(19)日本国特許庁(JP)　(12)公表特許公報(A)　(11)公表番号
特表2025-512264
(P2025-512264A)
(43)公表日　令和7年4月17日(2025.4.17)

(51)国際特許分類		FI		テーマコード（参考）
H04N 19/85 (2014.01)		H04N	19/85	5C159
H04N 19/70 (2014.01)		H04N	19/70	

審査請求　未請求　予備審査請求　未請求　（全87頁）

(21)出願番号　特願2024-554998(P2024-554998)
(86)(22)出願日　令和5年4月11日(2023.4.11)
(85)翻訳文提出日　令和6年11月7日(2024.11.7)
(86)国際出願番号　PCT/AU2023/050289
(87)国際公開番号　WO2023/197029
(87)国際公開日　令和5年10月19日(2023.10.19)
(31)優先権主張番号　2022202470
(32)優先日　令和4年4月13日(2022.4.13)
(33)優先権主張国・地域又は機関
オーストラリア(AU)
(81)指定国・地域　AP(BW,CV,GH,GM,KE,LR,LS,MW,MZ,NA,RW,SD,SL,ST,SZ,TZ,UG,ZM,ZW),EA(AM,AZ,BY,KG,KZ,RU,TJ,TM),EP(AL,AT,BE,BG,CH,CY,CZ,DE,DK,EE,ES,FI,FR,GB,GR,HR,HU,IE,IS,IT,LT,LU,LV
最終頁に続く

(71)出願人　000001007
キヤノン株式会社
東京都大田区下丸子３丁目３０番２号
(74)代理人　110003281
弁理士法人大塚国際特許事務所
(72)発明者　ロゼワーン， クリストフアー ジエームズ
オーストラリア国 ２１１３ ニューサウスウェールズ州 タラベラロード パーク エステイト５ ビルディング エー キヤノン オーストラリア プロプライエタリー リミテッド内
(72)発明者　グエン， ティ ホン ニュン
オーストラリア国 ２１１３ ニューサウスウェールズ州 タラベラロード パーク エステイト５ ビルディング エー キヤ
最終頁に続く

(54)【発明の名称】　テンソルを符号化および復号するための方法、装置、およびシステム

(57)【要約】
ビットストリームからテンソルを復号する方法であって、テンソルのチャネルにわたる平均テンソル値に対応する第１ユニットをビットストリームから復号することと、テンソルの分解の成分に対応する第２ユニットをビットストリームから復号することと、第１ユニットに関連する係数に対応する第３ユニットをビットストリームから復号することと、第２ユニットに関連する係数に対応する第４ユニットをビットストリームから復号することとを備える。本方法はそれぞれのチャネルについての係数に平均テンソル値とテンソルの分解に関連する係数を有するテンソルの分解の成分とを乗算することによって、チャネルごとにテンソルを生成することをさらに備え、ユニットのうちの少なくとも１つは、他のユニットのうちの少なくとも１つに関して独立して復号可能であり、各ユニットはサンプルの複数の２次元配列中に配置される。

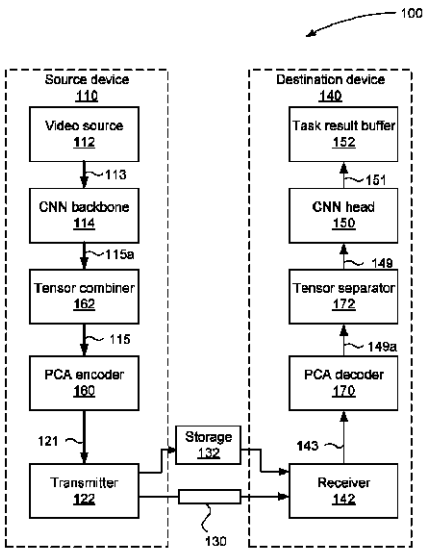


Fig. 1

【特許請求の範囲】
【請求項１】
ビットストリームからテンソルを復号する方法であって、
前記ビットストリームから情報の第１ユニットを復号することであって、情報の前記第１ユニットは、前記テンソルのチャネルにわたる平均テンソル値に対応する、復号することと、
前記ビットストリームから情報の第２ユニットを復号することであって、情報の前記第２ユニットは、前記テンソルの分解の成分に対応する、復号することと、
前記ビットストリームから情報の第３ユニットを復号することであって、情報の前記第３ユニットは、前記テンソルの情報の前記第１ユニットに関連する係数に対応する、復号することと、
前記ビットストリームから情報の第４ユニットを復号することであって、情報の前記第４ユニットは、前記テンソルの情報の前記第２ユニットに関連する係数に対応する、復号することと、
それぞれのチャネルについての前記係数に、前記平均テンソル値と、前記テンソルの前記分解に関連する係数を有する前記テンソルの前記分解の前記成分と、を乗算することによって、チャネルごとに前記テンソルを生成することと、を含み、前記ビットストリームからの情報の前記第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の前記他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される、前記方法。
【請求項２】
情報の前記第１、第２、第３、および第４ユニットの全ては、互いに独立して復号可能である請求項１に記載の方法。
【請求項３】
情報の前記第１、第２、第３、および第４ユニットは、水平に配置されたサブピクチャであり、ピクチャの幅を定義する、請求項１に記載の方法。
【請求項４】
情報の前記第１、第２、第３、および第４ユニットは、水平に配置されたサブピクチャであり、ピクチャの幅を定義し、情報の前記第４ユニットは、前記第１、第２、および第３ユニットの下に配置され、前記ピクチャの幅にまたがり、前記ピクチャの底部まで延在するサブピクチャである請求項１に記載の方法。
【請求項５】
前記方法は複数のテンソルに適用され、情報の前記第１、第２、第３、および第４ユニットの各々は、前記複数のテンソルの各々に対するそれぞれの情報を含む、請求項１に記載の方法。
【請求項６】
情報の前記第３ユニットは、２次元サンプル配列に配置され、前記サンプル配列のアスペクト比は、前記情報に基づいて、１：１または可能な限り１：１に近いように調整される、請求項１に記載の方法。
【請求項７】
情報の前記第３ユニットは、２次元サンプル配列に配置され、前記サンプル配列のアスペクト比は、前記情報に基づいて、２：１または可能な限り２：１に近いように調整される、請求項１に記載の方法。
【請求項８】
情報の前記第１、第２、第３、および第４ユニットの各々は、多用途ビデオ符号化のためのサブピクチャとして配置される、請求項１に記載の方法。
【請求項９】
情報の前記第１、第２、第３、および第４ユニットの各々は、多用途ビデオ符号化または高効率ビデオ符号化のためのタイルとして配置される、請求項１に記載の方法。
【請求項１０】

