

特に、現存するプルダウンパターンを全てテーブルに保持する必要がないため、簡単な制御で、かつ、低コストでオリジナル画像のフレーム周波数を推定することが可能となる。また、現存しないプルダウンパターンにも対応できるようになるので、将来的な拡張性を高めることができる。

【0087】

40

50

< 3 . フレームレート変換処理 >

次に、図11のフローチャートを参照して、上述のフレーム周波数推定処理によって推定されたオリジナル画像のフレーム周波数を用いた、画像処理装置1によるフレームレート変換処理について説明する。図11のフレームレート変換処理は、図8のフレーム周波数推定処理が終了した後に実行される。

【0088】

ステップS111において、動き検出部11は、画像処理装置1に入力される現フレームと、フレームメモリ12に記憶されている前フレームとの間の動きの有無を検出し、動き検出結果をフレームメモリ12に供給する。

【0089】

ステップS112において、動きベクトル検出部13は、画像処理装置1に入力される現フレームと、フレームメモリ12に記憶されている前フレームとを用いて、入力画像の動きベクトルを1フレーム毎に検出し、フレームメモリ12に供給する。なお、動きベクトルの検出は、例えば、ブロックマッチング法や、勾配法、位相相關法等によって行われる。

【0090】

ステップS113において、フレーム制御部15は、動き補償部16による動き補償処理に用いられるオリジナル画像のペアフレームを指定するフレーム制御信号を生成し、フレームメモリ12に供給する。

【0091】

ステップS114において、フレーム制御部15は、フレーム周波数推定部14からのオリジナル画像のフレーム周波数と、出力画像のフレーム周波数とに基づいて、動き補償部16による動き補償処理において生成される補間フレームが、オリジナル画像のペアフレーム間で時間的に位置する間隔を表す補間刻み幅を求める。ここで、出力画像のフレーム周波数を f_{out} とすると、補間刻み幅 w は、以下の式(2)によって与えられる。

【0092】

$$w = f_{org} / f_{out} \quad \dots (2)$$

【0093】

例えば、入力画像が、3-2プルダウン処理された、フレーム周波数 $f_{in} = 60Hz$ の画像である場合、上述したように、オリジナル画像のフレーム周波数 f_{org} は24Hzとなる。ここで、出力画像のフレーム周波数 $f_{out} = 120Hz$ とすると、式(2)によれば、補間刻み幅 w は、 $w = 24 / 120 = 0.2$ となる。これは、オリジナル画像のペアフレーム間を1単位時間としたときに、0.2ずつの時間間隔で補間フレームが配置される(出力される)ようになることを示している。

【0094】

ステップS115において、フレーム制御部15は、オリジナル画像のペアフレーム間ににおける時間的な位置を表す補間位相を0とし、動き補償部16に供給する。補間位相は、図12に示されるように、時刻 $t-1$ におけるオリジナル画像のフレーム $f(t-1)$ と、時刻 t におけるオリジナル画像のフレーム $f(t)$ との間ににおける、フレーム $f(t-1)$ からの補間フレームの時間的な位置を表しており、補間フレーム毎に、フレーム $f(t-1)$ から補間刻み幅 w の間隔で変化する。図12においては、補間面と記されている時間的位置に補間フレームが配置される。

【0095】

ステップS116において、フレーム制御部15は、補間位相が1未満であるか否かを判定する。ステップS116において、補間位相が1未満であると判定された場合、処理はステップS117に進み、動き補償部16は、フレームメモリ12に記憶されている、フレーム制御部15に指定されたオリジナル画像のペアフレームと、そのペアフレームに対応する入力画像についての動きベクトルとに基づいて動き補償を行い、フレーム制御部15から供給されたそのときの補間位相で、補間フレームを生成する。

【0096】

10

20

30

40

50

そして、ステップS118において、フレーム制御部15は、補間位相に補間刻み幅を加算し、処理はステップS116に戻る。ステップS116乃至S118の処理、すなわち、フレーム制御部15に指定されたオリジナル画像のペアフレームの動き補償処理は、ステップS116において補間位相が1未満でないと判定されるまで繰り返される。

