(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第5277180号
(P5277180)
(45)発行日　平成25年8月28日(2013.8.28)　(24)登録日　平成25年5月24日(2013.5.24)

(51)Int.Cl.　F I
H04N 7/26 (2006.01)　H04N 7/13 Z

請求項の数 38 (全 34 頁)

(21)出願番号	特願2009-554500 (P2009-554500)
(86)(22)出願日	平成19年11月2日 (2007.11.2)
(65)公表番号	特表2010-522470 (P2010-522470A)
(43)公表日	平成22年7月1日 (2010.7.1)
(86)国際出願番号	PCT/US2007/023172
(87)国際公開番号	WO2008/118145
(87)国際公開日	平成20年10月2日 (2008.10.2)
審査請求日	平成22年10月8日 (2010.10.8)
(31)優先権主張番号	60/919,702
(32)優先日	平成19年3月23日 (2007.3.23)
(33)優先権主張国	米国(US)
(31)優先権主張番号	60/934,634
(32)優先日	平成19年6月14日 (2007.6.14)
(33)優先権主張国	米国(US)

(73)特許権者　501263810
トムソン　ライセンシング
Ｔｈｏｍｓｏｎ　Ｌｉｃｅｎｓｉｎｇ
フランス国，　９２１３０　イツシー　レ　ムーリノー，　ル　ジヤンヌ　ダルク，　１－５
１－５，　ｒｕｅ　Ｊｅａｎｎｅ　ｄ’Ａｒｃ，　９２１３０　ＩＳＳＹ　ＬＥＳ　ＭＯＵＬＩＮＥＡＵＸ，　Ｆｒａｎｃｅ
(74)代理人　100070150
弁理士　伊東　忠彦
(74)代理人　100091214
弁理士　大貫　進介
(74)代理人　100107766
弁理士　伊東　忠重

最終頁に続く

(54)【発明の名称】符号化ビットストリームの修正

(57)【特許請求の範囲】
【請求項１】
処理装置によって実行される方法であって：
少なくとも第一の部分および第二の部分を含む算術エンコードされたデータにアクセスする段階であって、前記算術エンコードされたデータは、前記第二の部分のデコードが前記第二の部分のほか前記第一の部分にも依存するような仕方で算術符号でエンコードされたデータである、段階と；
少なくとも前記第一の部分および前記第二の部分を使って、前記第一の部分を置き換えるために使うことができる修正された第一の部分を決定する段階であって、前記修正された第一の部分は、前記第二の部分が、前記第二の部分および前記修正された第一の部分に基づいて、前記修正された第一の部分の代わりに前記第一の部分が使われたとした場合と同じ結果を生成するようデコードされることができるよう決定される、段階とを有しており、
前記決定する段階が、前記第一の部分についての少なくとも一つの状態値と同じ少なくとも一つのCABAC状態値を生成する修正された第一の部分を決定することを含む、
方法。
【請求項２】
前記第一の部分を前記修正された第一の部分で置き換える段階をさらに有する、請求項１記載の方法。
【請求項３】

前記修正された第一の部分が、前記第一の部分のビット長と等しいビット長をもつ値を有する、請求項１記載の方法。

【請求項４】

前記第一の部分のデコードが第一の結果を生成し、前記修正された第一の部分のデコードが、前記第一の結果とは異なる修正された第一の結果を生成する、請求項１記載の方法。

【請求項５】

前記修正された第一の部分を決定する段階が、前記第一の修正された第一の部分を、前記修正された第一の結果が、前記第一の結果と比べて（１）見る人にとっては知覚不能であり、かつ（２）処理装置によって検出可能である違いをもつよう決定する、請求項４記載の方法。

【請求項６】

前記修正された第一の部分が、前記第一の部分を前記修正された第一の部分で置き換えることによってエンコードされたデータに透かし入れするために好適である、請求項５記載の方法。

【請求項７】

前記第一の部分のデコードの結果が、前記第一の部分のエンコードから生成される状態に基づく、請求項１記載の方法。

【請求項８】

前記第一の部分が、CABAC符号化されたビットストリームにおけるシンタックス要素を含む、請求項１記載の方法。

【請求項９】

請求項８記載の方法であって、前記第一の部分を置き換えるために使うことができる修正された第一の部分を決定する段階がさらに：

前記シンタックス要素についてあらゆる可能な代替的な置き換え値を決定する段階と；

可能な代替的な置き換え値それぞれについて、その置き換え値が前記シンタックス要素について同じ状態変数値を生成するかどうかを判定する段階と；

前記シンタックス要素についての同じ状態変数値を生成する代替的な置き換え値のリストを保存する段階とを有する、

方法。

【請求項１０】

請求項９記載の方法であって、前記代替的な置き換え値を、該置き換え値が同じ状態変数を生成するかどうかを判定する前にエンコードする段階をさらに含む、請求項１０記載の方法。

【請求項１１】

前記第一の部分が符号化されたシンタックス要素を含み、前記修正された第一の部分が前記第一の部分のビット長と同じビット長をもつ値を含み、前記修正された第一の部分がCABACエンコーダにおいて、前記第一の部分によって生成されたであろうL値と等しいL値を生成する、請求項１記載の方法。

【請求項１２】

前記第二の部分が、それぞれ対応する状態変数値をもつ二つ以上の符号化されたシンタックス要素を含み、前記修正された第一の部分が、その後の第二の部分のすべてのシンタックス要素のすべてのデコードについて前記状態変数値を維持する、請求項１記載の方法。

【請求項１３】

前記修正された第一の部分が前記第二の部分に含まれる他のいかなる符号化されたシンタックス要素の値にも影響しない、請求項１２記載の方法。

【請求項１４】

請求項１記載の方法であって：

前記エンコードされたデータがエントロピー・エンコードされたデータであり、

前記第一の部分がシンタックス要素を含み、

前記修正された第一の部分が前記シンタックス要素の修正を含む、
方法。
【請求項１５】
少なくとも第一の部分および第二の部分を含む算術エンコードされたデータにアクセスするアクセス・ユニットであって、前記算術エンコードされたデータは、第二の部分のデコーダが該第二の部分のほか第一の部分にも依存するような仕方で算術符号でエンコードされたデータである、アクセス・ユニットと；
少なくとも前記第一の部分および前記第二の部分を使って、前記第一の部分を置き換えるために使うことができる修正された第一の部分を、前記第二の部分が、該第二の部分および前記修正された第一の部分に基づいて、前記修正された第一の部分の代わりに前記第一の部分が使われたとした場合と同じ結果を生成するようデコードされることができるよう決定する修正ユニットとを有しており、
前記修正ユニットが、前記第一の部分についての少なくとも一つの状態値と同じ少なくとも一つのCABAC状態値を生成する修正された第一の部分を決定する、
装置。
【請求項１６】
少なくとも第一の部分および第二の部分を含む算術エンコードされたデータにアクセスする手段であって、前記算術エンコードされたデータは、第二の部分のデコードが、該第二の部分のほか第一の部分にも依存するような仕方で算術符号でエンコードされたデータである、手段と；
少なくとも前記第一の部分および前記第二の部分を使って、前記第一の部分を置き換えるために使うことができる修正された第一の部分を、前記第二の部分が該第二の部分および前記修正された第一の部分に基づいて、前記修正された第一の部分の代わりに前記第一の部分が使われたとした場合と同じ結果を生成するようデコードされることができるよう決定する手段とを有しており、
前記決定する手段が、前記第一の部分についての少なくとも一つの状態値と同じ少なくとも一つのCABAC状態値を生成する修正された第一の部分を決定することを含む、
装置。
【請求項１７】
プロセッサ読み取り可能な媒体であって、プロセッサに：
少なくとも第一の部分および第二の部分を含む算術エンコードされたデータにアクセスする段階であって、前記算術エンコードされたデータは、第二の部分のデコードが、該第二の部分のほか第一の部分にも依存するような仕方で算術符号でエンコードされたデータである、段階と；
少なくとも前記第一の部分および前記第二の部分を使って、前記第一の部分を置き換えるために使うことができる修正された第一の部分を、前記第二の部分が該第二の部分および前記修正された第一の部分に基づいて、前記修正された第一の部分の代わりに前記第一の部分が使われたとした場合と同じ結果を生成するようデコードされることができるよう決定する段階とを少なくとも実行させるための前記プロセッサ読み取り可能な媒体上に記憶された命令を含んでおり、
前記決定する段階が、前記第一の部分についての少なくとも一つの状態値と同じ少なくとも一つのCABAC状態値を生成する修正された第一の部分を決定することを含む、
プロセッサ読み取り可能な媒体。
【請求項１８】
処理装置によって実行される方法であって：
少なくとも第一の部分および第二の部分を含む算術符号化されたビットストリームにアクセスする段階であって、前記算術符号化されたビットストリームは、前記第二の部分のデコードが、前記第二の部分のほか前記第一の部分にも依存するような仕方で算術符号で符号化されたデータを含む、段階と；
前記第一の部分を、少なくとも前記第一の部分および前記第二の部分に基づいて決定さ

れた置き換え値で置き換えて修正された符号化ビットストリームを生成する段階であって、修正された符号化ビットストリームについても前記第二の部分のデコードはやはり、前記修正された符号化ビットストリームにおいて前記置き換え値の代わりに前記第一の部分が使われたとした場合と同じ結果を生成し、前記置き換え値は、前記第一の部分についての少なくとも一つの状態値と同じ少なくとも一つのCABAC状態値を生成する、段階とを有する
方法。
【請求項１９】
前記第一の部分の位置の指標にアクセスする段階と；
前記第一の部分についての置き換え値にアクセスする段階とをさらに有する、請求項１８記載の方法であって、
前記置き換え値は、前記第一の部分を該置き換え値で置き換えた結果生じる修正された符号化ビットストリームが、前記第二の部分のデコードがやはり前記修正されたビットストリームにおいて前記置き換え値の代わりに前記第一の部分が使われたとした場合と同じ結果を生成するようなものである、方法。
【請求項２０】
前記第一の部分が置き換えられるべきか否かを指定する、および／または、前記置き換え値を含む複数の潜在的な置き換え値のうちのどれが前記第一の部分を置き換えるべきかを指定する情報にアクセスすることをさらに含む、請求項１８記載の方法であって、前記第一の部分を置き換える段階が、前記アクセスされた情報に基づいて前記第一の部分を置き換えることを含む、方法。
【請求項２１】
前記第一の部分を置き換える段階が前記符号化されたビットストリームへの透かし入れの一部であり、前記アクセスされた情報が前記透かしのペイロード情報を含む、請求項２０記載の方法。
【請求項２２】
前記算術符号がCABACである、請求項１８記載の方法。
【請求項２３】
前記置き換え値および前記第一の部分が同じCABAC状態値を生成する、請求項２２記載の方法。
【請求項２４】
前記のアクセスされた符号化されたビットストリームがCABAC符号化されたビットストリームであり、前記第一の部分が前記CABAC符号化されたビットストリーム中の少なくとも一つの符号化されたシンタックス要素を含む、請求項２２記載の方法。
【請求項２５】
前記第二の部分が二つ以上のシンタックス要素を含み、前記二つ以上のシンタックス要素のそれぞれが、前記修正された符号化ビットストリームのデコードの際に生成される対応する状態変数値を有しており、
前記第一の部分についての前記置き換え値が、前記二つ以上のシンタックス要素のそれぞれについての状態変数値を、前記修正された符号化ストリームにおいて前記置き換え値の代わりに前記第一の部分が使用されたとした場合と同じに維持する、
請求項１８記載の方法。
【請求項２６】
プロセッサ読み取り可能な媒体であって、プロセッサに：
少なくとも第一の部分および第二の部分を含む算術符号化されたビットストリームにアクセスする段階であって、前記算術符号化されたビットストリームは、第二の部分のデコードが、該第二の部分のほか第一の部分にも依存するような仕方で算術符号でエンコードされたデータを含む、段階と；
前記第一の部分を、少なくとも前記第一の部分および前記第二の部分に基づいて決定された置き換え値で置き換えて修正された符号化ビットストリームを生成する段階であって

、前記修正された符号化ビットストリームについても前記第二の部分のデコードはやはり、前記修正された符号化ビットストリームにおいて前記置き換え値の代わりに前記第一の部分が使われたとした場合と同じ結果を生成し、前記置き換え値は、前記第一の部分についての少なくとも一つの状態値と同じ少なくとも一つのCABAC状態値を生成する、段階、とを少なくとも実行させるための、
前記プロセッサ読み取り可能な媒体上に記憶された命令を含む、
プロセッサ読み取り可能な媒体。

【請求項２７】

処理装置によって実行される方法であって：

算術符号で符号化されたデータを含む修正されたデータのセットにアクセスする段階であって、前記修正されたデータのセットは第二の部分および修正された第一の部分を含み、前記修正された第一の部分は第一の部分の修正から帰結するものであり、前記第二の部分のデコードは、前記第二の部分のほか前記修正された第一の部分にも依存する、段階と；

前記第二の部分をデコードして結果を生成する段階であって、前記結果は、前記第二の部分のデコードが、前記第二の部分および前記修正された第一の部分ではなく前記第二の部分および前記第一の部分に基づいて実行されたとした場合と同じである、段階と；

前記修正された第一の部分をデコードする段階とを有し、

前記修正された第一の部分は、前記第一の部分についての少なくとも一つの状態値と同じ少なくとも一つのCABAC状態値を生成する、

方法。

【請求項２８】

前記修正された第一の部分が、前記第一の部分が置き換えられるべきか否かを指定する、および／または、前記修正された第一の部分を含む複数の潜在的な置き換え値のうちのどれが前記第一の部分を置き換えるべきかを指定する情報の値に基づいてなされた修正を反映しており、当該方法がさらに、前記修正された第一の部分のデコードに基づいて前記情報の値を判別する段階を有する、請求項２７記載の方法。

【請求項２９】

前記修正された第一の部分が前記修正されたデータのセットに含まれる透かしの少なくとも一部である、請求項２７記載の方法。

【請求項３０】

前記のアクセスされた修正されたデータのセットが、CABACエンコードされたビットストリームを含み、前記第一の部分が少なくとも一つの符号化されたシンタックス要素を含む、請求項２７記載の方法。