情報の前記第１、第２、第３、および第４ユニットの各々は、多用途ビデオ符号化または高効率ビデオ符号化のためのスライスとして配置される、請求項１に記載の方法。
【請求項１１】
テンソルをビットストリームに符号化する方法であって、
情報の第１ユニットを符号化することであって、情報の前記第１ユニットは、前記テンソルのチャネルにわたる平均テンソル値に対応する、符号化することと、
情報の第２ユニットを符号化することであって、情報の前記第２ユニットは、前記テンソルの分解の成分に対応する、符号化することと、
情報の第３ユニットを符号化することであって、情報の前記第３ユニットは、前記テンソルの情報の前記第１ユニットに関する係数に対応する、符号化することと、
情報の第４ユニットを符号化することであって、情報の前記第４ユニットは、前記テンソルの情報の前記第２ユニットに関する係数に対応する、符号化することと、
を含み、
情報の前記第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される、前記方法。
【請求項１２】
情報の前記第１、第２、第３、および第４ユニットの全ては、互いに独立して復号可能である請求項１１に記載の方法。
【請求項１３】
情報の前記第１、第２、第３、および第４ユニットは、水平に配置されたサブピクチャであり、ピクチャの幅を定義する、請求項１１に記載の方法。
【請求項１４】
情報の前記第１、第２、第３、および第４ユニットは、水平に配置されたサブピクチャであり、ピクチャの幅を定義し、情報の前記第４ユニットは、前記第１、第２、および第３ユニットの下に配置され、前記ピクチャの幅にまたがり、前記ピクチャの底部まで延在するサブピクチャである請求項１１に記載の方法。
【請求項１５】
前記方法は複数のテンソルに適用され、情報の前記第１、第２、第３、および第４ユニットの各々は、前記複数のテンソルの各々に対するそれぞれの情報を含む、請求項１１に記載の方法。
【請求項１６】
情報の前記第３ユニットは、２次元サンプル配列に配置され、前記サンプル配列のアスペクト比は、前記情報に基づいて、１：１または可能な限り１：１に近いように調整される、請求項１１に記載の方法。
【請求項１７】
情報の前記第３ユニットは、２次元サンプル配列に配置され、前記サンプル配列のアスペクト比は、前記情報に基づいて、２：１または可能な限り２：１に近いように調整される、請求項１１に記載の方法。
【請求項１８】
情報の前記第１、第２、第３、および第４ユニットの各々は、多用途ビデオ符号化のためのサブピクチャとして配置される、請求項１１に記載の方法。
【請求項１９】
情報の前記第１、第２、第３、および第４ユニットの各々は、多用途ビデオ符号化または高効率ビデオ符号化のためのタイルとして配置される、請求項１１に記載の方法。
【請求項２０】
情報の前記第１、第２、第３、および第４ユニットの各々は、多用途ビデオ符号化または高効率ビデオ符号化のためのスライスとして配置される、請求項１１に記載の方法。
【請求項２１】
ビットストリームからテンソルを復号する方法を実行するためのプログラムを格納する非