【0097】

ここで、図13を参照して、オリジナル画像のペアフレームの動き補償処理の例について説明する。図13において、入力画像は、3-2プルダウン処理が行われた、フレーム周波数60Hzの画像とされ、出力画像は、フレーム周波数120Hzの画像とされる。

【0098】

まず、入力画像において、プルダウン周期(cycleLength = 5)に含まれる5フレームA, A, B, B, Bに着目し、フレーム制御部15は、オリジナル画像のペアフレームとして、フレームA, Bを指定するフレーム制御信号を生成する。このとき、フレームメモリ12においては、入力画像がフレーム毎に記憶されているが、図示せぬフレームコントローラの制御の下、動き検出部11からの動き検出結果に基づいて、動きのあったフレームのみを記憶するようにして、動きのないフレーム(例えば、図13の入力画像の2フレーム目のフレームA)を削除し、その記憶領域を次の入力画像(フレームB)で上書きされるようにしてもよい。これにより、フレームメモリ12のメモリ容量を節約することが可能となる。

【0099】

オリジナル画像のペアフレームとしてフレームA, Bが指定されると、動き補償部16は、補間位相0でフレームAを出力する。その後、補間位相に0.2が加算されると、動き補償部16は、補間位相0.2で、フレームA, Bと、フレームA, Bについての動きベクトルとに基づいて動き補償を行った補間フレームA-Bを出力する。さらにその後、補間位相に0.2が加算されると、動き補償部16は、補間位相0.4で、補間フレームA-Bを出力する。このように、補間位相が0.2ずつ加算され、補間位相が1になると、図11のフローチャートのステップ116においては、補間位相が1未満でないと判定されるようになる。

【0100】

図11のフローチャートに戻り、ステップS116において補間位相が1未満でないと判定されると、処理はステップS119に進み、フレーム制御部15は、補間位相から1を減算する。すなわち、補間位相は0になる。

【0101】

そして、ステップS120において、フレーム制御部15は、ペアフレームを更新するフレーム制御信号を生成し、フレームメモリ12に供給する。このフレーム制御信号により、ペアフレームとともに、動き補償に用いられるそのペアフレームに対応する入力画像についての動きベクトルが更新されるようになる。

【0102】

ステップS121において、フレーム制御部15は、動き補償処理されていないペアフレームがフレームメモリ12にあるか否かを判定する。動き補償処理されていないペアフレームがフレームメモリ12にあると判定された場合、処理はステップS117に戻り、更新されたペアフレームについて、ステップS116乃至S118の処理、すなわち動き補償処理が繰り返される。

【0103】

すなわち、図13に示されるように、オリジナル画像のペアフレームとしてフレームB, Cが指定されると、動き補償部16は、補間位相0で画像Bを出力する。その後、補間位相に0.2が加算されると、動き補償部16は、補間位相0.2で、フレームB, Cと、フレームB, Cについての動きベクトルとに基づいて動き補償を行った補間フレームB-Cを出力する。さらにその後、補間位相に0.2が加算されると、動き補償部16は、補間位相0.4で、補間フレームB-Cを出力する。そして、補間位相が1になるまで、補間位相が0.2ずつ加算され、補間フレームが出力される。

10

20

30

40

50

【 0 1 0 4 】

このようにして、3-2 プルダウン処理が行われたフレーム周波数60Hzの入力画像から、プルダウン前のオリジナル画像が動き補償処理に用いられる画像として指定され、そのオリジナル画像（ペアフレーム）に基づいて補間フレームが生成されて、フレーム周波数120Hzの出力画像が出力されるようになる。なお、補間位相が0のときにはオリジナル画像が出力されるようになる。

【 0 1 0 5 】

一方、ステップS121において、動き補償処理されていないペアフレームがフレームメモリ12にないと判定された場合、処理は終了する。

【 0 1 0 6 】

以上の処理によれば、所定のプルダウンパターンでプルダウン処理された入力画像をフレームレート変換する際に、プルダウン前のオリジナル画像のフレーム周波数が推定されているので、プルダウン前のオリジナル画像に基づいて動き補償を行い、補間フレームを生成することが可能となる。すなわち、任意のプルダウンパターンの入力画像をフレームレート変換する際に、現存する全てのプルダウンパターンを保持したテーブルを用いることなく、簡単な構成でジャダーを低減することが可能となる。