【請求項３１】

前記修正された第一の部分が修正されたシンタックス要素を含み、前記修正されたシンタックス要素が、前記CABACエンコードされたビットストリームについて前記第一の部分と同じデコード変数を与える、請求項２７記載の方法。

【請求項３２】

前記修正された第一の部分が、前記第一の部分のビット長に等しいビット長をもつ修正されたシンタックス要素を含み、前記修正された第一の部分が、前記第一の部分によって生成されたであろうL値に等しいCABACエンコーダにおけるL値を与える、請求項３０記載の方法。

【請求項３３】

プロセッサ読み取り可能な媒体であって、プロセッサに：

算術符号で符号化されたデータを含む修正されたデータのセットにアクセスする段階であって、前記修正されたデータのセットは第二の部分および修正された第一の部分を含み、前記修正された第一の部分は第一の部分の修正から帰結するものであり、前記第二の部分のデコードは、前記第二の部分のほか前記修正された第一の部分にも依存する、段階と；

前記第二の部分をデコードして結果を生成する段階であって、前記結果は、前記第二の部分のデコードが、前記第二の部分および前記修正された第一の部分ではなく前記第二の部分および前記第一の部分に基づいて実行されたとした場合と同じである、段階と；
前記修正された第一の部分をデコードする段階とを少なくとも実行させるための、前記プロセッサ読み取り可能な媒体上に記憶された命令を含み、前記修正された第一の部分は、前記第一の部分についての少なくとも一つの状態値と同じ少なくとも一つのCABAC状態値を生成する、
プロセッサ読み取り可能な媒体。
【請求項３４】
請求項４記載の方法であって、前記第一の結果は第一の画像の一部であり、前記修正された第一の結果は修正された第一の画像の一部であり、当該方法はさらに：
前記修正された第一の部分を、前記第一の画像に一つまたは複数の透かし入れビットを適用するのに使うために提供する段階と；
前記修正された第一の画像から前記一つまたは複数の透かし入れビットを回復するのに使うために前記修正された第一の結果の位置を同定する位置情報を提供する段階とを含む、
方法。
【請求項３５】
前記位置情報がフレーム番号およびブロック番号を含む、請求項３４記載の方法。
【請求項３６】
請求項１５記載の装置であって、前記第一の部分のデコードが第一の画像の一部である第一の結果を生成し、前記修正された第一の部分のデコードが前記第一の結果とは異なり、修正された第一の画像の一部であるる修正された第一の結果を生成し、当該装置がさらに：
（ｉ）前記第一の画像に一つまたは複数の透かし入れビットを適用するのに使うための前記修正された第一の部分と、（ｉｉ）前記修正された第一の画像から前記一つまたは複数の透かし入れビットを回復するのに使うための、前記修正された第一の結果の位置を同定する位置情報とを記憶するメモリをさらに有する、
装置。
【請求項３７】
符号化されたデータを置き換える装置であって：
少なくとも第一の部分および第二の部分を含む算術符号化されたビットストリームにアクセスするアクセス・ユニットであって、前記算術符号化されたビットストリームは、前記第二の部分のデコードが、前記第二の部分のほか前記第一の部分にも依存するような仕方で算術符号で符号化されたデータを含む、アクセス・ユニットと；
前記第一の部分を、少なくとも前記第一の部分および前記第二の部分に基づいて決定された置き換え値で置き換えて修正された符号化ビットストリームを生成する置き換えユニットであって、修正された符号化ビットストリームについても前記第二の部分のデコードはやはり、前記修正された符号化ビットストリームにおいて前記置き換え値の代わりに前記第一の部分が使われたとした場合と同じ結果を生成し、前記置き換え値は、前記第一の部分についての少なくとも一つの状態値と同じ少なくとも一つのCABAC状態値を生成する、置き換えユニットとを有する
装置。
【請求項３８】
（ｉ）前記置き換え値と、（ｉｉ）前記置き換え値によって示される一つまたは複数の透かし入れビットを回復するのに使うための、デコードされた画像における位置を同定する位置情報とを記憶するメモリをさらに有する、請求項３７記載の装置。

【発明の詳細な説明】
【技術分野】

【０００１】
〈関連出願への相互参照〉
本願は、（１）2007年3月23日に出願された「符号化ビットストリームの修正」と題する米国仮出願第60/919,702号（代理人整理番号PU070063）および（２）2007年6月14日に出願された「符号化ビットストリームの修正」と題する米国仮出願第60/934,634号（代理人整理番号PU070137）の両方の利益を主張する。これらの優先権出願はいずれもここに参照によってその全体において組み込まれる。
【０００２】
〈技術分野〉
本開示は、たとえば符号化に関係する実装を提供する。
【背景技術】
【０００３】
透かし入れは典型的には、後刻透かし〔ウォーターマーク〕が検出できるような仕方でデータの一部を修正することに関わる。符号化されたデータを含め、さまざまな異なる型のデータに透かし入れできる。
【発明の概要】
【発明が解決しようとする課題】
【０００４】
しかしながら、符号化方式は進歩しつつあり、符号化データを修正するための既存の機構はそのような進歩する符号化方式のための所望される透かし入れ機能を提供しないことがありうる。
【課題を解決するための手段】
【０００５】
ある一般的な側面によれば、少なくとも第一の部分および第二の部分を含むエンコードされたデータがアクセスされる。第二の部分は、第二の部分のほか第一の部分にも基づく結果を生成するようデコードされることができる。修正された第一の部分が決定されるが、それは、第二の部分が、第二の部分および修正された第一の部分に基づいて前記結果を生成するようデコードされることができるよう決定される。
【０００６】
別の一般的な側面によれば、ある情報が、エンコードされたデータのセットの第一の部分についての置き換え値を特定する。前記エンコードされたデータのセットは第二の部分も含み、第二の部分のデコードは第二の部分および第一の部分に基づく。置き換え値は：（１）第一の部分を含む前記エンコードされたデータのセットに対してデコードが実行されると、第二の部分のデコードが特定のデコード結果を生成し、（２）第一の部分の代わりに前記置き換え値を含む前記エンコードされたデータのセットに対してデコードが実行されると、第二の部分のデコードが前記特定のデコード結果を生成するような属性をもつ。
【０００７】
別の一般的な側面によれば、少なくとも第一の部分および第二の部分を含む符号化されたビットストリームがアクセスされ、第二の部分は、第二の部分のほか第一の部分にも基づく結果を生成するようデコードされることができる。第一の部分は、置き換え値で置き換えられて修正された符号化ビットストリームを生成する。修正された符号化ビットストリームについても第二の部分のデコードはやはり前記の結果を生成する。
【０００８】
別の一般的な側面によれば、エンコードされたデータが修正された第一の部分および第二の部分を含み、修正された第一の部分は第一の部分の置き換えである。第二の部分は、第二の部分のほか修正された第一の部分にも基づく結果を生成するようデコードされることができる。その結果は、第二の部分および修正された第一の部分ではなく第二の部分および第一の部分に基づいて第二の部分のデコードが実行された場合と同じである。
【０００９】

別の一般的な側面によれば、修正された第一の部分および第二の部分を含む修正されたデータのセットがアクセスされる。修正された第一の部分は第一の部分の修正から帰結する。第二の部分はデコードされてある結果を生成することができ、その結果は第二の部分のほか修正された第一の部分にも基づく。第二の部分は前記結果を生成するようデコードされる。ここで、前記結果は、第二の部分のデコードが、第二の部分および修正された第一の部分ではなく第二の部分および第一の部分に基づいて実行された場合と同じである。修正された第一の部分がデコードされる。

【００１０】

一つまたは複数の実装の詳細が付属の図面および以下の記述において述べられる。ある特定の仕方で記載されているとしても、実装はさまざまな仕方で構成され、あるいは具現されうることは明らかなはずである。たとえば、実装は方法として実行されてもよいし、一組の動作を実行するよう構成された装置として具現されてもよいし、一組の動作を実行するための命令を記憶する装置として具現されてもよいし、あるいは信号において具現されてもよい。他の側面および特徴は、付属の図面および請求項との関連で考察される以下の詳細な記述から明白となるであろう。

【図面の簡単な説明】

【００１１】

【図１】エントロピー符号化されたストリーム透かし入れの実装のブロック図である。

【図２ａ】エンコードされたビットストリームを修正する際に使われる動作の順序付けを示すブロック図である。

【図２ｂ】エンコードされたビットストリームを修正する際に使われる動作の別の順序付けを示すブロック図である。

【図２ｃ】エンコードされたビットストリームを修正する方法のブロック図である。

【図２ｄ】エンコードされたビットストリームを修正する別の方法のブロック図である。

【図３】目標シンタックス要素を修正する方法の流れ図である。

【図４】バイパス・モードとしてエンコードされている目標シンタックス要素を修正する方法の流れ図である。

【図５】目標シンタックス要素を修正する別の方法の流れ図である。

【図６】符号化されたビットのブロックを置き換える方法の流れ図である。

【図７】修正されたビットストリームをデコードする方法の流れ図である。

【図８】透かし情報を検出する方法の流れ図である。

【図９】透かし情報を検出する別の方法の流れ図である。

【発明を実施するための形態】

【００１２】

少なくとも一つの実装は、エントロピー・デコード段階および再エンコード段階を必要とすることなく直接、エントロピー・エンコードされたビットストリームを修正する。さらに、エントロピー符号が算術符号である、より特定的にはコンテキスト・ベースの適応二値算術符号化（CABAC: Context-based Adaptive Binary Arithmetic Coding）として知られる算術符号化方式が使われる実装が提供される。CABACは、後の符号化されたシーケンスが前の符号化されたシーケンスに依存するという点で「メモリー」のある符号化方式として特徴付けることができる。よって、その実装では、所与の符号化シーケンスは、後の符号化シーケンスが相変わらず正しくデコードされることができるような仕方で修正される。さらに、エントロピー・デコード段階および再エンコード段階を必要とすることなく、エントロピー・エンコードされたビットストリームに直接、電子透かしを適用する実装が提供される。少なくとも一つの透かし入れ実装は、CABACエンコードされたビットストリームを修正する。CABACはH.264/AVC規格において広く使用されている。

【００１３】

多くの用途では、圧縮されたビデオ・ストリームの透かし入れに対する、増加し続ける市場の要求がある。初期のアプローチはストリームを圧縮解除し、ピクセル領域で透かし入れを適用し、それから再圧縮していた。初期の進歩は、もとの圧縮されたストリームか

らの情報を使って再圧縮を単純化することであった。これは、符号化ビットストリームのエントロピー・デコードおよびパース（parsing）を適用することによってストリームを部分的に圧縮解除する技法をもって改善された。エントロピー・デコード後、透かし入れアルゴリズムは、係数および動きベクトルのようなシンタックス要素に対して直接作用した。ストリームが透かしデータを表すよう修正されたのち、エントロピー符号化が適用される。本開示において我々が提案するのは、エントロピー・デコード段階および再エンコード段階なしで直接、エントロピー・エンコードされたビットストリームに対して透かし入れを適用する実装である。

【００１４】

CABACストリームに透かし入れすることは、CABACストリーム中の符号化されたシンタックス要素を変えることを伴う。しかしながら、上述したように、CABACビットストリームの符号化されたシンタックス要素を、他のいかなるシンタックス要素の符号化も変えることなく修正することによって透かしを適用することが望ましい。これは、CABACストリーム中の各要素の値が後続の値の解釈に影響するので難しい。一般に、符号化されたシンタックス要素を変更すると、将来のすべての符号化された要素がデコーダによって誤解釈されるまたはデコード不能になる。

【００１５】

いくつかの応用は、エンコードされたビットストリームに対してなされる変更はビットストリームの長さを変えることはできないという追加的な制約をもつ。極端な場合には、符号化された要素が代替的な値で置き換えられるのは、代替的な値のビット長がもとの値のビット長と同じである場合にのみであることが要求される。

【００１６】

最初に、CABACエンコードされたH.264/AVCストリームにおけるビットのストリングが異なるビットのストリングで置き換えられることのできるシナリオであって、その修正が目標とされるシンタックス要素のみに影響し、他のいかなる符号化されたシンタックス要素の正しい解釈にも干渉しないような仕方でなされるようなものを議論する。そのようなシナリオでは、置き換え用のビットのストリングがもとのビットのストリングと同じ長さであることは要求されない。H.264/AVCビットストリーム中のシンタックス要素のすべてがCABACエンコードされているのではなく、これらの非CABAC要素を修正することが可能でありうる。しかしながら、本稿に記載される少なくとも一つの実装は、特にCABACでエンコードされているシンタックス要素に集中する。

【００１７】

第二に、置き換えビット・ストリングがもとのビット・ストリングと同じ長さであることを要求する場合に使うことのできる効率的なアプローチを記述し、示す。

【００１８】

第三に、ビットストリーム中に情報を埋め込むために、上記のようなCABACエンコードされたビットストリーム中のシンタックス要素をどのように修正するかを記述し、示す。さらに、この情報は、修正されたビットストリームを解析することによって、あるいはビットストリームをデコードしてピクセル領域画像のシーケンスにしてそれらの画像を解析することによって、のちに回復されることができる。

【００１９】

最後に、符号化されたビットストリームのための透かし埋め込みプロセスの効率を改善するために「バイパス・モード」がいかに活用できるかを記述し、示す。

【００２０】

さまざまな実装の理解を助けるために、CABAC、バイパス・モードおよび透かし入れについての簡単な議論が与えられる。これらの議論はしばしば個別的である。しかしながら、これらの議論は完全であることを意図したものではなく、さらに、これらの議論のあらゆる詳細が必ずしも本願において記載されるあらゆる実装に適用されなくてもよい。さらに、これらの議論は、当業者には既知である多くの内容を含んでいる。しかしながら、これらの議論は、たとえ個別的に事例を指摘しないとしても、変わることなく新規な内容、

あるいは内容の編成を含むであろう。これらの議論は本願の広さを制限することを意図したものではない。むしろ、これらの議論は、読者が実装を理解するのを助けるために、数多くの可能なコンテキストのうちの一つまたは複数の個別的なコンテキストを提供する。