一時的なコンピュータ可読格納媒体であって、前記方法は、
前記ビットストリームから情報の第１ユニットを復号することであって、情報の前記第１ユニットは、前記テンソルのチャネルにわたる平均テンソル値に対応する、復号することと、
前記ビットストリームから情報の第２ユニットを復号することであって、情報の前記第２ユニットは、前記テンソルの分解の成分に対応する、復号することと、
前記ビットストリームから情報の第３ユニットを復号することであって、情報の前記第３ユニットは、前記テンソルの情報の前記第１ユニットに関連する係数に対応する、復号することと、
前記ビットストリームから情報の第４ユニットを復号することであって、情報の前記第４ユニットは、前記テンソルの情報の前記第２ユニットに関連する係数に対応する、復号することと、
それぞれのチャネルについての前記係数に、前記平均テンソル値と、前記テンソルの前記分解に関連する係数を有する前記テンソルの前記分解の前記成分と、を乗算することによって、チャネルごとに前記テンソルを生成することと、を含み、前記ビットストリームからの情報の前記第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の前記他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される。
【請求項２２】
ビットストリームからテンソルを復号するデコーダであって、前記デコーダは、
前記ビットストリームから情報の第１ユニットを復号し、情報の前記第１ユニットは、前記テンソルのチャネルにわたる平均テンソル値に対応し、
前記ビットストリームから情報の第２ユニットを復号し、情報の前記第２ユニットは、前記テンソルの分解の成分に対応し、
前記ビットストリームから情報の第３ユニットを復号し、情報の前記第３ユニットは、前記テンソルの情報の前記第１ユニットに関連する係数に対応し、
前記ビットストリームから情報の第４ユニットを復号し、情報の前記第４ユニットは、前記テンソルの情報の前記第２ユニットに関連する係数に対応し、
それぞれのチャネルについての前記係数に、前記平均テンソル値と、前記テンソルの前記分解に関連する係数を有する前記テンソルの前記分解の前記成分と、を乗算することによって、チャネルごとに前記テンソルを生成する
ように構成され、前記ビットストリームからの情報の前記第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の前記他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される。
【請求項２３】
テンソルをビットストリームに符号化する方法を実行するためのプログラムを格納する非一時的なコンピュータ可読格納媒体であって、前記方法は、
情報の第１ユニットを符号化することであって、情報の前記第１ユニットは、前記テンソルのチャネルにわたる平均テンソル値に対応する、符号化することと、
情報の第２ユニットを符号化することであって、情報の前記第２ユニットは、前記テンソルの分解の成分に対応する、符号化することと、
情報の第３ユニットを符号化することであって、情報の前記第３ユニットは、前記テンソルの情報の前記第１ユニットに関する係数に対応する、符号化することと、
情報の第４ユニットを符号化することであって、情報の前記第４ユニットは、前記テンソルの情報の前記第２ユニットに関する係数に対応する、符号化することと、
を含み、
情報の前記第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される。

【請求項２４】
テンソルをビットストリームに符号化するエンコーダであって、前記エンコーダは、
情報の第１ユニットを符号化し、情報の前記第１ユニットは、前記テンソルのチャネルにわたる平均テンソル値に対応し、
情報の第２ユニットを符号化し、情報の前記第２ユニットは、前記テンソルの分解の成分に対応し、
情報の第３ユニットを符号化し、情報の前記第３ユニットは、前記テンソルの情報の前記第１ユニットに関する係数に対応し、
情報の第４ユニットを符号化し、情報の前記第４ユニットは、前記テンソルの情報の前記第２ユニットに関する係数に対応する、
ように構成され、情報の前記第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される。
【発明の詳細な説明】
【技術分野】
【０００１】
関連出願への言及
本出願は２０２２年４月１３日に出願されたオーストラリア国特許出願第２０２２２０２４７０号の出願日の３５Ｕ．Ｓ．Ｃ§１１９に基づく利益を主張し、その全体が本明細書に完全に記載されているかのように、参照により本明細書に組み込まれる
本発明は一般に、デジタルビデオ信号処理に関し、特に、畳み込みニューラルネットワークからテンソルを符号化および復号するための方法、装置、およびシステムに関する。本発明はまた、ビデオ圧縮技術を使用して畳み込みニューラルネットワークからテンソルを符号化および復号するためのコンピュータプログラムを記録したコンピュータ可読媒体を含むコンピュータプログラム製品に関する。
【背景技術】
【０００２】
ビデオ圧縮は、ビデオデータの送信および記憶のためのアプリケーションを含む、多くのアプリケーションをサポートするために使用されるユビキタス技術である。多くのビデオコーディング規格が開発されており、他のものは現在開発中である。ビデオコーディング標準化における最近の発展は、「Joint Video Experts Team」（ＪＶＥＴ）と呼ばれるグループの形成につながった。Joint Video Experts Team（JVET）は、２つの標準設定組織（Standards Setting Organisations）（SSO）、つまり、「Video Coding Experts Group」（ＶＣＥＧ）としても知られる、International Telecommunication Union（ITU）のTelecommunication Standardisation Sector（ＩＴＵ－Ｔ）のStudy Group １６、Ｑｕｅｓｔｉｏｎ ６（ＳＧ１６／Ｑ６）と、「Moving Picture Experts Group」（ＭＰＥＧ）としても知られる、国際標準化機構／国際電気標準会議合同技術委員会１／小委員会２９／Working Group １１(ISO／IEC ＪＴＣ１／ＳＣ２９／ＷＧ１１）と、のメンバを含む。
【０００３】
Joint Video Experts Team(JVET)は、「versatile video coding」（ＶＶＣ）と呼ばれるビデオ圧縮規格を開発した。
【０００４】
畳み込みニューラルネットワーク（ＣＮＮ）は、とりわけ、マシンビジョンを伴うユースケースに対処する新興技術である。ＣＮＮは典型的には全結合レイヤを含む畳み込みレイヤなどの多くのレイヤを含み、データは、「テンソル」の形態で１つのレイヤから次のレイヤに渡される。各レイヤの重みは、トレーニング段階で決定され、ここで、非常に大量のトレーニングデータがＣＮＮを通過し、決定された結果がトレーニングデータに関連するグラウンドトゥルースと比較される。確率的勾配降下などのネットワーク重みを更新するためのプロセスは、ネットワークが所望のレベルの精度で実行するまで、ネットワー