10

【 0 1 0 7 】

なお、フレーム周波数推定処理を、フレームレート変換処理と並行して実行させることで、入力画像のフレームレートが変化した場合、すなわち、補間刻み幅が変化するような場合には、フレームレート変換処理を最初から実行させるようにしてもよい。また、画像処理装置1に、入力画像のシーンチェンジを検出する構成を設けるようにして、シーンチェンジが検出された場合には、フレームレート変換処理を最初から実行せるようにしてもよい。

20

【 0 1 0 8 】

また、以上においては、入力画像のフレーム周波数が60Hzで、出力画像のフレーム周波数が120Hzであるものとしたが、それぞれ、その他のフレーム周波数であってもよい。

【 0 1 0 9 】

さらに、本技術を、左眼用画像と右眼用画像とを出力することで立体画像を表示させる立体画像表示システムに適用することで、左眼用画像および右眼用画像が所定のプルダウンパターンでプルダウン処理された画像であっても、左眼用画像および右眼用画像のいずれか一方について動き検出を行い、プルダウン前のフレーム周波数を推定することが可能となる。さらに、左眼用画像および右眼用画像のそれについて動き検出を行い、プルダウン前のフレーム周波数を推定するようにしてもよい。

30

【 0 1 1 0 】

また、以上においては、動き検出として、フレーム間の輝度値の差分を用いるようにしたが、フレーム間の差分を検出できる特徴量であれば、動きベクトルの絶対値和や平均輝度レベル、色差信号の差分等を用いるようにしてもよい。また、これらの特徴量を複数組み合わせて用いるようにしてもよい。

【 0 1 1 1 】

さらに、以上においては、画像処理装置1に入力される入力信号は、プログレッシブ信号であるものとしたが、デジタルテレビジョン放送信号のようなインターレース信号が入力される場合には、前段に、インターレース・プログレッシブ(IP)変換を行うIP変換器を設けるようにして、プログレッシブ信号が入力されるようにしてもよい。

40

【 0 1 1 2 】

また、異なるフレーム周波数が混在する入力信号が入力される場合には、前段に、所定のプルダウンパターンでプルダウン処理を行うプルダウン処理部を設けるようにして、フレーム周波数を統一させた入力信号が入力されるようにしてもよい。

【 0 1 1 3 】

上述した一連の処理は、ハードウェアにより実行することもできるし、ソフトウェアにより実行することもできる。一連の処理をソフトウェアにより実行する場合には、そのソ

50

ソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行することができる、例えば汎用のパーソナルコンピュータ等に、プログラム記録媒体からインストールされる。

【0114】

図14は、上述した一連の処理をプログラムにより実行するコンピュータのハードウェアの構成例を示すプロック図である。

【0115】

コンピュータにおいて、CPU(Central Processing Unit)901, ROM(Read Only Memory)902, RAM(Random Access Memory)903は、バス904により相互に接続されている。
10

【0116】

バス904には、さらに、入出力インターフェース905が接続されている。入出力インターフェース905には、キーボード、マウス、マイクロホン等よりなる入力部906、ディスプレイ、スピーカ等よりなる出力部907、ハードディスクや不揮発性のメモリ等よりなる記憶部908、ネットワークインターフェース等よりなる通信部909、磁気ディスク、光ディスク、光磁気ディスク、或いは半導体メモリ等のリムーバブルメディア911を駆動するドライブ910が接続されている。

【0117】

以上のように構成されるコンピュータでは、CPU901が、例えば、記憶部908に記憶されているプログラムを、入出力インターフェース905およびバス904を介して、RAM903にロードして実行することにより、上述した一連の処理が行われる。
20

【0118】

コンピュータ(CPU901)が実行するプログラムは、例えば、磁気ディスク(フレキシブルディスクを含む)、光ディスク(CD-ROM(Compact Disc-Read Only Memory), DVD(Digital Versatile Disc)等)、光磁気ディスク、もしくは半導体メモリ等よりなるパッケージメディアであるリムーバブルメディア911に記録して、あるいは、ローカルエリアネットワーク、インターネット、デジタル衛星放送といった、有線または無線の伝送媒体を介して提供される。