【００２１】

〈CABACの議論〉

多くのビデオ圧縮方法の最終段階は、圧縮されたデータの可逆的エントロピー符号化である。H.264/AVCでは、CABACがエントロピー符号化のために普通に用いられている。CABACは、各シンタックス要素型について個別のコンテキストを維持し、これらのコンテキストを各符号化された要素をもって適応させることによって、改善された圧縮性能を達成する算術符号化方式である。これは、典型的な算術符号化方式のいくつかの拡張を表す。第一に、CABACは二値データを算術エンコードするよう設計されている。換言すれば、各シンボルは0または1である。本来的に二値ではないシンタックス要素値は、算術符号化に先立ってまず二値化される。これは、0の確率と1の確率があるだけであるということを意味する。二値化の個別的な方法は、その後の算術符号化の効果性を最適化するよう意図される。二値化された（binarized）各シンタックス要素はビン・ストリング（bin string）と呼ばれる。第二の拡張は、符号が適応的（adaptive）であるということである。ビン・ストリングの各ビットが処理される際、符号化を制御するデコード変数が変化することを許される。最後に、この適応はコンテキスト・ベース（context-based）であることが許される。各ビットに、いくつかの変数からなるコンテキストが関連付けられる。コンテキスト変数のいくつかはいくつかのコンテキストの間で共有され、いくつかは単一のコンテキストの専用である。一つのシンタックス要素の符号化は、関連付けられたコンテキストを適応させる。H.264/AVCは、エンコードおよびデコードの間に維持される460の別個のコンテキストを定義している。

【００２２】

上記で使った用語「デコード変数（Decoding Variables）」は二組の変数を指す。第一の組は「コンテキスト変数（Context Variables）」と呼ばれ、0または1になる確率を表す二つの変数を含む（これらのシンボル（symbol）のうちどちらが可能性が低い（least probable）LPSかの特定およびこのLPSがビットストリーム中で次に現れる確率を示す確率の表中への確率状態インデックスIによって表される）。各コンテキストは、一組のコンテキスト変数を維持するコンテキスト・インデックスに関連付けられる。第二の組は、「状態変数（State Variables）」と呼ばれ、内的なCABAC値を定義するための二つの変数からなる（範囲（range）の底Lおよび範囲の幅Rによって表される）。LおよびRの変数はすべてのシンタックス要素の間で共有され、一方、別個のLPSおよびIの値が各コンテキストについて維持される。用語「状態変数」「状態値」および「状態変数値」は本稿では交換可能に使われる。

【００２３】

二値化ののち、適切なデコード変数が取得され、ビン・ストリングを算術エンコードするために使用される。各ビットがエンコード（またはデコード）されたのち、対応するコンテキストはエンコード（またはデコード）されたビットおよび前のコンテキスト値に基づいて更新される。

【００２４】

特定のシンタックス要素について、符号化されるべき次のシンボルは、MPSであると期待される（最も可能性の高いほうのシンボル（the most probable symbol）は、LPSではない二値シンボルである）。これが起こる場合、LPSが次に現れる期待または確率におけるさらなる減少を反映するよう、それはインデックスIの変化を引き起こす。LPSが現れる場合、インデックスは、LPSの確率の上昇を反映するよう変化することになる。LPSの確率は50%を超えることはできない。50%を超えるということは、そのシンボルがもはや最も可能性が低いものではないということを意味するであろう。この場合、LPSは他方のシンボルに変わり、インデックスは再び50%より低い確率を表すことになる。

【００２５】

LおよびRの値は、すべての算術エンコード・プロセスと同じ仕方で更新される。どのシンボルがエンコードされるかに依存して、Lは現行の区間（interval）の下限であり、Rは現行の区間の幅である。

【００２６】

一般に、状態変数（具体的にはLおよびR）に対する同じ変化につながる二つのビン・ストリングはない。これは、デコーダにその後のすべての符号化された要素を誤解釈させることなく、算術エンコードされたビットストリームを修正することは不可能であることを含意するであろう。しかしながら、CABACは、LおよびRが固定長の二進数で表現されていることを要求する。これは、再規格化（renormalization）として知られるプロセスを通じて達成される。LおよびRの再規格化についての個別的な詳細は、たとえばH.264規格に見出すことができる。

【００２７】

この再規格化プロセスの一つの結果は、いくつかの異なるビン・ストリングがLおよびRに対する同じ変化を誘起することが可能になるということである。少なくとも一つの実装は、この事実を活用して、CABACデコーダにおけるもとのLおよびR状態変数値を維持しながら、異なる値をもって置き換えることのできるシンタックス要素値を特定する。

【００２８】

〈バイパス・モードの議論〉

二つのシンボルが等しい確率をもつような特殊なシンタックス要素の場合には、LPSおよび状態インデックスは符号化効率の面でいかなる利得を導入することもなく常時更新されることになる。インデックスによって表現される確率は本質的に50%であり、これは範囲Rはあるビットから次のビットにかけて同じ予測可能な変化を経験するであろうことを意味する。

【００２９】

符号化効率を損なうことなくエンコード／デコード・プロセスを高速化するため、シンボルが等しい確率をもつと期待されるシンタックス要素が識別され、バイパス・モードと呼ばれる特殊な仕方で扱われる。ここでは、コンテキストは使用されず、両方のシンボルの確率は0.5に固定される。結果は、範囲Rは、その値によらず、それぞれのデコードされたビンのあとの値の半分になるというものである。再規格化はRを二倍にし、よってRの最終的な固定長表現はバイパス・モード・デコードにおいて不変なままとなる。範囲の下限Lだけが新たなシンボルが符号化されるにつれて適応される。少なくとも一つの実装は、CABACデコーダにおいてもとの状態変数値を維持しながら異なる値で置き換えることのできるシンタックス要素値の検索を単純化するために、バイパス・モードを活用する。

【００３０】

H.264規格は、三つのシンタックス要素がバイパス・モードのCABACで符号化されなければならないことを規定している。これらは係数の符号、動きベクトル差分のサフィックスおよび係数データのサフィックスである。

【００３１】

〈透かし入れの議論〉

ここで透かし入れの一般的な分野およびこの分野内での、一つまたは複数の記載される実装が当てはまるより特定的な領域について簡単に論じておく。透かし入れの分野の当業者はここに呈示される情報の多くになじみがあるであろう。

【００３２】

本稿で使用される「デジタル透かし入れ（digital watermarking）」の句は、典型的には、芸術作品（典型的には画像、動画シーケンスまたはオーディオ・クリップ）を、何らかのペイロード・データに従って、次の条件が満たされるように修正する方法を指す：

１．その作品の修正されたバージョンがその作品のもとのバージョンと知覚的に区別不可能であり、かつ、

２．後刻、その作品の修正されたバージョンからペイロード・データを取り出せる。

【００３３】

「堅牢なデジタル透かし入れ（robust digital watermarking）」と称されるデジタル透かし入れ方法のサブクラスは、第三の条件を導入する：

３．その作品の修正されたバージョンのゆがめられたバージョンからペイロード・データが取り出せる。ここで、ゆがみは、その修正された作品に適用されたことがありうる一般的な信号処理および信号取り扱い（たとえば、圧縮、ノイズ削減フィルタ処理、色強調（color enhancement）など）によって導入されたものであってもよいし、あるいはゆがみはペイロード・データを取り出し不能にしようと試みる攻撃者によって意図的に導入されたものであってもよい。

【００３４】

堅牢なデジタル透かし入れには多くの応用がある。それには、これに限られるものではないが、以下のものが含まれる。

【００３５】

１．所有者識別：透かしペイロードが作品の所有者を特定する。

【００３６】

２．コピー・コントロール：透かしペイロードが作品に関連付けられた著作権を示す。閲覧、複製、記録、印刷、配布または他のいかなる動作のための装置もペイロード・データを取り出すことができ、動作を著作権によって許されているものに制限できる。

【００３７】

３．トランザクション追跡：透かしペイロードは作品のそのコピーが合法的に配布された受取人を特定する。これは、合法的な受取人がその作品をさらに配布する権利は受けないときに有用となりうる。許諾されていない作品が発見された場合、もとのコンテンツ所有者は、許諾されていないコピーからペイロードを取り出し、許諾されていない使用について責任のある受取人を特定することができる。

【００３８】

当業者は、堅牢なデジタル透かし入れの他の多くの応用があることを認識するであろう。

【００３９】

透かし入れは、「ベースバンド」の、あるいは圧縮された作品に対して実行されることができる。ベースバンドの画像または動画シーケンスは、たとえばピクセル領域表現である。ベースバンドのオーディオ作品は、たとえばオーディオ・サンプルである。いくつかの応用は圧縮された作品の透かし入れを要求する。この場合、出力も圧縮された作品である。圧縮の最後の段階は典型的にはエントロピー符号化であり、圧縮解除の最初のステップは典型的にはエントロピー復号である。エントロピー・エンコード／デコード・プロセスは典型的には可逆的〔無損失〕である。圧縮された作品に透かし入れするための一つのアプローチは、まずエントロピー・デコードを適用し、次いで圧縮解除を適用してベースバンド表現を得ることである。ベースバンド表現は透かし入れされ、結果として得られる透かし入れされた作品が圧縮され、エントロピー符号化される。このアプローチは時間がかかり、再圧縮のため知覚的な品質の劣化につながることがありうる。

【００４０】

知覚的な品質を改善し、必要とされる計算を軽減するため、もとの圧縮された作品からの動きベクトル、モード決定、品質因子およびその他の情報といった情報が圧縮解除の際に保存され、再圧縮の際に使用されることができる。これは、再圧縮はいかなる動き推定も実行する必要がない（よって計算および／または時間が節約される）ということを意味し、もとの量子化因子およびモード決定が改善された知覚的品質に帰結しうる。

【００４１】

「圧縮領域透かし入れ（compressed domain watermarking）」と呼ばれる透かし入れアルゴリズムのクラスが登場している。これらの方法は、先述した「完全デコード」ではなく透かし入れに先立つ「部分デコード」を実行することとして記述できる。ここで、圧縮された作品はまずエントロピー・デコードされて圧縮された作品のシンタックス要素（syntax element）を暴露する。シンタックス要素は、係数（たとえばブロックDCTまたは

ウェーブレット係数）、動きベクトル、ピクチャー型、予測モードおよび他の多くのシンタックス要素を含みうる。透かし入れアルゴリズムは次いで直接、それらのシンタックス要素のいくつかを修正する。最後に、修正されたシンタックス要素がエントロピー符号化されて修正された圧縮された作品を得る。

【００４２】

本願に記載される少なくとも一つの実装は、「エントロピー符号化ストリームの透かし入れ（entropy coded stream watermarking）」と呼びうる新しいクラスの透かし入れアルゴリズムに属する。これらの方法は、図１に示されるようにエントロピー符号化されたストリームを直接修正する。

【００４３】

図１を参照すると、もとの圧縮されたストリームは、ストリーム修正１０２によってペイロードに従って直接修正され、出力は結果として得られる透かし入れされた圧縮ストリーム１０３である。ペイロードはたとえば、プレーヤーのシリアル番号、プレーヤーのモデルまたはのちの識別のために使うことが望まれる本質的に他のいかなる情報であってもよい。ストリーム修正プロセスは、なされるべきすべての変更の位置およびなされるべき実際の変更について、前処理段１０４の間に生成できるメタデータによって知らされる。メタデータ１０５は、もとの圧縮されたストリームをデコード１０６し、透かしを生成１０８することによって生成される。メタデータはストリーム内で変更がなされるべき（すなわち、透かし生成のための）位置を特定し、種々のペイロード・シンボルに応答してストリームがどのように変化すべきかを指示する。図１は、メタデータが、圧縮された作品のデコードされたシンタックス要素の解析によって生成される特別な場合を示している。このような透かし入れ方法の一つの困難な側面は、メタデータの生成である。以下は、一つまたは複数の実装についてこれがどのようにしてできるかを説明する。

【００４４】

前の段落および図１は、前処理段１０４がありうることを示唆している。三つの重要な時がある状況を指摘しておくことが有用である。第一の時は、圧縮されたビットストリームが解析されて何らかのメタデータを生成するところ／時である。第二の時は、第一の時に生成されたメタデータの一部または全部が、ペイロードとして知られる特定のシンボルのシーケンスとともに使用されて、ビットストリームを修正するところ／時である。第三の時は、第一の時に生成されたメタデータの全部が使用されるか全く使用されないかして、修正されたビットストリームまたは該修正されたビットストリームを圧縮解除することによって得られる動画シーケンスまたは該修正されたビットストリームを圧縮解除することによって得られる動画シーケンスのゆがめられたバージョンを解析するところ／時である。第三の時におけるこの解析の目的は、ペイロードを取り出すことである。

【００４５】

コンテキストを理解するために、第一の時が作品の配布より先にあり、第二の時が複製の間にあり、ここで各コピーについて使われるペイロードがそのコピーを一意的に特定するものとし、第三の時は作品の不正なコピーがみつかった後にあり、その時点でペイロードが取り出されて配布されたコピーのうちのどれがその不正コピーのソースであったかを明らかにすると考える。この例は単に例示であり、これらの概念の応用に対するいかなる制限を示唆することも意図されていない。さらに、応用固有の要求があることもありえ、第一の時（前処理）と第二の時（埋め込み）が異なっていなければならないという技術的な必要性はない。

【００４６】

追加的な例として、先の図１を参照する。前処理段は、たとえば頒布用の映画のオーサリングの際に実行されてもよい。一つのシナリオでは、認容可能な置き換えが決定された、符号化された要素の位置を特定するためのプロセスが実行される。そのプロセスの結果（たとえばそのような符号化された要素の位置および認容可能な置き換え値）が、符号化された映画とともに含められるメタデータ内に記憶される。

【００４７】

ストリーム修正は、映画の再生の際に実行されてもよい。ここで、たとえば、プレーヤー（たとえば、ソフトウェア・プレーヤー、セットトップボックス・プレーヤーまたはDVDプレーヤー）のシリアル番号がペイロードとして使われる。ペイロードは、たとえば、特定された符号化された要素を置き換えるかどうかを決定するために使われる。たとえば、単一の認容可能な代替がメタデータ内に与えられていれば、ペイロード・シーケンス内の「0」は関連付けられた符号化シーケンスについての「置き換えなし」を示してもよく、「1」は「置き換え」を示してもよい。もう一つの例として、特定された位置がすべて置き換えられてもよく、ペイロードは（位置によって特定される各符号化されたシーケンスについて）二つの用意された置き換え値のどちらが使用されるべきかを示してもよい。たとえば、「0」は第一の置き換えを使うことを示してもよく、「1」は第二の置き換えを使うことを示してもよい。