ク重みを反復的に精緻化するために適用される。畳み込みステージが１より大きい「ストライド」を有する場合、畳み込みからの出力テンソルは、対応する入力テンソルよりも低い空間分解能を有する。「max pooling」（または「Ｍａｘｐｏｏｌ」）などの演算も、入力テンソルと比較して出力テンソルの空間サイズを縮小する。最大プーリングは入力テンソルをデータサンプルのグループ（たとえば、データサンプルの２×２グループ）に分割することによって出力テンソルを生成し、各グループから、出力テンソル内の対応する値のための出力として最大値を選択する。入力を用いてＣＮＮを実行し、入力を出力に漸進的に変換するプロセスは、一般に「推論」と呼ばれる。

【０００５】

一般に、テンソルは４つの次元、すなわち、バッチ、チャネル、高さおよび幅を有する。ビデオデータを推論するときのサイズ「１」の第１次元「バッチ」は、一度に１つのフレームがＣＮＮを通過することを示す。ネットワークをトレーニングするとき、所定の「バッチサイズ」に従って、ネットワークの重みが更新される前に複数のフレームがネットワークを通過するように、バッチ次元の値を増加させることができる。マルチフレームビデオは、所与のビデオのフレームの数に従ってサイズが増加されたバッチ次元を有する単一のテンソルとして通過され得る。しかしながら、メモリ消費およびアクセスに関する実際的な考慮事項のために、ビデオデータの推論は、典型的にはフレーム単位で実行される。「チャネル」次元は、所与のテンソルに対する同時（concurrent）の「特徴マップ」の数を示す。高さおよび幅の次元は、ＣＮＮの特定のステージにおける特徴マップのサイズを示す。チャネルカウントは、ネットワークアーキテクチャに従ってＣＮＮを介して変化する。また、特徴マップサイズは、特定のネットワークレイヤで発生するサブサンプリングに応じて変化する。

【０００６】

ＣＮＮの第１レイヤへの入力は、典型的には第１レイヤへのテンソル入力の次元との互換性のためにリサイズされる、画像またはビデオフレームである。テンソルの次元はＣＮＮアーキテクチャに依存し、一般に、入力幅および高さに関連するいくつかの次元と、さらなる「チャネル」次元と、を有する。

【０００７】

チャネル次元に基づいてテンソルをスライスすることはいわゆる「特徴マップ」のセットをもたらし、これはテンソルの各スライスが対応する入力画像と何らかの関係を有し、様々なエッジタイプなどの特性を捕捉するためである。ネットワークへの入力からさらに離れたレイヤでは、プロパティがより抽象的であり得る。ＣＮＮの「タスク性能」は、特定の入力を用いてタスクを実行する際のＣＮＮの結果を、一般に人間によって準備され、「正しい」結果を示すとみなされる、提供されたグラウンドトゥルースと比較することによって測定される。

【０００８】

ネットワークトポロジが決定されると、ネットワーク重みは、より多くの訓練データが利用可能になるにつれて時間更新され得る。また、ＣＮＮの一部分を再トレーニングし、ネットワークの他の部分（複数可）における重みを変更しないままにすることも可能である。ＣＮＮの全体的な複雑さは高くなる傾向があり、比較的多数の積和演算が実行され、多数の中間テンソルがメモリに書き込まれ、メモリから読み出される。いくつかのアプリケーションでは、ＣＮＮが完全に「クラウド」に実装され、高価で高価な処理能力が必要となる。他のアプリケーションでは、ＣＮＮがカメラまたは携帯電話などのエッジデバイスに実装され、柔軟性は低くなるが、処理負荷はより分散される。新しいアーキテクチャは、ネットワークを部分に分割することを含み、そのうちの１つはエッジデバイスで実行され、別の１つはクラウドで実行される。そのような分散ネットワークアーキテクチャは、「協調インテリジェンス」と呼ばれることがある。協調インテリジェンスは、ネットワークの第１部分からの部分的な結果を、いくつかの異なる第２部分、おそらく、各部分は異なるタスクに対して最適化される、と共に、再使用することなどの利点を提供する。協調インテリジェンスアーキテクチャは、ＷＡＮなどのネットワークを介した送信のために

、テンソルデータの効率的な圧縮の必要性を導入する。テンソルデータは、クラウド内のサーバ上に、または様々なタスクのために後で使用される部分的に処理されたデータとしてデバイス内に圧縮された形態で記憶され得る。

【０００９】

ＶＶＣは特に、ビデオフォーマットが（たとえば、より高い解像度およびより高いフレームレートを有する）能力を増加させるにつれて、ますます高い圧縮性能に対する継続的な需要に対処し、帯域幅コストが比較的高いＷＡＮを介したサービス配信に対する増大する市場需要に対処することが予想される。ＶＶＣは、現代のシリコンプロセスにおいて実施可能であり、達成された性能と実施コストとの間の許容可能なトレードオフを提供する。実装コストはたとえば、シリコンエリア、ＣＰＵプロセッサロード、メモリ利用率、および帯域幅のうちの１つまたは複数に関して考慮され得る。ＶＶＣ規格の汎用性の一部は、ビデオデータを圧縮するために利用可能なツールの幅広い選択、ならびにＶＶＣが適している広範囲のアプリケーションにある。