【0119】

そして、プログラムは、リムーバブルメディア911をドライブ910に装着することにより、入出力インターフェース905を介して、記憶部908にインストールすることができる。また、プログラムは、有線または無線の伝送媒体を介して、通信部909で受信し、記憶部908にインストールすることができる。その他、プログラムは、ROM902や記憶部908に、あらかじめインストールしておくことができる。

【0120】

なお、コンピュータが実行するプログラムは、本明細書で説明する順序に沿って時系列に処理が行われるプログラムであっても良いし、並列に、あるいは呼び出しが行われたとき等の必要なタイミングで処理が行われるプログラムであっても良い。

【0121】

また、本技術の実施の形態は、上述した実施の形態に限定されるものではなく、本技術の要旨を逸脱しない範囲において種々の変更が可能である。

【0122】

さらに、本技術は以下のよう構成をとることができる。

(1) プルダウンされた入力画像におけるプルダウンパターンを検出するプルダウンパターン検出部と、

前記プルダウンパターン、および前記入力画像のフレーム周波数である第1のフレーム周波数に基づいて、前記入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出するフレーム周波数算出部と

を備える画像処理装置。

10

20

30

40

50

(2) 前記プルダウンパターン検出部は、前記入力画像のフレーム間の動きの有無のパターンに基づいて、前記プルダウンパターンが繰り返されるフレームの周期であるプルダウン周期を検出するとともに、前記プルダウン周期におけるフレーム間の動きの数を表す動きフレーム数をカウントし、

前記フレーム周波数算出部は、前記プルダウン周期、前記動きフレーム数、および前記第1のフレーム周波数に基づいて、前記第2のフレーム周波数を算出する

(1)に記載の画像処理装置。

(3) 前記原画像のフレーム間の動きベクトルを検出する動きベクトル検出部と、

前記動きベクトル、前記第2のフレーム周波数、および、出力画像のフレーム周波数である第3のフレーム周波数に基づいて動き補償を行い、前記原画像の補間フレームを生成する動き補償部とをさらに備える

(1)または(2)に記載の画像処理装置。

(4) 前記第2のフレーム周波数および前記第3のフレーム周波数に基づいて、前記原画像のフレーム間における前記補間フレームの時間的な位置を表す補間位相を求める補間位相決定部をさらに備え、

前記動き補償部は、前記原画像のフレーム間における前記補間位相で、前記補間フレームを生成する

(3)に記載の画像処理装置。

(5) プルダウンされた入力画像におけるプルダウンパターンを検出するプルダウンパターン検出ステップと、

前記プルダウンパターン、および前記入力画像のフレーム周波数である第1のフレーム周波数に基づいて、前記入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出するフレーム周波数算出ステップと

を含む画像処理方法。

(6) プルダウンされた入力画像におけるプルダウンパターンを検出するプルダウンパターン検出ステップと、

前記プルダウンパターン、および前記入力画像のフレーム周波数である第1のフレーム周波数に基づいて、前記入力画像がプルダウンされる前の原画像のフレーム周波数である第2のフレーム周波数を算出するフレーム周波数算出ステップと

を含む処理をコンピュータに実行させるプログラム。

10

20

30

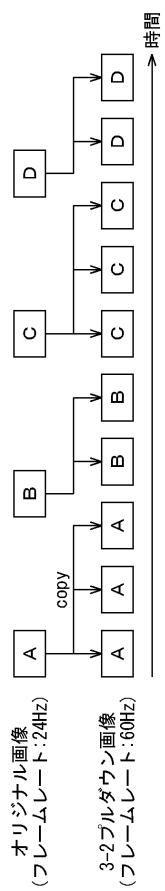
【符号の説明】

【0 1 2 3】

1 画像処理装置， 11 動き検出部， 13 動きベクトル検出部， 14 フレーム周波数推定部， 15 フレーム制御部， 16 動き補償部， 41 プルダウン周期検出部， 42 フレーム周波数算出部