【００４８】

ペイロードの取り出しは、ストリーム修正が起こった位置に無関係な位置で実行されてもよい。鍵は、修正されたストリームに基づくデータを含む文書へのアクセスがあることである。たとえば、修正されたストリームは電子的にコピーされて記録可能DVDに入れられてもよく、あるいは修正されたストリームは呈示されて次いで再記録され、エンコードされ、次いで記録可能DVDに入れられてもよい。この記録可能DVDが取得される場合、この記録可能DVDがペイロードを取り出すために解析されることができる。

【００４９】

〈AVC符号化ビットストリームにおけるCABACベースのデータ埋め込み〉

圧縮されたストリームに透かし入れする一つの方法は、以下の要件／条件に従って一つまたは複数のシンタックス要素の値を変えることである。

【００５０】

Ｒ１．修正された圧縮されたストリームは相変わらず有効なストリームである（特定の圧縮規格に準拠したままである）。

【００５１】

Ｒ２．修正された圧縮されたストリームを圧縮解除することによって得られる動画シーケンスは、もとの圧縮されたストリームを圧縮解除することによって得られる動画シーケンスから知覚的に識別不能である。

【００５２】

Ｒ３．ストリームへの修正は、修正された圧縮されたストリームを圧縮解除することによって得られる動画シーケンスにおける測定可能な変化に帰結する。

【００５３】

圧縮技術が算術符号エントロピー符号化技術を使うとき、要件Ｒ１は特に困難である。

【００５４】

まず、要件Ｒ１を考え、CABACエンコードされたH.264/AVCストリーム内のビットのストリングがどのようにして異なるビットのストリングで置き換えられ、修正された圧縮されたストリームが相変わらず有効なストリームであるようにできるかを論じる。これは、修正が目標とされるシンタックス要素のみに影響し、他のいかなる符号化されたシンタックス要素の正しい解釈にも干渉しないことを要求する。この実装では、置き換えビット・ストリングがもとのビット・ストリングと同じ長さであることは要求されない（この制約はのちに議論する）。説明の目的のため、以下の記述は「目標要素（target element）」と称される単一のシンタックス要素を修正することを扱う。しかしながら、当業者は、同じ概念が二つ以上の目標要素を含む目標要素の組に適用されることができることを認識するであろう。

【００５５】

目標要素は、二値化されて「もとのビン・ストリング」にされた「もとの値」をもつ。もとのビン・ストリングはCABAC符号化され、他のシンタックス要素とともにビットからなる「もとの符号化ブロック」を与える。CABAC符号化プロセスでは、もとのビン・ストリングは、シンタックスに関連付けられた状態変数およびコンテキストを、もとの値から

修正された値に修正する。

【００５６】

さまざまな記載される実装において、要件Ｒ１を満たすようにCABACエンコードされたビットストリームを修正するには二つの段階があるが、それらの段階はどの順に実行されてもよい。これらの段階を論じるに当たって、「段階１」「段階２」という言い方をするが、これらの段階番号はそのいかなる特定のまたは個別的な実行順序をも含意または要求することを意図したものではない。図２ａおよび２ｂは、要件Ｒ１を満たす段階を示すブロック図である。

【００５７】

CABACエンコードされたH.264/AVCビットストリームが与えられたとして、段階１は目標要素からもとの符号化ブロックへのマッピングを構築する（２０２）。一つの実際的な実装では、これはビットストリームをデコードして、どのビットストリームがどのシンタックス要素（および対応する状態変数値）を生成するかを追跡することによってなされる。第二の実際的な実装では、圧縮されたビットストリームにCABACデコーダが適用されてシンタックス要素を暴露する。これらのシンタックス要素は次いで、もとの圧縮されたビットストリームを取り出すために、装備のある（instrumented）CABACエンコーダによって処理される。ここでは取り出されたビットストリームには関心はない。むしろ、有用なのはCABACエンコーダの装備（instrumentation）である。装備のあるCABACエンコーダは、シンタクス要素と圧縮ビットストリーム内でのその最終的な位置との間の対応を追跡するために使うことができる。

【００５８】

段階２は、目標要素についての一つまたは複数の認容可能な代替値を探索する。認容可能な代替値を探索する一つの方法は、あらゆる可能な代替値を調べて、認容可能なものがあるとすればどれかを決定することである。可能な代替的なシンタックス値のそれぞれは、二値化されて代替的なビン・ストリングにされる。代替的なビン・ストリングはCABAC符号化され、他のシンタックス要素とともに代替的な符号化されたビットのブロックを与える。CABAC符号化プロセスでは、代替的なビン・ストリングは状態変数および影響されるコンテキストをそのもとの値から代替値に修正する。代替的な状態変数値および影響されるコンテキストのすべてが修正された状態変数値および影響されるコンテキストのすべてと同一である場合、可能な代替シンタックス値は「認容可能な代替値」と呼ばれる。換言すれば、変更は、状態変数および影響されるコンテキストが、変更がなかったとした場合に有したであろう値と同じ値を有する場合に、認容可能となる。

【００５９】

これら二つの段階２０２、２０４の二つの異なる順序付けが図２ａおよび２ｂに描かれており、段階２０４の詳細が図３に示されている。段階１（２０２）におけるマッピングおよび割り当ては、少なくとも一つの代替的なシンタックス値が段階２（２０４）で見出された目標要素についてのみ要求されることを注意しておく。両段階の図２ｂの第二の順序付けでは、段階１（２０２）のマッピングはそれらの同定された目標要素について実行される必要があるのみである。

【００６０】

図２ｃは、符号化ビットストリームを修正するための方法２５０のブロック図を示している。最初に、第一の部分および第二の部分をもつ符号化ビットストリームがアクセスされる（２５２）。置き換え値が第一の部分について決定され（２５４）、第一の部分は次いで決定された置き換え値で置き換えられる（２５６）。ひとたび置き換え値が第一の部分を置き換えたら、修正された符号化されたビットストリームが送られる（２５８）。「送る」とは、たとえば、電子的な送信またはディスクもしくはマスター形式で送ることでありうる。

【００６１】

ここでいう第二の部分は、次の隣接するシンタックス要素（および対応する状態変数値）であることができ、あるいは二つ以上のシンタックス要素および対応する状態変数値か

らなることのできるビットストリームの残りの部分であることができる。こうして、第一の部分についての置き換え値を決定するとき、その後のすべてのシンタックス要素およびそのそれぞれの状態変数値について配慮されることができる。第一の部分についての代替的なまたは決定された置き換え値が第二の部分における他のシンタックス要素についての状態変数値を変えたり変更したりしないとき、第一の置き換え値は、第二の部分の所望される結果的なデコードを提供することになる。

【００６２】

置き換え値を決定する動作２５４は、修正された第一の部分を決定するとして特徴付けされることもできる。議論している実装では、第一の部分の修正の前には、第二の部分はもともと、ある特定の結果を生じるようデコード可能である（デコードは部分的に、修正されていない第一の部分に依存する）。さらに、第一の部分の修正後には、第二の部分は同じ特定の結果を生じるようデコード可能である（デコードは部分的に、修正された第一の部分に依存する）。

【００６３】

方法２５０に似た別の実装は二つの動作を伴う。第一の動作は、少なくとも第一の部分および第二の部分を含むエンコードされたデータにアクセスすることである。第二の部分は、第二の部分のほか第一の部分にも基づく結果を生じるようデコード可能である。第二の動作は、第二の部分が第二の部分および修正された第一の部分に基づいて前記結果を生じるようデコード可能であるよう、修正された第一の部分を決定することである。

【００６４】

符号化された要素の位置および可能な置き換え値を特定するデータの全部または一部は記憶装置上に記憶されることができ、あるいは電子的に送られることができる。一つの実装は、DVD、ハードディスクまたは他の記憶装置といった装置である。該装置は、情報が記憶されたプロセッサ読み取り可能な媒体を含む。記憶された情報はエンコードされたデータのセットの第一の部分についての置き換え値を同定する。エンコードされたデータのセットはまた第二の部分をも含み、第二の部分のデコードは第二の部分および第一の部分に基づく。置き換え値は、（１）デコードが第一の部分を含むエンコードされたデータのセットに対して実行される場合、第二の部分のデコードが、特定のデコードされた結果を生じ、（２）デコードが第一の部分の代わりに当該置き換え値を含むエンコードされたデータのセットに対して実行される場合、第二の部分のデコードが、前記特定のデコードされた結果を生じるという性質をもつ。別の実装は、この装置のプロセッサ読み取り可能な媒体上に記憶されている情報を含むようフォーマットされた信号を提供する。

【００６５】

図２ｄは、符号化ビットストリームを修正する方法２６０のもう一つの実装のブロック図を示している。最初に、第一の部分および第二の部分をもつ符号化されたビットストリームがアクセスされる（２６２）。次いで、符号化ビットストリーム内の第一の部分の位置についての情報を与えるメタデータがアクセスされる（２６４）。次いで第一の部分についての置き換え値がアクセスされ（２６６）、第一の部分がアクセスされた置き換え値で置き換えられる（２６８）。ひとたび置き換え値が第一の部分を置き換えると、修正された符号化されたビットストリームが送られる（２７０）。「送る」とは、たとえば、電子的な送信またはディスクもしくはマスター形式で送ることでありうる。

【００６６】

方法２６０に似た別の実装は二つの動作を伴う。第一の動作は、少なくとも二つの部分を含む符号化されたデータにアクセスすることである。第二の部分は、第二の部分のほか第一の部分にも基づく結果を生じるようデコード可能である。第二の動作は、第二の部分のデコードが相変わらず前記結果を生じるような修正された符号化されたビットストリームを生成するよう、第一の部分を置き換え値で置き換えることである。

【００６７】

追加的な諸実装は、方法２６０のような諸方法の結果に向けられる。たとえば、一つの実装は、修正されたエンコードされたデータを含む、DVD、ハードディスクまたは他の記

憶装置といった装置であり、別の実装は、そのような修正されたエンコードされたデータを含むようフォーマットされた信号である。より具体的には、一つの実装は、修正された第一の部分と第二の部分とを含むエンコードされたデータが記憶されたプロセッサ読み取り可能な媒体を含む。修正された第一の部分は第一の部分の置き換えであり、ここで、（１）第二の部分は、第二の部分のほか修正された第一の部分にも基づく結果を生じるようデコード可能であり、（２）前記結果は、第二の部分のデコードが第二の部分および修正された第一の部分ではなく第二の部分および第一の部分に基づいて実行されたとした場合と同じである。別の実装は、この装置のプロセッサ読み取り可能な媒体上に記憶されている前記エンコードされたデータを含むようフォーマットされた信号である。

【００６８】

図３は、目標要素のすべての認容可能な代替値を見出す段階についての一つの実装を表す流れ図３００を示している。最初に、目標要素についてのあらゆる潜在的な代替（置き換え）値のリストが用意される（３０２）。リスト内に何らかの代替（置き換え）値が残されているかどうかについての判定がなされる（３０４）。リスト内に代替（置き換え）値があるとき、次の可能な代替（置き換え）値が取得され、シンタックス値がエンコードされる（３０６）。次いで、状態変数値およびコンテキストを含む代替（置き換え）デコード変数値（decoding variable values）が修正された代替（置き換え）デコード変数値と等しいかどうかについての判定がなされる（３０８）。等しいとき、代替（置き換え）値は認容可能な代替値のリストに保存され（３１２）、次いで試験済みの値がリストから除去される（３１４）。次いで、潜在的な代替（置き換え）値のリストが尽くされるまで、このプロセスはステップ３０４で再び開始される。

【００６９】

ここで第二の要件Ｒ２を考える。所与の目標要素について、そのもとの値を要件Ｒ１を満たす代替値のそれぞれで置き換えることの知覚上の影響を評価または予測する必要がある。この評価または予測が圧縮されたストリームのシンタックス要素のすべてに対する完全なアクセスをもち、根底にある画像および人間の視覚もしくは聴覚系（ビデオの場合は人間の視覚系に関心がある）の感度の、修正に対するマスキング効果（masking effect）を予測するためにこれらを使うことができることを想起されたい。そのような予測の詳細は知覚的モデリングの当業者にはよく知られている。知覚的なモデルは、圧縮文献および透かし入れ文献のいずれにおいても幅広く引用されている。

【００７０】

知覚可能性の圧縮領域での計算的予測の使用に加えて、ストリームを完全に圧縮解除して動画シーケンスにすることができるという贅沢を有してもよい。これは、たとえば、解析が前処理として実行される場合に当てはまりうる。この場合、計算的予測は実際のピクセル・データを調べることができる。これは知覚可能性のより正確な予測につながりうる。さらに、もとのストリームを圧縮解除して基準動画シーケンスを取得し、次いでシンタックス要素値を代替値で置き換え結果として得られるストリームを圧縮解除して第二の動画シーケンスを取得することによって、もとのシンタックス要素値を代替的な値で置き換えることの知覚可能性にアクセスすることができてもよい。前記二つの動画シーケンスの間の差の知覚可能性を評価するためには、多くのよく知られた技法が使うことができる。

【００７１】

最後に、主観的な評価に頼るという贅沢を有してもよい。人間の閲覧者が基準動画シーケンスと修正された動画シーケンスとの間の差の知覚可能性を評価することができる。試験者がその差を知覚することができない場合、その修正は要件Ｒ２を満たす。

【００７２】

これらは、要件Ｒ１を満たすあらゆる変更のうちどれが要件Ｒ２をも満たすかを決定するために使うことのできる方法のほんの一部である。

【００７３】

ここで、第三の要件Ｒ３を考える。目標シンタックス要素についての代替的なシンタックス値は、修正されたストリームがのちに圧縮解除されて動画シーケンスにされるときに

何らかの測定可能な変化を誘起しなければならない。誘起されることのできる測定可能な変化には、「直接」および「間接」の二つの種類がある。

【００７４】

「直接変化」では、シンタックス要素は動画シーケンスの測定可能なアーチファクトに直接対応する。たとえば、輝度ブロックDC係数の修正は、直接、対応する圧縮解除されたブロックの平均輝度の測定可能な変化につながる。他方、「間接変化」では、画像シーケンスにおいて測定されるアーチファクトは、ストリーム中の修正には間接的に関係しているのみである。たとえば、動きベクトルの修正は誤ったブロックが予測として使われる結果につながり、したがって対応する圧縮解除されたブロックにおける正しくないピクセル・データにつながることになる。どの動きベクトルが使われたかを決定するのは難しいことがありうるが、異なる動きベクトルの使用は他の測定可能なアーチファクトに影響することができる。動きベクトルは、より高いまたはより低い平均輝度をもつ再構成されたブロックを与えるために使われることができる。