【００１０】

ビデオデータは、画像データのフレームのシーケンスを含み、各フレームは、１つまたは複数のカラーチャネルを含む。一般に、１つの一次カラーチャネルおよび２つの二次カラーチャネルが必要とされる。一次カラーチャネルは一般に「ルマ」チャネルと呼ばれ、二次カラーチャネルは一般に「クロマ」チャネルと呼ばれる。ビデオデータは典型的にはＲＧＢ（赤－緑－青）色空間で表示されるが、ＲＧＢ色空間は３つのそれぞれの構成要素の間に高い相関度を有する。エンコーダまたはデコーダによって見られるビデオデータ表現はしばしば、ＹＣｂＣｒなどの色空間を使用している。ＹＣｂＣｒは、伝達関数に従って「ルマ」にマッピングされた輝度を、Ｙ（一次）チャネルに集中させ、ＣｂおよびＣｒ（二次）チャネルに彩度を集中させる。非相関ＹＣｂＣｒ信号の使用により、ルマチャネルの統計は、クロマチャネルの統計と著しく異なる。主要な差異は、量子化の後、クロマチャネルが対応するルマチャネルブロックの係数と比較して、所与のブロックのための比較的少数の有意係数を含むことである。さらに、ＣｂおよびＣｒチャネルは、「４：２：０クロマフォーマット」として知られる、ルマチャネルと比較してより低いレート、例えば、水平方向に半分および垂直方向に半分で空間的にサンプリング（サブサンプリング）されてもよい。４：２：０クロマフォーマットは、インターネットビデオストリーミング、ブロードキャストテレビジョン、Blu-Ray™ディスクへの保存など、「コンシューマ」アプリケーションで一般的に使用される。ルマサンプルのみが存在する場合、結果として生じるモノクロフレームは、「４：０：０クロマフォーマット」を使用すると言われる。

【００１１】

ＶＶＣ規格は、「ブロックベース」アーキテクチャを規定し、フレームは最初に、「コーディングツリーユニット」（ＣＴＵ）として知られる領域の正方形アレイに分割される。一般に、ＣＴＵは、１２８×１２８ルマサンプルのような比較的大きな面積を占める。ＶＶＣ規格を使用するときの他の可能なＣＴＵサイズは、３２×３２および６４×６４である。ただし、各フレームの右端および下端のＣＴＵは、面積がより小さくなり得、暗黙の分割はＣＢがフレーム内に留まることを確実にすることを生じる。各ＣＴＵに関連付けられているのは、ルマチャネルおよびクロマチャネルの両方のための「コーディングツリー」（「共有ツリー」）、またはルマチャネルおよびクロマチャネルのそれぞれのための別個のツリーである。コーディングツリーは、ＣＴＵのエリアの、「コーディングブロック」（ＣＢ）とも呼ばれるブロックのセットへの分解を定義する。共有ツリーが使用中であるとき、単一のコーディングツリーはルマチャネルおよびクロマチャネルの両方のためのブロックを指定し、その場合、コロケートされたコーディングブロックのコレクションは「コーディングユニット」（ＣＵ）と呼ばれる（すなわち、各カラーチャネルのためのコーディングブロックを有する各ＣＵ）。ＣＢは、特定の順序で符号化または復号するために処理される。４：２：０クロマフォーマットの使用の結果として、１２８×１２８ルマサンプルエリアのためのルマコーディングツリーを有するＣＴＵは、１２８×１２８ル

マサンプルエリアと並置された６４×６４クロマサンプルエリアのための対応するクロマコーディングツリーを有する。単一のコーディングツリーがルマチャネルおよびクロマチャネルのために使用されているとき、所与のエリアのためのコロケートされたブロックのコレクションは概して、「ユニット」、たとえば、上記のＣＵ、ならびに「予測ユニット」（ＰＵ）および「変換ユニット」（ＴＵ）と呼ばれる。４：２：０クロマフォーマットビデオデータのカラーチャネルにまたがるＣＵを有する単一のツリーは、クロマブロックを、対応するルマブロックの幅および高さの半分にする。所与のエリアのために別個のコーディングツリーが使用されるとき、上述のＣＢ、ならびに「予測ブロック」（ＰＢ）および「変換ブロック」（ＴＢ）が使用される。

【００１２】

「ユニット」と「ブロック」との間の上記の区別にもかかわらず、「ブロック」という用語は、すべてのカラーチャネルに動作が適用されるフレームのエリアまたは領域の総称として使用され得る。