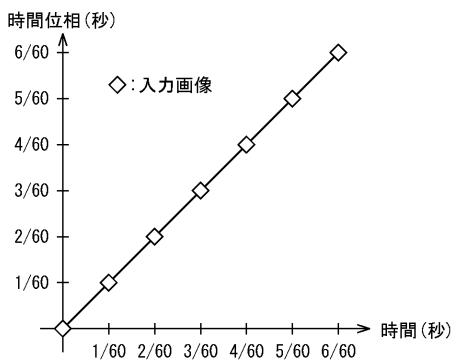
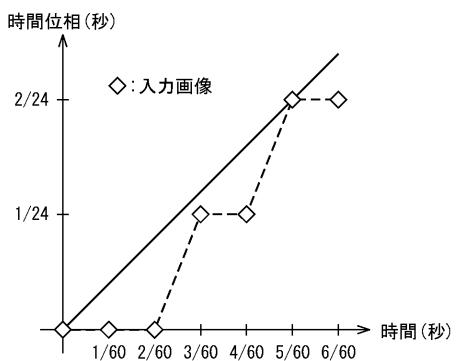
【図1】

図1



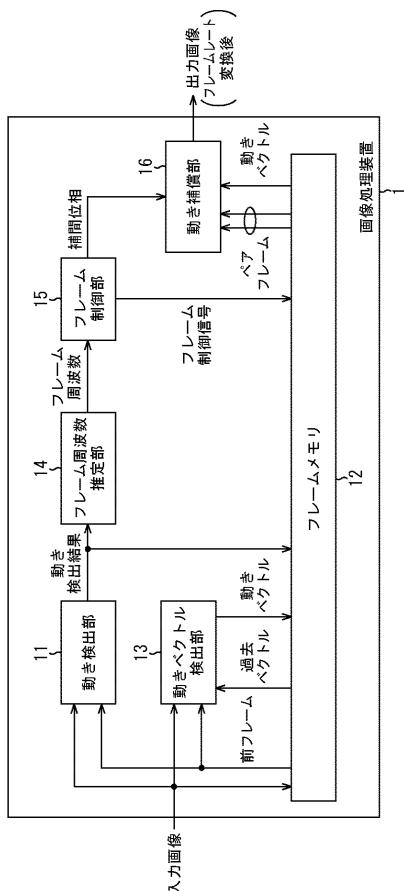
【図2】

図2



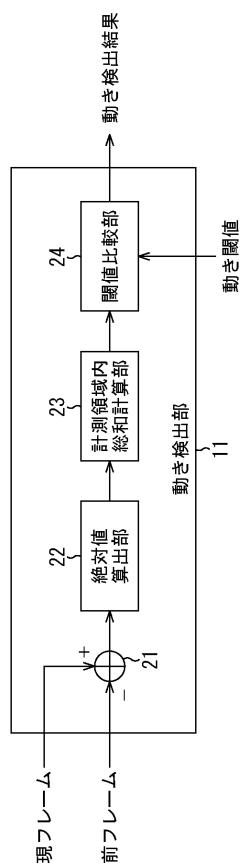
【図3】

図3



【図4】

図4