【００７５】

要件Ｒ３の測定可能な変化は、ペイロードを取り出すために、位置を特定することにおいておよびどんな変更がなされたかを見ることにおいて支援する。これは海賊版のコピーが得られたときに特に有用である。なされた変更を「見る」ことによって、海賊版コピーのソースが取得できる。

【００７６】

ある実装では、どの動きベクトル・シンタックス要素が要件Ｒ１およびＲ２を満たす少なくとも二つの代替的なシンタックス値をもち、該代替的なシンタックス値の一つの使用がより高い平均輝度をもつ再構成されたブロックを与え、異なる代替的なシンタックス値の使用がより低い平均輝度をもつ再構成されたブロックを与えるようになっているかが判別される。これら二つの代替的な値のそれぞれの、CABAC符号化に対応する符号化されたビットのブロックは、CABACストリーム内でのビット位置とともに、メタデータとして埋め込み器に送られる。対応するペイロード・ビットの値に基づいて、埋め込み器はもとのビットのブロックを、一方または他方の代替的なビットのブロックに対応するビットのブロックで置き換える。たとえば、埋め込み器は、ペイロード・ビットが「0」の場合、対応するブロックの平均輝度を下げるシンタックス値に関連付けられたビットのブロックを選ぶまたは選択し、ペイロード・ビットが「1」の場合、対応するブロックの平均輝度を上げるシンタックス値に関連付けられたビットのブロックを選ぶ。明らかに、埋め込まれるべき各ペイロード・ビットについて、要件Ｒ１、Ｒ２およびＲ３のすべてを満たす少なくとも一つの目標要素が必要である。

【００７７】

メタデータは、圧縮解除された動画シーケンス中で輝度の変化が誘起されるフレーム番号およびブロック番号をも含むべきである。さらなる堅牢性のため、前処理段はそのブロックのもとの平均輝度をメタデータ中に記憶してもよい。この情報は、取り出し（たとえばデコード）の際に、修正されたブロックを識別し、その平均輝度をメタデータに含まれているもとの値と比較するために使われる。与えられている例では、もとの値より高い平均輝度は対応するペイロード・ビットが「1」であることを含意し、もとの値より低い平均輝度は対応するペイロード・ビットが「0」であることを含意する。

【００７８】

〈ビット長制約〉

代替的なビットの符号化されたブロックがもとのビットの符号化されたブロックと同じ長さをもつという追加された制約があるとき、その制約を単に要件Ｒ１に加えることができる。換言すれば、変化が認容可能になるのは、状態変数および影響されるコンテキストのすべてが同じままであり、かつ結果として得られる符号化されたブロックがもとと同じ長さである場合である。たとえば、再び図３を参照すると、判断ブロック３０８は、デコード変数値が同じままであるかどうかを判定するのみならず、結果として得られる符号化されたブロックがもとと同じ長さをもつかどうかをも判定するよう修正されることができ

る。もう一つの例としては、再び図３を参照して、ブロック３０２が、目標要素のあらゆる潜在的な代替値のうち、もとと同じ長さをもつ符号化されたビットのブロックに帰結するもののリストを提供するよう修正されることができる。

【００７９】

ビット長制約を加える前述したアプローチは一つの可能なアプローチである。しかしながら、応用によっては、代替的なアプローチが好適であることもある。ここで、認容可能な代替値の検索を制限することを意図した異なるアプローチを提示する。図５に記述されるこの実装では、第一の段階は、あらゆるシンタックス要素からビットストリーム中のその対応する符号化されたビットへのマッピングを構築することである（５０２）。第二の段階では、リストからの一つの目標要素が同定され、その要素を表すビットストリーム中の符号化されたビットのブロックが同定される（５０４）。

【００８０】

CABACはシンタックス要素を表すために諸ビットの諸部分を割り当てることができるので、異なるシンタックス要素を表す諸ビットの諸ブロックが重なることができる可能性がある。第三の段階は、符号化されたビットストリーム中での表現が修正されるべき要素の表現と重なる他の任意のシンタックス要素の同定である（５０６）。これらのシンタックス要素はここでは重なりシンタックス要素（Overlapping Syntax Elements）と称される。同じビット長をもつあらゆる可能な代替的ビット組み合わせのリストが決定される（５０８）。同じビット長をもつ同定された可能な代替的なビット組み合わせのどれが使用されるかを決定するためのその後の残りの段階（つまり、段階５１０～５２２）は、それぞれ、目標要素の潜在的な代替値のどれが使用されるかを決定するために記述された段階３０４～３１６と実質的に同一である。一つの違いは、ブロック５１４は、デコード変数値が不変のままであるかどうかを判定することのほか、その代替的なビット組み合わせが任意の重なりシンタックス要素のデコードされた値が不変のままであるという条件を満たすかどうかを判定することを含むということである。

【００８１】

符号化されたビットのブロックが長さnをもつ場合、そのブロックが表すことのできる2^n通りの異なる可能な値があることになる。次の二つの基準を満たすそのような何らかの値を探す：

１．重なりシンタックス要素のデコードされた値が不変のままである；かつ

２．符号化されたビットのブロックによって変更された状態変数およびコンテキスト変数のすべてが、修正されていない符号化されたビットのブロックと同じ値のままに残されている。

【００８２】

ここで、もとの符号化されたビットのブロックと同じサイズの代替的な符号化されたビットのブロックにつながる代替値のみに検索を限定した。

【００８３】

実装は重なりシンタックス要素およびビット長の制約の両方を扱う必要はない。むしろ、ある種の実装は重なりシンタックス要素のみ、あるいはビット長制約のみを扱って、他方の配慮は扱わなくてもよい。

【００８４】

さらに、さまざまな実装は、複数のシンタックス要素に対して一緒に作用してもよい（たとえば、データ・ストリーム中で連続しているシンタックス要素）。たとえば、ある実装は、複数の連続しているシンタックス要素（重なりまたは非重なりシンタックス要素）を単一のグループとして扱い、そのグループについての置き換えを、そのグループの末尾でデコード変数値が不変のままとなるよう決定する（だがグループ内のシンタックス要素どうしの間では必ずしもそうではない）。一つのそのような実装は、二つの連続するシンタックス要素を目標要素として考え、第二のシンタックス要素が重なりシンタックス要素であり、両方のシンタックス要素が修正されることを許容する。

【００８５】

〈情報埋め込み〉
上記の議論は、目標シンタックス要素を解析し、符号化ビットストリーム中のビットのブロックを代替的なビットのブロックで置き換えることによって置換できる認容可能な代替値をもしあればすべて同定するための方法を記述する。H.264/AVC符号化シーケンス中のあらゆるシンタックス要素を調べることによって、少なくとも一つの認容可能な代替値が存在するシンタックス要素のリストを構築できる。これは「変更可能なシンタックス要素」のリストである。変更できる各シンタックス要素について、認容可能な代替値のリストを構築できる。

【００８６】
これがビットストリーム中に情報を埋め込むためにどのように使えるかの例が提示された。情報埋め込みの目標は、あらゆる変更可能なシンタックス要素のリストにあるシンタックス要素のサブセットをペイロードに従って修正し、それらの値をそれらのもとの値からリストされた認容可能な代替値の一つに変えることである。一つの実際的な実装では、これは図６に示される三つの段階において達成される。

【００８７】
図６を参照すると、プロセス６００がある。プロセス６００の第一の段階では、変更可能なシンタックス要素のサブセットが選択される（６０２）。このサブセット選択は、変更がデコードされた画像に対してもつであろう知覚的な影響の推定に基づくことができる。サブセット選択は、信号処理または不正工作によって画像が修正された後のデコードされた画像における変化を検出することの困難さの推定に基づくことができる。サブセット選択は、たとえば、変更されたブロック内の連続するビットの数を制限するまたは任意の二つの変更されたブロックの間になければならない不変のビットの最少数を制限する実装要件によって影響されることができる。好ましい実装では、この第一の段階は前処理の際に実行される。

【００８８】
第二の段階では、メタデータがのちの使用のために保存される（６０４）。このメタデータは「埋め込みデータ（embedding data）」および「検出データ（detection data）」からなる。埋め込みデータは、もとの符号化されたビットのブロックが見出せる、圧縮されたビットストリーム中の位置と、ビットからなる二つのブロックである。二つのブロックの一方は対応するペイロード・ビットが「0」の場合に置き換えブロックとして使われるべきものであり、他方は対応するペイロード・ビットが「1」の場合に置き換えブロックとして使われるべきものである。ある実装では、これらのブロックの一方はもとのビットのブロックそのものであり、他方は代替的なビットのブロックである。この場合、変更可能なシンタックス要素のリストは、認容可能な代替値が一つだけあるシンタックス要素を含むことができる。別の実装では、二つのブロックの両方は、異なるシンタックス値に対応する代替的なビットのブロックである。この場合、変更可能なシンタックス要素のリストは、少なくとも二つの認容可能な代替があるシンタックス要素のみを含むことができる。

【００８９】
検出データは、埋め込みのために使用される変更可能な要素のセットおよびそれらの要素のもとの値を含む。また、圧縮解除された動画シーケンス中でビットストリーム修正が検出可能となる位置をも含む。これはフレーム番号およびブロック番号またはマクロブロック番号またはピクセル位置または取り出しのために必要とされる他の任意の位置指定子として指定されうる。検出データはまた、変更がなされなかったとした場合に測定されたであろう検出指標の値をも含むことができる。つまり、これは検出指標のもとの値を含むことができる。先の例では、検出指標は特定のブロックの平均輝度であり、検出データはそのブロックのもとの平均輝度を含むことができる。ある実装では、第二の段階は前処理の際に実行される。

【００９０】
輝度またはコントラストにおけるグローバルまたはローカルな変化に対する透かし入れ

技法の堅牢性を高めるために、検出データは、透かし入れプロセスによって修正されると予期されないブロックについても保存されてもよい。たとえば、検出データは、透かしによって変更されないブロックのもとの平均輝度を含んでもよい。そうすれば、検出器はこれらのもとの平均輝度を、画像全体または画像の基準領域における少なくとも一部が輝度の変更を受けたかどうかを判定するために基準として使うことができる。基準ブロック内の測定された輝度が検出データに記録されている輝度に一致しない場合、ペイロードの取り出しに先立って補償が行われてもよい。

【００９１】

第三の段階では、変更可能な要素に対応するもとの符号化されたビットのブロックが、埋め込みデータおよびペイロードを使って異なるビットのブロックで置き換えられる（６０６）。二つの場合を考える。第一の場合では、表現されるべきペイロード・ビットがあるシンボル値、たとえば「1」をもつ場合、置き換え用ビットのブロックは、認容可能な代替値の一つに対応する符号化されたビットのブロックである。表現されるべきペイロード・ビットが他方のシンボル値、たとえば「0」をもつ場合、置き換え用ビットのブロックはもとの符号化されたビットのブロックである。第二の場合では、サブセット選択は、少なくとも二つの認容可能な代替値がある要素に制約される。この場合、置き換え用の符号化されたビットのブロックは、表現されるべきビットが「0」である場合には認容可能な代替値の一つに対応する符号化されたビットのブロックであり、表現されるべきビットが「1」である場合には異なる認容可能な代替値に対応する符号化されたビットのブロックである。二つの代替値は検出データ内にそれが表すラベル「0」または「1」とともに記憶されている。好ましい実施形態では、この第三の段階は埋め込みの際に実行される。

【００９２】

ペイロードは検出データの支援でH.264/AVCビットストリームから取り出されることができる。検出データは、隠されたペイロード情報を表す特定のシンタックス要素およびそれらの要素のもとの値を示す。埋め込みプロセスが「0」を表すのにもとの値を使い、「1」を表すのに代替値を使う場合、検出器は、ビットストリーム中の値を検出データに記憶されているもとの値と比較することができる。両者が一致すれば、検出器は「0」のビットを報告し、両者が一致しなければ、検出器は「1」のビットを報告する。埋め込みプロセスがもとの値を二つの代替的な値の一方で置き換えた場合には、それらの二つの代替的な値およびそれらの対応するラベルが検出データから取り出される。すると検出器はビットストリーム中の値をそれら二つの代替のそれぞれと比較し、対応するラベルを報告する。

【００９３】

隠されたデータは、デコードされたピクセル領域画像から取り出されることもできる。このプロセスは、サブセット選択プロセスにおいて使われる特定の指標に固有である。一つの実際的な実装では、その指標はピクセルのブロックの平均輝度である。修正されたブロックそれぞれのもとの平均輝度が検出データから取り出される。デコーダがデコードされた画像シーケンス中の指定されたピクセルのブロックの平均輝度を計算し、その値を検出データに記憶されているもとの値と比較する。計算された値がもとの値より大きければ、検出器は「1」のビットを報告する。計算された値がもとの値より小さければ、検出器は「0」のビットを報告する。

【００９４】

この取り出し方法は、デコードされた画像がデコード後に、しかし検出より前に輝度の一様な変更によって修正される場合には問題がある。これに対処するため、いくつかの基準ブロックの平均輝度が検出データ内に記憶されることができる。これらは、埋め込みプロセスによって変更されないブロックである。検出器は取り出しの直前に基準ブロックの平均輝度を計算し、隠されたデータに関係のない任意の輝度変化を発見する。これらの輝度変化は、もとの値をしかるべく調節することによって補償されることができる。

【００９５】

情報埋め込みが多様な応用を含みうることは明らかなはずである。一つのそのような応

用は、データ・ストリームにペイロードが埋め込まれる透かし入れである。他の応用は、通信されることが望まれる実質的に任意の情報を埋め込むことを含む。さらに、いくつかの情報埋め込み応用は異なる要件を使用してもよい。たとえば、ユーザーによって知覚可能な変化につながる変更をデータ・ストリームに対してなすことが望ましいこともありうる。