【００１３】

ＣＵごとに、フレームデータの対応する領域のコンテンツ（サンプル値）の予測ユニット（ＰＵ）が生成される（「予測ユニット」）。さらに、予測と、エンコーダへの入力で見られる領域の内容と、の間の差分（または「空間領域」残差）の表現が形成される。各カラーチャネルにおける差分は、残差係数のシーケンスとして変換され、コーディングされ得、所与のＣＵのための１つまたは複数のＴＵを形成する。適用される変換は、残差値の各ブロックに適用される、離散コサイン変換（ＤＣＴ）または他の変換であり得る。変換は分離可能に適用される（すなわち、２次元変換は、２つのパスで実行される）。ブロックは最初に、ブロック内のサンプルの各行に１次元変換を適用することによって変換される。次いで、部分結果は、部分結果の各列に１次元変換を適用することによって変換され、残差サンプルを実質的に非相関化する変換係数の最終ブロックを生成する。様々なサイズの変換は、矩形形状ブロックの変換を含むＶＶＣ規格によってサポートされ、各サイド寸法は２のべき乗である。変換係数は、ビットストリームへのエントロピー符号化のために量子化される。

【００１４】

ＶＶＣは、イントラフレーム予測およびインターフレーム予測を特徴とする。イントラフレーム予測は、フレーム内のデータサンプルの現在のブロックの予測を生成するために使用されているフレーム内の以前に処理されたサンプルの使用を伴う。インターフレーム予測は、以前に復号されたフレームから取得されたサンプルのブロックを使用して、フレーム中のサンプルの現在のブロックの予測を生成することを伴う。以前に復号されたフレームから取得されたサンプルのブロックは、しばしばフィルタリングが適用された動きベクトルに従って、現在のブロックの空間位置からオフセットされる。イントラフレーム予測ブロックは、（ｉ）均一サンプル値（「ＤＣイントラ予測」）、（ｉｉ）オフセットおよび水平および垂直勾配を有するプレーン（「平面イントラ予測」）、（ｉｉｉ）特定の方向に適用された隣接サンプルを有するブロックの母集団（「角度イントラ予測」）、または（ｉｖ）隣接サンプルおよび選択された行列係数を使用した行列乗算の結果であり得る。予測されたブロックと対応する入力サンプルとの間のさらなる不一致は、「残差」をビットストリームに符号化することによって、ある程度補正され得る。残差は一般に、空間領域から周波数領域に変換されて、「一次変換領域において残差係数を形成し、これは、「二次変換領域」において残差係数を生成するために、「二次変換」のアプリケーションによってさらに変換され得る。残差係数は量子化パラメータに従って量子化され、デコーダで生成されたサンプルの再構成の精度の損失をもたらすが、ビットストリームにおけるビットレートの低減を伴う。ピクチャのシーケンスは、イントラ予測を使用するピクチャおよびイントラ予測またはインター予測を使用するピクチャの指定された構造に従って符号化され得、符号化順序において先行するピクチャに対する指定された依存性は、表示順序または配信順序とは異なり得る。「ランダムアクセス」構成は、周期的なイントラピクチャをもたらし、デコーダおよびがビットストリームの復号を開始するエントリポイン

トを形成する。ランダムアクセス構成における他のピクチャは一般に、インター予測を使用して、指定された深度の階層構造に従って、表示または配信順序で現在のピクチャの前後のピクチャからそれらのコンテンツを予測する。現在のピクチャを予測するために、表示順において現在のピクチャの後のピクチャの使用は、所与のピクチャの復号と、所与のピクチャの表示（およびバッファからの除去）との間のある程度のピクチャバッファリングおよび遅延が必要である。

【発明の概要】

【００１５】

本発明の目的は、既存の構成の１つまたは複数の欠点を実質的に克服するか、または少なくとも改善することである。

【００１６】

本開示の態様は、ビットストリームからテンソルを復号する方法であって、ビットストリームから情報の第１ユニットを復号することであって、情報の第１ユニットは、テンソルのチャネルにわたる平均テンソル値に対応する、復号することと、ビットストリームから情報の第２ユニットを復号することであって、情報の第２ユニットは、テンソルの分解の成分に対応する、復号することと、ビットストリームから情報の第３ユニットを復号することであって、情報の第３ユニットは、テンソルの情報の第１ユニットに関連する係数に対応する、復号することと、ビットストリームから情報の第４ユニットを復号することであって、情報の第４ユニットは、テンソルの情報の第２ユニットに関連する係数に対応する、復号することと、それぞれのチャネルについての係数に、平均テンソル値と、テンソルの分解に関連する係数を有するテンソルの分解の成分と、を乗算することによって、チャネルごとにテンソルを生成することと、を含み、ビットストリームからの情報の第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される、方法を提供する。

【００１７】

本開示の別の態様は、テンソルをビットストリームに符号化する方法であって、情報の第１ユニットを符号化することであって、情報の第１ユニットは、テンソルのチャネルにわたる平均テンソル値に対応する、符号化することと、情報の第２ユニットを符号化することであって、情報の第２ユニットは、テンソルの分解の成分に対応する、符号化することと、情報の第３ユニットを符号化することであって、情報の第３ユニットは、テンソルの情報の第１ユニットに関する係数に対応する、符号化することと、情報の第４ユニットを符号化することであって、情報の第４ユニットは、テンソルの情報の第２ユニットに関する係数に対応する、符号化することと、を含み、情報の第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される、方法を提供する。