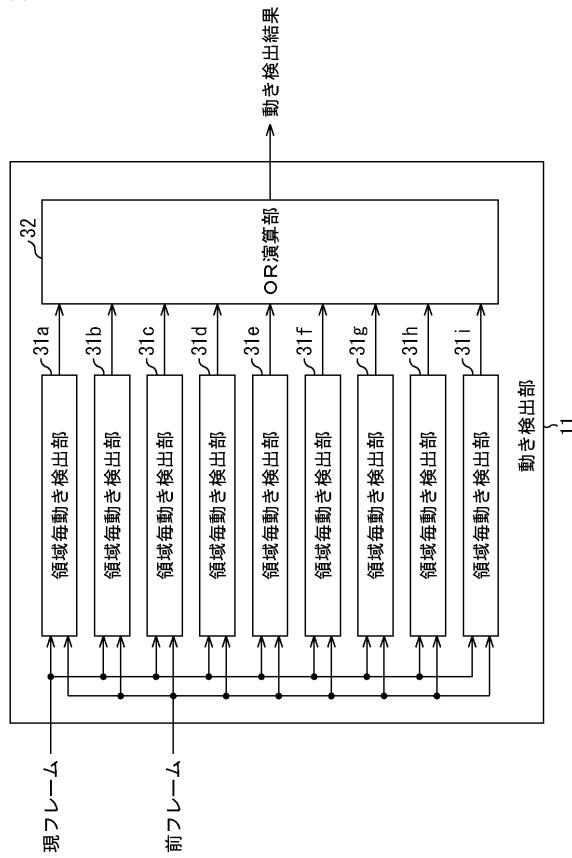
【図5】

図5

a	b	c
d	e	f
g	h	i

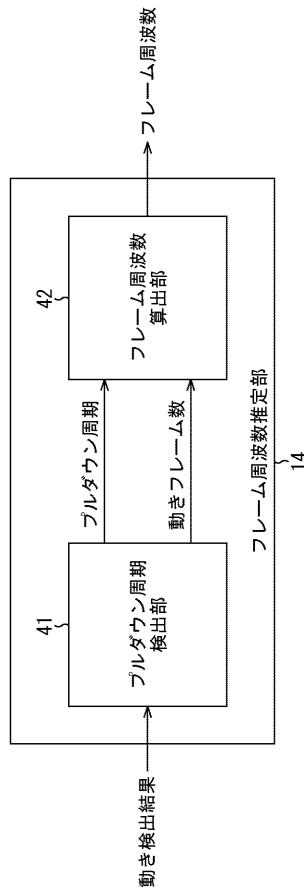
【図6】

図6



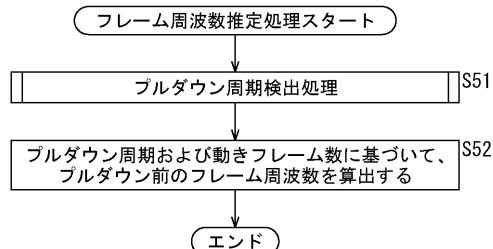
【図7】

図7

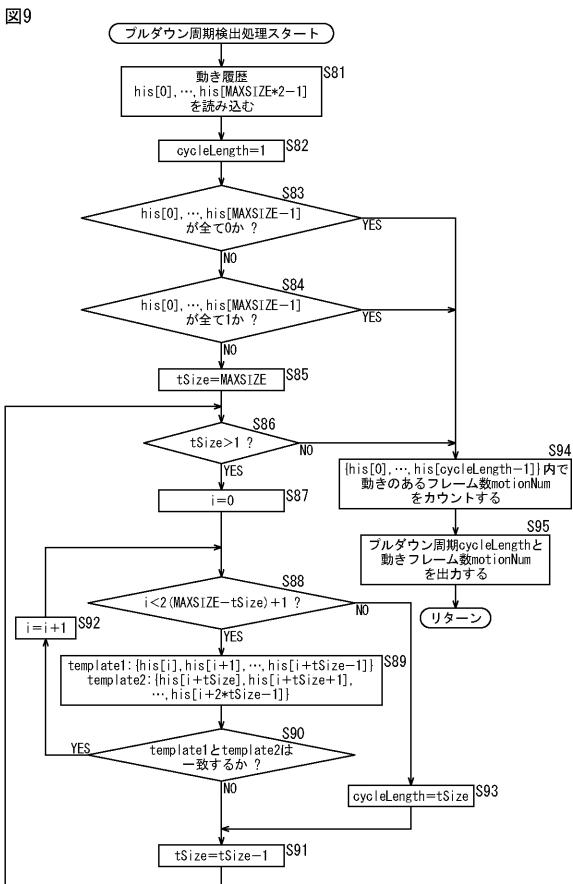


【図8】