【００９６】

〈バイパス・モードの活用〉

H.264/AVCはバイパス・モードにおいて符号化される三つのシンタックス要素を規定する。これらのシンタックス要素については、エンコードの間、一つの変数のみが修正される。諸ビットが符号化される際、範囲の下限を表すL値のみが変化する。このように、もとと同じデコード変数値を与える代替的なシンタックス値を見出す可能性は、バイパス・モード・シンタックス要素ではより高い。ある実際的な実装では、動きベクトル差分シンタックス要素のサフィックスのみが考えられる。これらのそれぞれについて、（１）符号化されたビットのブロックにおいて同数のビットを与え、（２）他のいかなるシンタックス要素の値にも影響せず、（３）その対応するビン・ストリングが処理されたのちCABACエンコーダ中で同じL値を与えるような代替値を探す。これらの代替値から、上記のように（たとえば図６参照）埋め込みのために使われるべきサブセットを選択する。バイパス・モードの固定長置換のための諸段階は、図４に描かれる方法４００によって示される。ここで段階４０２～４１６のプロセスはそれぞれ図３の方法に示された段階３０２～３１６において前記したのと同一である。ただし、段階４０８は代替的なデコード変数値対修正されたデコード変数値の判定（つまり、段階３０８）ではなく、代替的なLがCABAC符号化についての修正されたLに等しいということの判定である。

【００９７】

明らかなはずだが、修正されていないビットストリームはデコードされて、目標シンタックス要素および該目標シンタックス要素に続く一つまたは複数のシンタックス要素についての値を含め、もとのシンタックス値を生成することができる。先述したように、前記目標シンタックス要素に続く前記一つまたは複数のシンタックス要素の符号化は、少なくとも部分的に、前記目標シンタックス要素の値に依存する。前記目標シンタックス要素の値が修正されたのち、結果的な修正されたビットストリームが生成される。この結果的な修正されたビットストリームは、修正されたシンタックス値と、今や修正された目標シンタックス要素に続く前記一つまたは複数のシンタックス要素の値に対応するものである。

【００９８】

図７を参照すると、たとえばデコーダ、DVDプレーヤーまたは修正されたビットストリームにアクセスしてこれをデコードする他の装置によって実行されうるプロセス７００が示されている。修正されたビットストリームのデコードは、目標シンタックス要素の修正された値を生成し、目標シンタックス要素に続く前記一つまたは複数のシンタックス要素のもとの値をも生成する。

【００９９】

より特定的には、プロセス７００を参照すると、修正されたデータのセットがアクセスされる（７００）。修正されたデータのセットは、修正された部分と、この実装では修正されていない、別の部分とを含む。上に示したように、修正されていない部分は、修正されていない部分および修正された部分に基づく結果を生じるようデコードされることができる。

【０１００】

プロセス７００は、修正された部分（７２０）および修正されていない部分（７３０）をデコードすることを含む。修正されていない部分のデコードは、修正された部分が修正されなかったとした場合と同じ結果を生じる。

【０１０１】

図８を参照すると、ペイロード情報を取り出す、あるいはより一般に電子透かしを検出するためのプロセス８００が示されている。プロセス８００は可能な透かしがあるかどう

かを検査する位置を判別することを含む（８１０）。透かしは一般にその後の検出を許容するよう修正または挿入されたデータをいうものであるが、必ずしも透かしが挿入されている必要はないことを注意しておく。プロセス８００は、判別された位置からのデータにアクセスし（８２０）、これを解析する（８３０）ことを含む。プロセス８００はさらに、前記解析に基づいて何らかの透かしが存在していれば透かし情報を同定することを含む（８４０）。同定された透かし情報はたとえば、ビットまたは他の情報単位であってもよい。プロセス８００は、ペイロードをなす一つまたは複数の追加的なビットを同定するために一つまたは複数の他の位置について繰り返されてもよい。こうして、ペイロード全体がプロセス８００を使って取り出されうる。

【０１０２】

プロセス８００の実装は、ピクセル領域のデータを解析することのほか修正されたビットストリームを解析することも含む。たとえば、ビットストリームは、前記した実装の一つに従って、標的要素に関連付けられた諸ビットを修正することによって修正されてもよい。そのようなビットストリームにおいて、透かしは、適切なビット位置にアクセスし、それらの位置において透かし（修正）について試験することによって検出されうる。メタデータ内に見出されるようなビット位置および修正された値（または修正された可能性のある値）も検出データに含められてもよい。

【０１０３】

代替的にまたは追加的に、そのような修正されたビットストリームはデコードされ、再エンコードを含む他の処理にかけられてもよい。再エンコードの場合、そのような再エンコードされたビットストリームは、もとの修正されたビットストリームと同じ修正された目標要素を同じ位置に含むとは期待されない。しかしながら、再エンコードされたビットストリームはデコードされて、もとの修正の効果が相変わらず典型的には存在するピクセル領域のデジタル表現を生成しうる。こうして、もとの修正がその影響をもったフレームおよびブロック番号を同定する検出データが有用となる。たとえば、目標要素はもともとある特定のブロックについての動きベクトルであってもよく、その動きベクトルを修正することの影響はその特定のブロックの平均輝度を上げることであってもよい。平均輝度の上昇は典型的には、もとの修正されたビットストリームの処理にもかかわらず保持される。しかしながら、動きベクトルおよび該動きベクトルに対するもとの変化は、典型的には厳密に同じではない。

【０１０４】

図９を参照すると、プロセス９００は、ピクセル領域データから透かし情報が検出される実装を記述している。プロセス９００は、可能な透かしについてフレーム識別子およびブロック識別子を判別し（９１０）、判別されたフレーム識別子およびブロック識別子に対応するピクセル領域データにアクセスする（９２０）ことを含む。プロセス９００は、アクセスされたデータについてのある特徴の値を決定し（９３０）、決定された値をベンチマークと比較する（９４０）ことを含む。次いでプロセス９００は、その比較に基づいて、もしあれば透かし情報を同定する（９５０）ことを含む。

【０１０５】

プロセス８００またはプロセス９００のさまざまな実装は、多様なデータの一つまたは複数を含む記憶されている検出データを使う。たとえば、検出データは、可能性のある透かしについて検査するための位置を特定する位置情報を含んでいてもよい。位置情報はたとえば、フレーム識別子および／またはブロック識別子を含んでいてもよい。

【０１０６】

検出データはたとえば、ある特徴についてのベンチマーク値を含んでいてもよい。ベンチマークはたとえば、前記位置が修正される前の前記特徴の値であってもよい。ベンチマークはたとえば、アクセスされた位置（８２０）についての前記特徴の実際の値（新しいベンチマーク）と比較されて、変化の指標を与えてもよい。たとえば、比較は、その値（新しいベンチマーク）が修正の結果として増大したか減少したかを示してもよい。

【０１０７】

検出データは例えば、ある異なる（透かし入れされていない）位置からのある特徴についての基本値を含んでいてもよい。そのような基本値はたとえば、透かし入れされていない位置の平均輝度であってもよい。基本値は、先に説明したように、透かし入れされた位置以上のものに影響したデータ変更があったかどうかを判定するために使用されうる。ある種の状況では、グローバルな変更（たとえばフレーム全体への変更）があったことがありうる。たとえば、フレーム全体の平均輝度が変更されたことがありうる。その場合、基本値は、透かし入れされていない位置における前記特徴の新しい値（新しい基本値）と比較されることができる。その差がある閾値を超えていれば、グローバルな変更があったと判定されうる。代替的に、その差が単にグローバルな差と解釈されてよく、その差が、そのグローバルな変化を取り入れるために（可能性のある）透かし入れされた位置について判別された値（新しいベンチマーク）に加えられてもよい。この代替では、差は前記値（新しいベンチマーク）ではなく前記ベンチマークに（代替的に）加えられてもよいことを注意しておく。さらに、ある種の実装では、基本値と新しい基本値との間の比が決定されてもよく、ベンチマークまたは新しいベンチマークのいずれかがこの比を乗算されてもよい。

【０１０８】

明らかに、プロセス８００は、処理装置を含む多様な装置によって実行されうる。そのような装置は、たとえばアクセスされるデータ、ベンチマーク値、基本値および可能性のある透かし情報の位置（単数または複数）といったデータを記憶するための記憶装置をも含みうる。さらに、プロセス８００は、命令のセットとして具現されてもよい。

【０１０９】

明らかであろうが、シンタックス要素は、さまざまな異なる型の情報の一つまたは複数を表しうる。たとえば、シンタックス要素は動きベクトル、残差DCT（離散コサイン変換）係数、マクロブロック型、量子化パラメータまたはフラグでありうる。したがって、シンタックス要素はみな同じ長さをもつ必要はなく、さまざまな異なる長さをもつことができる。本願に開示されているさまざまな実装は、シンタックス要素によってどんな型の情報が表されるかに関わりなく、またシンタックス要素のサイズ／長さに関わりなく、いかなる型のシンタックス要素に対して作用してもよい。

【０１１０】

さらに、先に示したように、さまざまな実装は、複数のシンタックス要素に対して一緒に（たとえばデータ・ストリーム中で連続している複数のシンタックス要素に）作用してもよい。たとえば、ある実装は複数の連続するシンタックス要素（重なりまたは非重なり）を単一のグループとして扱い、そのグループの末尾のデコード変数値が不変なままとなるよう（だがそのグループ内のシンタックス要素間では必ずしもそうではない）、そのグループについての置き換えを決定する。

【０１１１】

本稿に記載された実装は、たとえば、方法またはプロセス、装置またはソフトウェア・プログラムにおいて実装されうる。たとえ単一の形の実装のコンテキストでしか議論されていなくても（たとえば、方法としてのみ議論されていても）、議論された諸特徴の実装は他の形（たとえば装置またはプログラム）でも実装されうる。装置はたとえば、適切なハードウェア、ソフトウェアおよびファームウェアにおいて実装されてもよい。諸方法はたとえば、プロセッサなどといった装置において実装されてもよい。プロセッサは処理装置一般を指し、たとえばコンピュータ、マイクロプロセッサ、集積回路またはプログラム可能な論理デバイスを含む。処理装置はまた、たとえばコンピュータ、携帯電話、ポータブル／パーソナル・デジタル・アシスタント（「PDA」）および他の装置といったエンドユーザー間の情報の通信を容易にする通信装置をも含む。

【０１１２】

本稿に記載されたさまざまなプロセスおよび特徴の実装は、多様な異なる設備または応用において、特にたとえばデータ・エンコードおよびデコードに関連する設備または応用において具現されうる。設備の例は、ビデオ符号化器、ビデオ復号器、ビデオ・コーデッ

ク、ウェブ・サーバー、セットトップボックス、ラップトップ、パーソナル・コンピュータ、携帯電話、PDAおよび他の通信デバイスを含む。明らかであるはずだが、該設備は、移動型であってもよく、移動式乗物に設置されてもよい。

【０１１３】

さらに、諸方法は、プロセッサによって実行される命令によって実装されてもよく、そのような命令は、たとえば集積回路、ソフトウェア担体または他の記憶デバイスといったプロセッサ読み取り可能な媒体上に記憶されてもよい。他の記憶デバイスはたとえば、ハードディスク、コンパクト・ディスケット、ランダム・アクセス・メモリ（「RAM」）または読み出し専用メモリ（「ROM」）といったものである。命令は、プロセッサ読み取り可能な媒体上に具体的に実現されたアプリケーション・プログラムをなしていてもよい。命令はたとえば、ハードウェア、ファームウェア、ソフトウェアまたは組み合わせであってもよい。命令はたとえば、オペレーティング・システム、別個のアプリケーションまたは両者の組み合わせにおいて見出されてもよい。したがって、プロセッサは、プロセスを実行するよう構成された装置およびプロセスを実行するための命令を有するコンピュータ読み取り可能な媒体を含む装置の両方として特徴付けされることができる。

【０１１４】

当業者には明白なはずだが、実装は、たとえば記憶または伝送されうる情報を担持するようフォーマットされた信号を生成してもよい。情報はたとえば、方法を実行するための命令、あるいは記載された実装の一つによって生成されるデータを含んでもよい。そのような信号は、たとえば、電磁波として（たとえばスペクトルの電波周波数部分を使って）あるいはベースバンド信号としてフォーマットされてもよい。フォーマットは、たとえば、データ・ストリームをエンコードし、エンコードされたデータ・ストリームで搬送波を変調することを含みうる。信号が搬送する情報はたとえば、アナログまたはデジタル情報でありうる。信号は、知られているように、多様な異なる有線または無線のリンクを通じて伝送されうる。

【０１１５】

いくつかの実装が記載されたが、さまざまな修正がなしうることは理解されるであろう。たとえば、種々の実装の要素が組み合わされ、補間され、修正され、あるいは除去されて他の実装を生じてもよい。さらに、当業者は、他の構造およびプロセスを開示されたものの代用としてもよく、結果として得られる実装が、開示される実装と少なくとも実質的に同じ機能を少なくとも同じ仕方で実行して少なくとも実質的に同じ結果を達成することを理解するであろう。したがって、これらおよびその他の実装は、本願によって考えられており、付属の請求項の範囲内である。

いくつかの態様を記載しておく。

〔態様１〕

少なくとも第一の部分および第二の部分を含むエンコードされたデータにアクセスする段階であって、ここで、前記第二の部分は、前記第二の部分のほか前記第一の部分にも基づく結果を生成するようデコードされることができる、段階と；

修正された第一の部分を決定する段階であって、前記第二の部分が、前記第二の部分および前記修正された第一の部分に基づいて前記結果を生成するようデコードされることができるよう決定する、段階とを有する、