【００１８】

本開示の別の態様は、ビットストリームからテンソルを復号する方法を実行するためのプログラムを格納する非一時的なコンピュータ可読格納媒体であって、方法は、

ビットストリームから情報の第１ユニットを復号することであって、情報の第１ユニットは、テンソルのチャネルにわたる平均テンソル値に対応する、復号することと、

ビットストリームから情報の第２ユニットを復号することであって、情報の第２ユニットは、テンソルの分解の成分に対応する、復号することと、

ビットストリームから情報の第３ユニットを復号することであって、情報の第３ユニットは、テンソルの情報の第１ユニットに関連する係数に対応する、復号することと、

ビットストリームから情報の第４ユニットを復号することであって、情報の第４ユニットは、テンソルの情報の第２ユニットに関連する係数に対応する、復号することと、

それぞれのチャネルについての係数に、平均テンソル値と、テンソルの分解に関連する係数を有するテンソルの分解の成分と、を乗算することによって、チャネルごとにテンソ

ルを生成することと、を含み、ビットストリームからの情報の第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される、非一時的なコンピュータ可読格納媒体を提供する。

【００１９】

本開示の別の態様は、ビットストリームからテンソルを復号するデコーダであって、デコーダは、ビットストリームから情報の第１ユニットを復号し、情報の第１ユニットは、テンソルのチャネルにわたる平均テンソル値に対応し、ビットストリームから情報の第２ユニットを復号し、情報の第２ユニットは、テンソルの分解の成分に対応し、ビットストリームから情報の第３ユニットを復号し、情報の第３ユニットは、テンソルの情報の第１ユニットに関連する係数に対応し、ビットストリームから情報の第４ユニットを復号し、情報の第４ユニットは、テンソルの情報の第２ユニットに関連する係数に対応し、それぞれのチャネルについての係数に、平均テンソル値と、テンソルの分解に関連する係数を有するテンソルの分解の成分と、を乗算することによって、チャネルごとにテンソルを生成する、ように構成され、ビットストリームからの情報の第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される、デコーダを提供する。

【００２０】

本開示の別の態様は、テンソルをビットストリームに符号化する方法を実行するためのプログラムを格納する非一時的なコンピュータ可読格納媒体であって、方法は、情報の第１ユニットを符号化することであって、情報の第１ユニットは、テンソルのチャネルにわたる平均テンソル値に対応する、符号化することと、情報の第２ユニットを符号化することであって、情報の第２ユニットは、テンソルの分解の成分に対応する、符号化することと、情報の第３ユニットを符号化することであって、情報の第３ユニットは、テンソルの情報の第１ユニットに関する係数に対応する、符号化することと、情報の第４ユニットを符号化することであって、情報の第４ユニットは、テンソルの情報の第２ユニットに関する係数に対応する、符号化することと、を含み、情報の第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される、非一時的なコンピュータ可読格納媒体を提供する。

【００２１】

本開示の別の態様は、テンソルをビットストリームに符号化するエンコーダであって、エンコーダは、情報の第１ユニットを符号化し、情報の第１ユニットは、テンソルのチャネルにわたる平均テンソル値に対応し、情報の第２ユニットを符号化し、情報の第２ユニットは、テンソルの分解の成分に対応し、情報の第３ユニットを符号化し、情報の第３ユニットは、テンソルの情報の第１ユニットに関する係数に対応し、情報の第４ユニットを符号化し、情報の第４ユニットは、テンソルの情報の第２ユニットに関する係数に対応する、ように構成され、情報の第１、第２、第３、および第４ユニットのうちの少なくとも１つは、情報の他のユニットのうちの少なくとも１つに関して独立して復号可能であり、情報の各ユニットは、サンプルの複数の２次元配列に配置される、エンコーダを提供する。