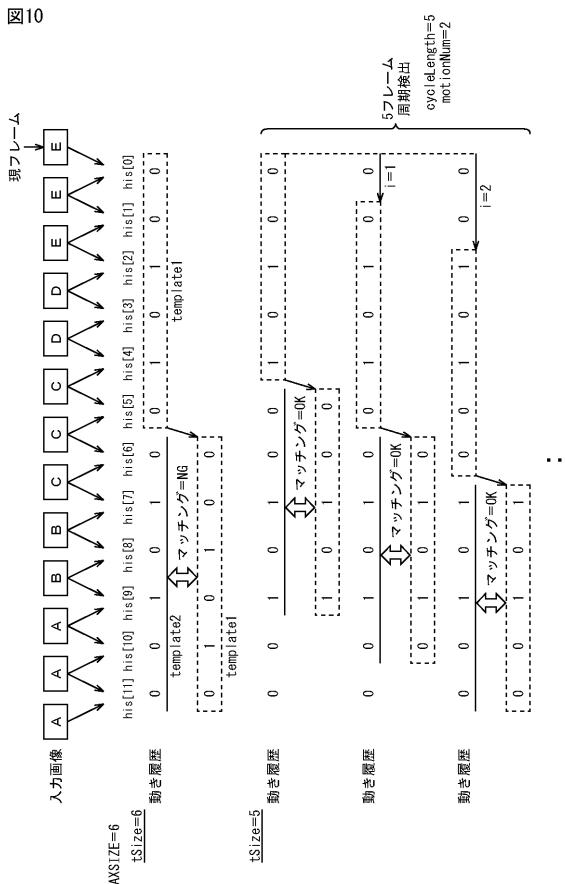
図8



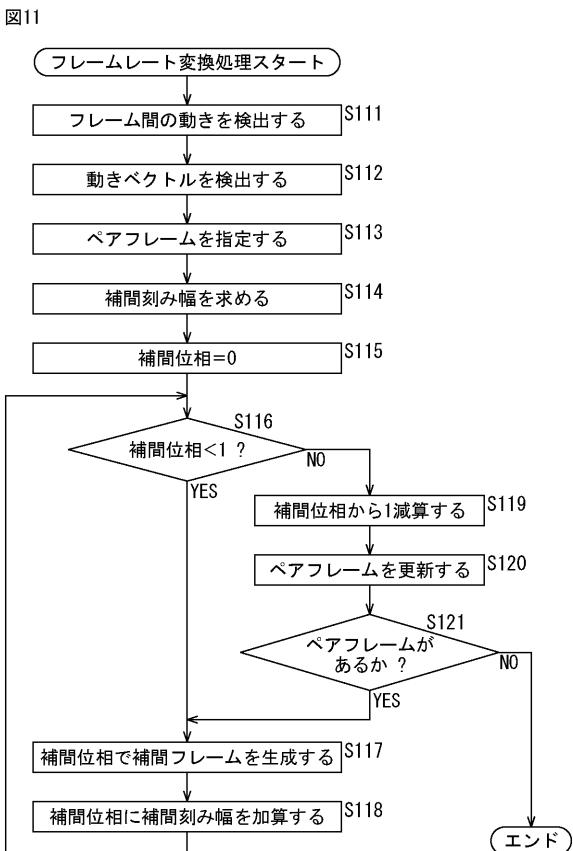
【図9】



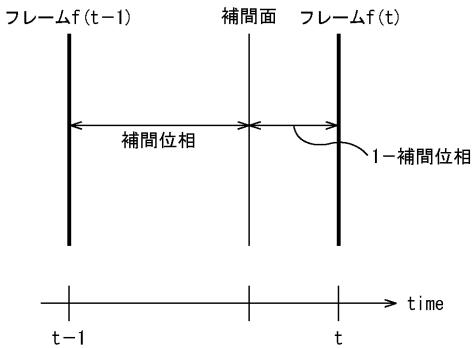
【図10】



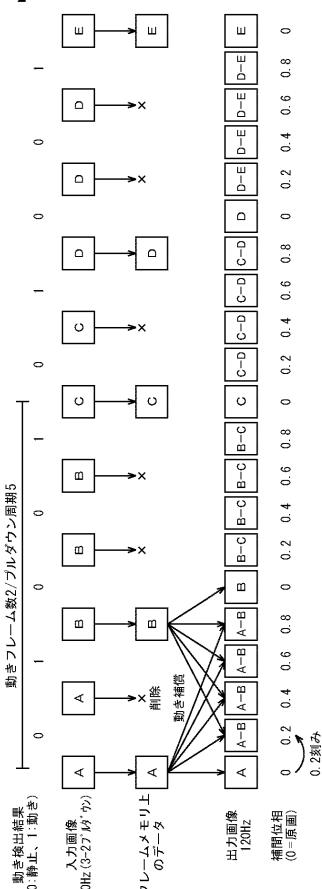
【図11】



【図12】



【 図 1 3 】



【 図 1 4 】