方法。

〔態様２〕

前記第一の部分を前記修正された第一の部分で置き換える段階をさらに有する、態様１記載の方法。

〔態様３〕

前記修正された第一の部分が、前記第一の部分のビット長と等しいビット長をもつ値を有する、態様１記載の方法。

〔態様４〕

前記決定する段階が、前記第一の部分についての状態値と同じCABAC状態値を生成する

修正された第一の部分を決定することを含む、態様１記載の方法。
〔態様５〕
前記第一の部分のデコードが第一の結果を生成し、前記修正された第一の部分のデコードが、前記第一の結果とは異なる修正された第一の結果を生成する、態様１記載の方法。
〔態様６〕
前記修正された第一の部分を決定する段階が、前記第一の結果とは（１）見る人にとっては知覚不能であり、かつ（２）処理装置によって検出可能である違いをもつ修正された第一の結果を与えることに基づく、態様１記載の方法。
〔態様７〕
前記修正された第一の部分が、前記第一の部分を前記修正された第一の部分で置き換えることによってエンコードされたデータに透かし入れするために好適である、態様６記載の方法。
〔態様８〕
前記結果が、前記第一の部分のエンコードから生成される状態に基づくことによって前記第一の部分に基づく、態様１記載の方法。
〔態様９〕
前記第一の部分が、CABAC符号化されたビットストリームにおけるシンタックス要素を含む、態様１記載の方法。
〔態様１０〕
態様９記載の方法であって、前記決定する段階がさらに：
前記シンタックス要素についてあらゆる可能な代替的な置き換え値を決定する段階と；
可能な代替的な置き換え値それぞれについて、その置き換え値が前記シンタックス要素について同じ状態変数値を生成するかどうかを判定する段階と；
前記シンタックス要素についての同じ状態変数値を生成する代替的な置き換え値のリストを保存する段階とを有する、
方法。
〔態様１１〕
態様１０記載の方法であって、前記代替的な置き換え値を、該置き換え値が同じ状態変数を生成するかどうかを判定する前にエンコードする段階をさらに含む、態様１０記載の方法。
〔態様１２〕
前記第一の部分が符号化されたシンタックス要素を含み、前記修正された第一の部分が前記第一の部分のビット長と同じビット長をもつ値を含み、前記修正された第一の部分がCABACエンコーダにおいて、前記第一の部分によって生成されたであろうL値と等しいL値を生成する、態様１記載の方法。
〔態様１３〕
前記第二の部分が、それぞれ対応する状態変数値をもつ二つ以上の符号化されたシンタックス要素を含み、前記修正された第一の部分が、その後の第二の部分のすべてのシンタクス要素のすべてのデコードについて前記状態変数値を維持する、
〔態様１４〕
前記修正された第一の部分が前記データのセット内の他のいかなる符号化されたシンタックス要素の値にも影響しない、態様１３記載の方法。
〔態様１５〕
態様１記載の方法であって：
前記エンコードされたデータがエントロピー・エンコードされたデータであり、
前記第一の部分がシンタックス要素を含み、
前記修正された第一の部分が前記シンタックス要素の修正を含む、
方法。
〔態様１６〕
少なくとも二つの部分を含み、第二の部分は該第二の部分のほか第一の部分にも基づく

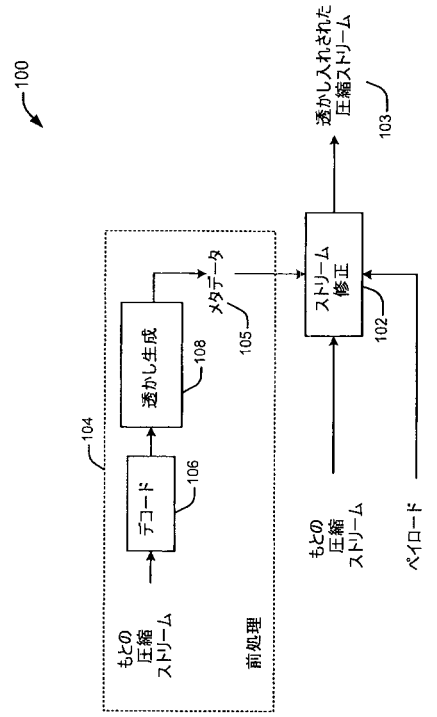
結果を生じるようデコードされることができる、エンコードされたデータにアクセスするアクセス・ユニットと；
修正された第一の部分を、前記第二の部分が該第二の部分および前記修正された第一の部分に基づいて前記結果を生成するようデコードされることができるよう決定する修正ユニットとを有する、
装置。
〔態様１７〕
少なくとも二つの部分を含み、第二の部分は該第二の部分のほか第一の部分にも基づく結果を生じるようデコードされることができる、エンコードされたデータにアクセスする手段と；
修正された第一の部分を、前記第二の部分が該第二の部分および前記修正された第一の部分に基づいて前記結果を生成するようデコードされることができるよう決定する手段とを有する、
装置。
〔態様１８〕
プロセッサ読み取り可能な媒体を有する装置であって：
少なくとも二つの部分を含み、第二の部分は該第二の部分のほか第一の部分にも基づく結果を生じるようデコードされることができる、エンコードされたデータにアクセスする段階と；
修正された第一の部分を、前記第二の部分が該第二の部分および前記修正された第一の部分に基づいて前記結果を生成するようデコードされることができるよう決定する段階とを少なくとも実行するための、
前記プロセッサ読み取り可能な媒体上に記憶された命令を含む、装置。
〔態様１９〕
プロセッサ読み取り可能な媒体を有する装置であって、エンコードされたデータのセットの第一の部分についての置き換え値を指定する前記プロセッサ読み取り可能な媒体上に記憶された情報を含み、前記エンコードされたデータのセットは第二の部分も含み、前記第二の部分のデコードは前記第二の部分および前記第一の部分に基づき、前記置き換え値は：
前記第二の部分のデコードが、デコードが前記第一の部分を含む前記エンコードされたデータのセットに対して実行されると、ある特定のデコードされた結果を生じ；
前記第二の部分のデコードが、デコードが前記第一の部分の代わりに前記置き換え値を含む前記エンコードされたデータのセットに対して実行されると、前記特定のデコードされた結果を生じる、
という性質をもつ、装置。
〔態様２０〕
エンコードされたデータのセットの第一の部分についての置き換え値を指定する情報を含むようフォーマットされた信号であって、前記エンコードされたデータのセットは第二の部分も含み、前記第二の部分のデコードは前記第二の部分および前記第一の部分に基づき、前記置き換え値は：
前記第二の部分のデコードが、デコードが前記第一の部分を含む前記エンコードされたデータのセットに対して実行されると、ある特定のデコードされた結果を生じ；
前記第二の部分のデコードが、デコードが前記第一の部分の代わりに前記置き換え値を含む前記エンコードされたデータのセットに対して実行されると、前記特定のデコードされた結果を生じる、
という性質をもつ、装置。
〔態様２１〕
前記信号がデジタル情報を表す、態様２０記載の信号。
〔態様２２〕
前記信号が変調された電磁波である、態様２０記載の信号。

〔態様２３〕
少なくとも第一の部分および第二の部分を含む符号化されたビットストリームにアクセスする段階であって、前記第二の部分は、前記第二の部分のほか前記第一の部分にも基づく結果を生成するようデコードされることができる、段階と；
前記第一の部分を置き換え値で置き換えて修正された符号化ビットストリームを生成する段階であって、修正された符号化ビットストリームについても前記第二の部分のデコードはやはり前記の結果を生成する、段階とを有する
方法。
〔態様２４〕
前記第一の部分の位置の指標にアクセスする段階と；
前記第一の部分についての置き換え値にアクセスする段階とをさらに有する、態様２３記載の方法であって、
前記置き換え値は、前記第一の部分を該置き換え値で置き換えた結果生じる修正された符号化ビットストリームが、前記第二の部分のデコードがやはり前記の結果を生成するようなものである、方法。
〔態様２５〕
ある情報にアクセスすることをさらに含む、態様２３記載の方法であって、前記第一の部分を置き換える段階が、前記アクセスされた情報に基づいて前記第一の部分を置き換えることを含む、方法。
〔態様２６〕
前記第一の部分を置き換える段階が前記符号化されたビットストリームへの透かし入れの一部であり、前記アクセスされた情報がペイロード情報を含む、態様２５記載の方法。
〔態様２７〕
前記ペイロード情報の値が前記第一の部分が置き換えられるべきか否かを指定する、態様２６記載の方法。
〔態様２８〕
前記ペイロード情報の値が、前記置き換え値を含む複数の潜在的な置き換え値のうちのどれが前記第一の部分を置き換えるべきかを指定する、態様２６記載の方法。
〔態様２９〕
前記符号化されたビットストリームにアクセスする段階が、エントロピー符号でエンコードされたビットストリームにアクセスすることを含む、態様２３記載の方法。
〔態様３０〕
前記エントロピー符号が算術符号である、態様２９記載の方法。
〔態様３１〕
前記算術符号がCABACである、態様３０記載の方法。
〔態様３２〕
前記置き換え値および前記第一の部分が同じCABAC状態値を生成する、態様３１記載の方法。
〔態様３３〕
前記のアクセスされた符号化されたビットストリームがCABAC符号化されたビットストリームであり、前記第一の部分が前記CABAC符号化されたビットストリーム中の少なくとも一つの符号化されたシンタックス要素を含む、態様３１記載の方法。
〔態様３４〕
前記アクセスされる指標がメタデータを含む、態様２４記載の方法。
〔態様３５〕
前記第二の部分が二つ以上のシンタックス要素を含み、前記二つ以上のシンタックス要素のそれぞれが、前記修正された符号化ビットストリームのデコードの際に生成される対応する状態変数値を有しており、
前記第一の部分についての前記置き換え値が、前記二つ以上のシンタックス要素のそれぞれについての状態変数値を、前記修正された符号化ストリームにおいて前記置き換え値

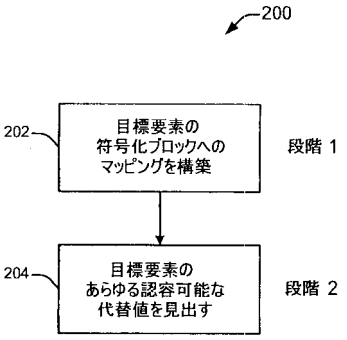
の代わりに前記第一の部分が使用されたとした場合と同じに維持する、
態様２３記載の方法。
〔態様３６〕
プロセッサ読み取り可能な媒体を有する装置であって：
少なくとも二つの部分を含み、第二の部分は該第二の部分のほか第一の部分にも基づく結果を生じるようデコードされることができる、符号化されたビットストリームにアクセスする段階と；
前記第二の部分のデコードが相変わらず前記結果を生じるような修正された符号化ビットストリームを生成するよう、前記第一の部分を置き換え値で置き換える段階とを少なくとも実行するための、
前記プロセッサ読み取り可能な媒体上に記憶された命令を含む、装置。
〔態様３７〕
プロセッサ読み取り可能な媒体を有する装置であって、前記プロセッサ読み取り可能な媒体上には修正された第一の部分と第二の部分とを含むエンコードされたデータが記憶されており、前記修正された第一の部分は第一の部分の置き換えであり：
前記第二の部分は、該第二の部分のほか前記修正された第一の部分にも基づく結果を生じるようデコードされることができ、
前記結果は、前記第二の部分のデコードが前記第二の部分および前記修正された第一の部分ではなく前記第二の部分および前記第一の部分に基づいて実行されたとした場合と同じである、
装置。
〔態様３８〕
前記修正された第一の部分は、前記エンコードされたデータに含まれる透かしの少なくとも一部である、態様３７記載の装置。
〔態様３９〕
修正された第一の部分と第二の部分とを含むエンコードされたデータを含むようフォーマットされた信号であって、前記修正された第一の部分は第一の部分の置き換えであり：
前記第二の部分は、該第二の部分のほか前記修正された第一の部分にも基づく結果を生じるようデコードされることができ、
前記結果は、前記第二の部分のデコードが前記第二の部分および前記修正された第一の部分ではなく前記第二の部分および前記第一の部分に基づいて実行されたとした場合と同じである、
信号。
〔態様４０〕
前記修正された第一の部分は、CABACエンコードされたビットストリーム内の少なくとも一つの符号化されたシンタックス要素を含む、態様３９記載の信号。
〔態様４１〕
前記修正された第一の部分は、修正前の第一の部分のビット長に等しいビット長をもつ、態様４０記載の信号。
〔態様４２〕
前記修正された第一の部分は、前記第一の部分が生成するであろうのと同じ、前記第二の部分のCABACデコードに必要な状態変数を生成する、態様４０記載の信号。
〔態様４３〕
修正された第一の部分と第二の部分とを含む修正されたデータのセットにアクセスする段階であって、前記修正された第一の部分は第一の部分の修正から帰結するものであり、前記第二の部分はデコードされて前記第二の部分のほか前記修正された第一の部分にも基づく結果を生成することができる、段階と；
前記第二の部分を前記結果を生成するようデコードする段階であって、前記結果は、前記第二の部分のデコードが、前記第二の部分および前記修正された第一の部分ではなく前記第二の部分および前記第一の部分に基づいて実行されたとした場合と同じである、段階

と；
前記修正された第一の部分をデコードする段階とを有する、
方法。
〔態様４４〕
前記修正された第一の部分が、ある情報の値に基づいてなされた修正を反映しており、当該方法がさらに、前記修正された第一の部分のデコードに基づいて前記情報の値を判別する段階を有する、態様４３記載の方法。
〔態様４５〕
前記修正された第一の部分が前記修正されたデータのセットに含まれる透かしの少なくとも一部である、態様４３記載の方法。
〔態様４６〕
前記のアクセスされた修正されたデータのセットが、CABACエンコードされたビットストリームを含み、前記第一の部分が少なくとも一つの符号化されたシンタックス要素を含む、態様４３記載の方法。
〔態様４７〕
前記修正された第一の部分が修正されたシンタックス要素を含み、前記修正されたシンタックス要素が、前記CABACエンコードされたビットストリームについて前記第一の部分と同じデコード変数を与える、態様４６記載の方法。
〔態様４８〕
前記修正された第一の部分が、前記第一の部分のビット長に等しいビット長をもつ修正されたシンタックス要素を含み、前記修正された第一の部分が、前記第一の部分によって生成されたであろうL値に等しいCABACエンコーダにおけるL値を与える、態様４６記載の方法。
〔態様４９〕
プロセッサ読み取り可能な媒体を有する装置であって：
修正された第一の部分と第二の部分とを含む修正されたデータのセットにアクセスする段階であって、前記修正された第一の部分は第一の部分の修正から帰結するものであり、前記第二の部分はデコードされて前記第二の部分のほか前記修正された第一の部分にも基づく結果を生成することができる、段階と；
前記第二の部分を前記結果を生成するようデコードする段階であって、前記結果は、前記第二の部分のデコードが、前記第二の部分および前記修正された第一の部分ではなく前記第二の部分および前記第一の部分に基づいて実行されたとした場合と同じである、段階と；
前記修正された第一の部分をデコードする段階とを少なくとも実行するための、
前記プロセッサ読み取り可能な媒体上に記憶された命令を含む、装置。

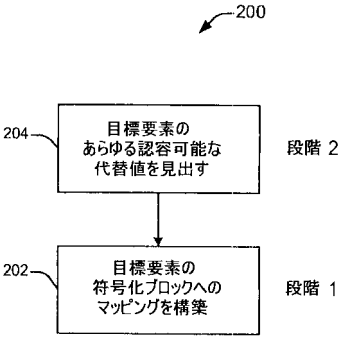
【図１】



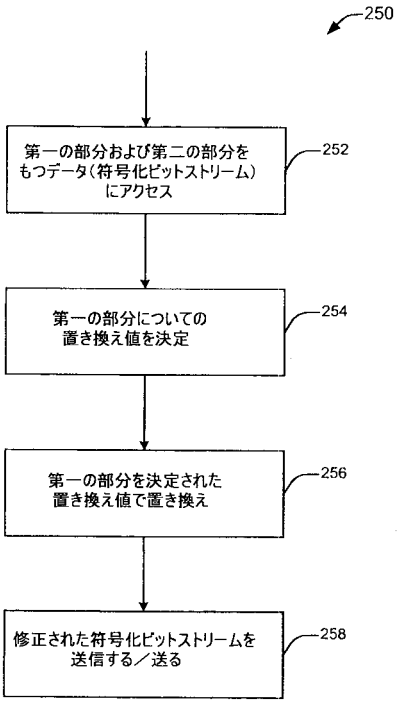
【図２ａ】



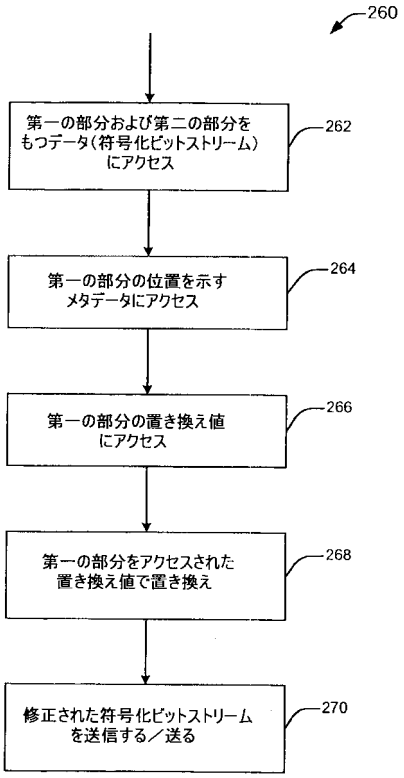
【図２ｂ】



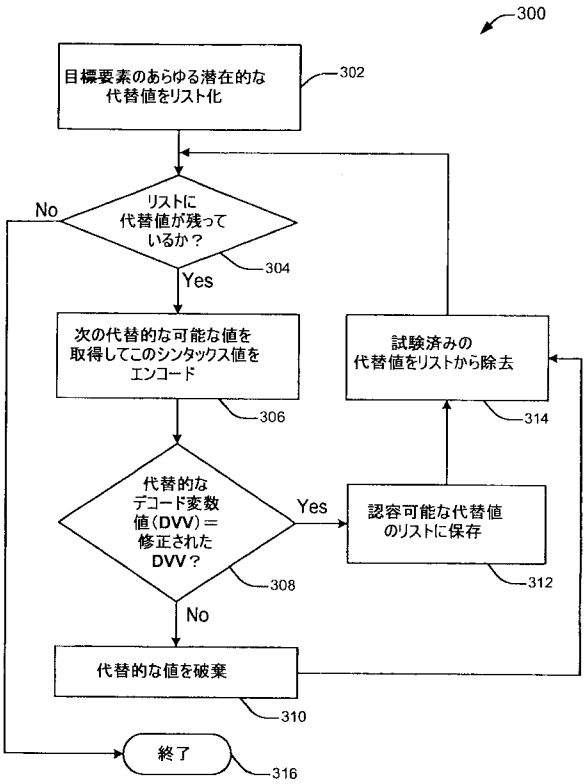
【図２ｃ】



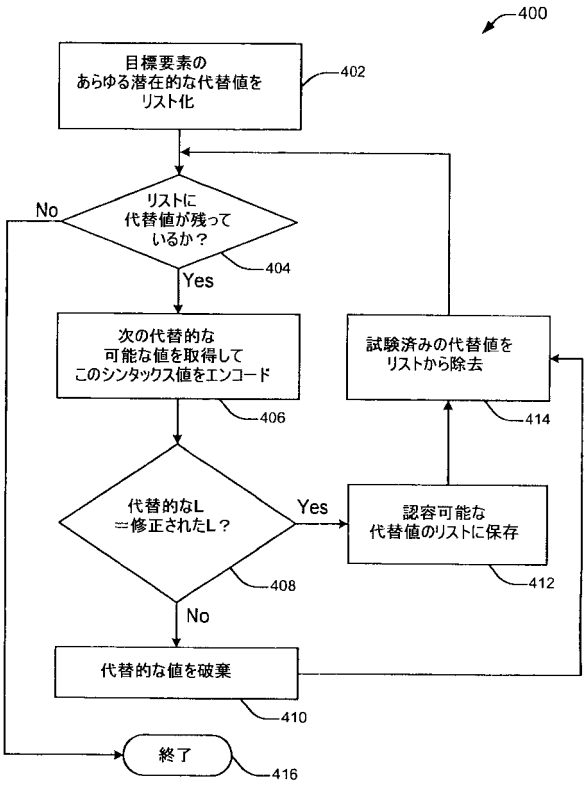
【図２ｄ】



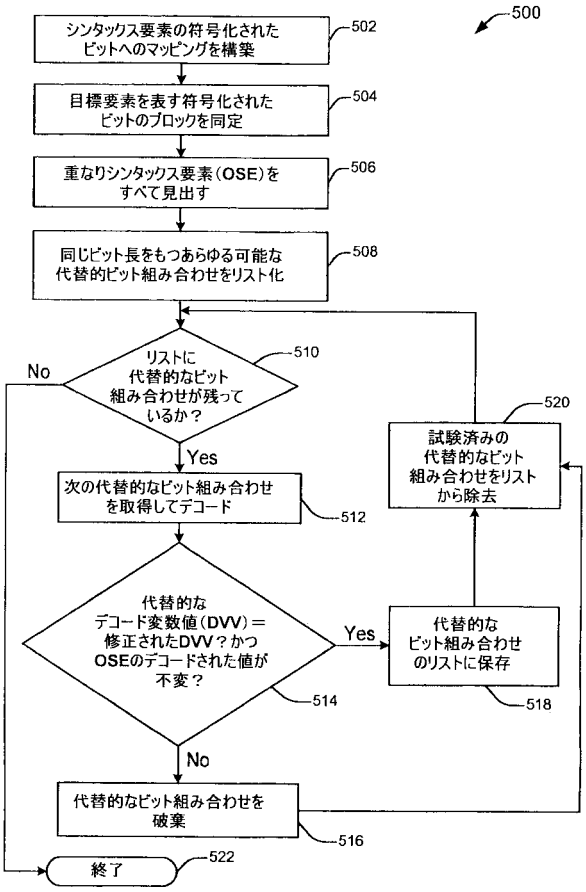
【図３】


300
目標要素のあらゆる潜在的な代替値をリスト化
302
リストに代替値が残っているか？
304
No
Yes
次の代替的な可能な値を取得してこのシンタックス値をエンコード
306
代替的なデコード変数値(DVV)＝修正されたDVV？
308
Yes
認容可能な代替値のリストに保存
312
試験済みの代替値をリストから除去
314
No
代替的な値を破棄
310
終了
316

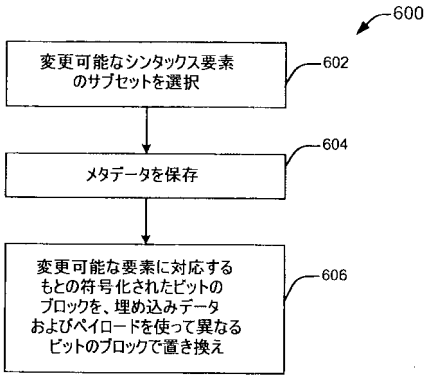
【図４】


400
目標要素のあらゆる潜在的な代替値をリスト化
402
リストに代替値が残っているか？
404
No
Yes
次の代替的な可能な値を取得してこのシンタックス値をエンコード
406
代替的なL＝修正されたL？
408
Yes
認容可能な代替値のリストに保存
412
試験済みの代替値をリストから除去
414
No
代替的な値を破棄
410
終了
416

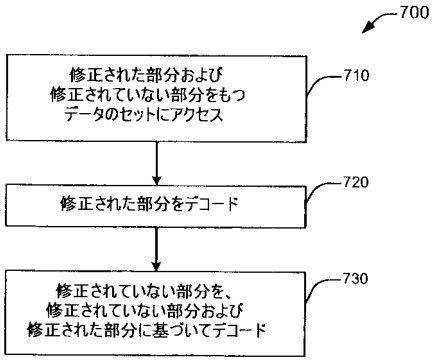
【図５】


500
シンタックス要素の符号化されたビットへのマッピングを構築
502
目標要素を表す符号化されたビットのブロックを同定
504
重なりシンタックス要素(OSE)をすべて見出す
506
同じビット長をもつあらゆる可能な代替的ビット組み合わせをリスト化
508
リストに代替的なビット組み合わせが残っているか？
510
No
Yes
次の代替的なビット組み合わせを取得してデコード
512
代替的なデコード変数値(DVV)＝修正されたDVV？かつOSEのデコードされた値が不変？
514
Yes
代替的なビット組み合わせのリストに保存
518
試験済みの代替的なビット組み合わせをリストから除去
520
No
代替的なビット組み合わせを破棄
516
終了
522

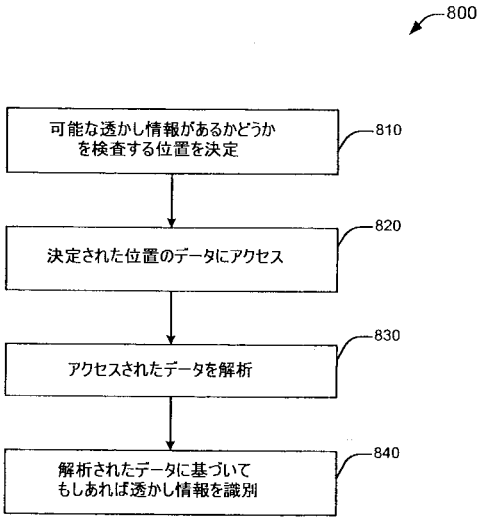
【図６】


600
変更可能なシンタックス要素のサブセットを選択
602
メタデータを保存
604
変更可能な要素に対応するもとの符号化されたビットのブロックを、埋め込みデータおよびペイロードを使って異なるビットのブロックで置き換え
606

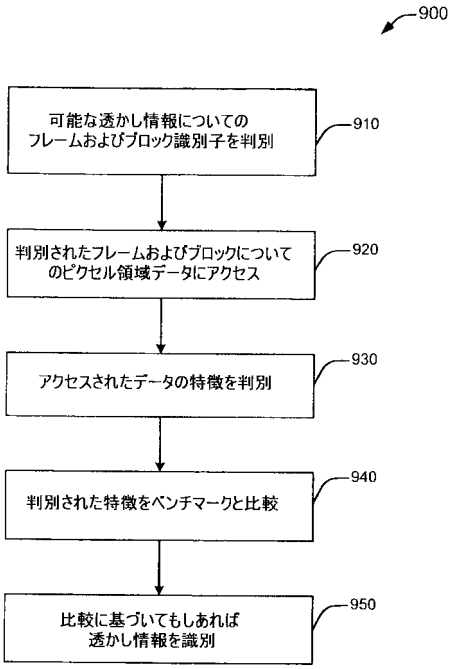
【図７】


700
修正された部分および修正されていない部分をもつデータのセットにアクセス
710
修正された部分をデコード
720
修正されていない部分を、修正されていない部分および修正された部分に基づいてデコード
730

【図８】


800
可能な透かし情報があるかどうか
を検査する位置を決定
810
決定された位置のデータにアクセス
820
アクセスされたデータを解析
830
解析されたデータに基づいて
もしあれば透かし情報を識別
840

【図９】


900
可能な透かし情報についての
フレームおよびブロック識別子を判別
910
判別されたフレームおよびブロックについて
のピクセル領域データにアクセス
920
アクセスされたデータの特徴を判別
930
判別された特徴をベンチマークと比較
940
比較に基づいてもしあれば
透かし情報を識別
950

フロントページの続き

(72)発明者　ゾウ，デクン
アメリカ合衆国，ニュージャージー州　０８８１７，プレインズボロ，ハンターズ・グレン・ドライヴ　５６１３
(72)発明者　ブルーム，ジェフリー，アダム
アメリカ合衆国，ニュージャージー州　０８８５０，ウエスト・ウィンザー，バーリントン・ドライヴ　１４
(72)発明者　イン，ペン
アメリカ合衆国，ニュージャージー州　０８５５０，ウエスト・ウィンザー，ウォーウィック・ロード　６５
(72)発明者　ディヴォラ　エスコーダ，オスカー
アメリカ合衆国，ニュージャージー州　０８５４０，プリンストン，セイアー・ドライヴ　１２０１

審査官　畑中　高行

(56)参考文献　特開２００６－０９９７８０（ＪＰ，Ａ）
特開２００３－３１９１６６（ＪＰ，Ａ）
Bijan G. Mobasseri and Yatish Naik Raikar, Authentication of H.264 streams by direct watermarking of CAVLC blocks, SECURITY. STEGANOGRAPHY, AND WATERMARKING OF MULTIMEDIA CONTENTS IX., SPIE-IS&T, ２００７年　2月２７日, Vol.6505, p.65051W-1 - 65051W-5

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｈ０４Ｎ７／２４－７／６８
Ｈ０４Ｎ１／４１－１／４１９
Ｈ０３Ｍ３／００－１１／００