【００２２】

他の態様も開示される。

【図面の簡単な説明】

【００２３】

次に、本発明の少なくとも１つの実施形態を、以下の図面および付録を参照して説明する。

【図１】図１は、分散マシンタスクシステムを示す概略ブロック図である。

【図２Ａ】図２Ａは、図１の分散マシンタスクシステムが実施され得る汎用コンピュータ

システムの概略ブロック図を形成する。
【図２Ｂ】図２Ｂは、図１の分散マシンタスクシステムが実施され得る汎用コンピュータシステムの概略ブロック図を形成する。
【図３Ａ】図３Ａは、ＣＮＮのバックボーン部分の機能モジュールを示す概略ブロック図である。
【図３Ｂ】図３Ｂは、図３Ａの残差ブロックを示す概略ブロック図である。
【図３Ｃ】図３Ｃは、図３Ａの残差ユニットを示す概略ブロック図である。
【図３Ｄ】図３Ｄは、図３ＡのＣＢＬモジュールを示す概略ブロック図である。
【図４】図４は、ＣＮＮの代替バックボーン部分の機能モジュールを示す概略ブロック図である。
【図５】図５は、分散マシンタスクシステムの一部としてのチャネル間非相関ベースのテンソルエンコーダを示す概略ブロック図である。
【図６】図６は、チャネル間非相関ベースのテンソルエンコーダの一部として、特徴マップパッカ、パックフレームエンコーダ、およびアンパッカを示す概略ブロック図である。
【図７】図７は、ビデオエンコーダの機能モジュールを示す概略ブロック図である。
【図８】図８は、分散マシンタスクシステムの一部としてのチャネル間非相関ベースのテンソルデコーダを示す概略ブロック図である。
【図９】図９は、ビデオデコーダの機能モジュールを示す概略ブロック図である。
【図１０Ａ】図１０Ａは、ＣＮＮのヘッド部を示す概略ブロック図である。
【図１０Ｂ】図１０Ｂは、図１０Ａのアップスケーラモジュールを示す概略ブロック図である。
【図１０Ｃ】図１０Ｃは、図１０Ａの検出モジュールを示す概略ブロック図である。
【図１１】図１１は、ＣＮＮの代替ヘッド部を示す概略ブロック図である。
【図１２Ａ】図１２Ａは、チャネル間非相関テンソルデータを保持するためのサブピクチャの配置を示す概略ブロック図である。
【図１２Ｂ】図１２Ｂは、チャネル間非相関テンソルデータを保持するためのサブピクチャの配置を示す概略ブロック図である。
【図１３】図１３は、符号化されたチャネル間非相関特徴マップおよび関連するメタデータを保持するビットストリームを示す概略ブロック図である。
【図１４】図１４は、ＣＮＮの第１部分を実行し、非相関化し、結果として得られる特徴マップを符号化するための方法を示す。
【図１５】図１５は、ビットストリームを復号し、非相関特徴マップを再構成し、ＣＮＮの第２部分を実行するための方法を示す。
【図１６Ａ】図１６Ａは、２つの連続するデータセットおよび関連する固有ベクトルを示す。
【図１６Ｂ】図１６Ｂは、２つの連続するデータセットおよび関連する固有ベクトルを示す。
【図１７】図１７は、ＣＮＮの第１部分から生成されたテンソルに対する連続する非相関演算から生じる基底ベクトルを調整するための方法を示す。
【図１８Ａ】図１８Ａは、分解を用いて圧縮する前にＦＰＮ内の異なるレイヤからのテンソルを結合するためのテンソル低減モジュールを示す概略ブロック図である。
【図１８Ｂ】図１８Ｂは、解凍後に結合テンソルから異なるＦＰＮレイヤのテンソルを抽出するためのテンソル分離モジュールを示す概略ブロック図である。
【図１９】図１９は、ＣＮＮのマルチレイヤ第１部分からの別のテンソルとの非相関のためにＣＮＮのマルチレイヤ第１部分からテンソルを再サンプリングする方法を示す。
【図２０】図２０は、ＣＮＮの第２部分について、ビットストリームから再構成されたテンソルを元の解像度に再サンプリングするための方法を示す。
【図２１】図２１は、ＣＮＮの第１部分からテンソルを符号化するための基底ベクトルを決定するための方法を示す。
【図２２】図２２は、ＣＮＮの第１部分からテンソルを符号化するために使用される基底

ベクトルを復号する方法を示す。
【図２３】図２３は、ネットワーク内の分割点における異なるレイヤのテンソルと、テンソルコンバイナから符号化テンソルを再生するための結果として生じるオプションと、の間の関係を示す概略ブロック図である。
【００２４】
付録Ａは、ビットストリームにおける基底ベクトルパッキングに関連するメタデータを表すためのＳＥＩメッセージフォーマットおよび関連するセマンティクスを示す。
【発明を実施するための形態】
【００２５】
添付の図面のいずれか１つまたは複数において、同じ参照符号を有するステップおよび／または特徴が参照される場合、それらのステップおよび／または特徴は反対の意図がない限り、本説明の目的のために、同じ機能または動作を有する。
【００２６】
分散マシンタスクシステムは、中間圧縮データを生成するネットワークカメラまたはスマートフォンなどのエッジデバイスを含むことができる。分散マシンタスクシステムはまた、タスク結果を生成するために中間圧縮データについて動作する、サーバファームベース（「クラウド」）アプリケーションなどの最終デバイスを含み得る。タスク結果の例には、オブジェクト検出およびインスタンスセグメンテーションタスクのパフォーマンスを示すために使用されるｍＡＰ（平均精度）、および複数のビデオフレームにわたってオブジェクトを追跡する精度を測定するために使用されるＭＯＴＡ（複数オブジェクト追跡精度）が含まれる。他のタスク結果の例は、人間の姿勢推定および行動認識を含む。加えて、エッジデバイス機能は、クラウドにおいて実施されてもよく、中間圧縮データは潜在的に、必要に応じて複数の異なるタスクのために、後の処理のために記憶されてもよい。
【００２７】
中間圧縮データの便利な形態は、高性能圧縮規格およびその実装の利用可能性のために、圧縮ビデオビットストリームである。ビデオ圧縮規格は、典型的には平面アレイに配置された、１０ビットなど、いくつかの所与のビット深度の整数サンプルに対して動作する。カラービデオは、用途に応じて、例えば色成分Ｙ、Ｃｂ、Ｃｒ、またはＲ、Ｇ、Ｂに対応する３つの平面アレイを有する。ＣＮＮは、典型的にはテンソルの形成で浮動小数点データに対して動作する。テンソルは一般に、ＣＮＮが動作する入力ビデオデータと比較して、比較的小さい空間次元数を有する。テンソルは一般に、カラービデオデータに典型的な３つのチャネルよりも多くのチャネルを有する。
【００２８】
テンソルは典型的には、以下の寸法：フレーム、チャネル、高さ、および幅を有する。例えば、寸法［１、２５６、７６、１３６］のテンソルは、各々がサイズ１３６×７６の２５６個の特徴マップ（チャネル）を含むと言われる。ビデオデータの場合、推論は、典型的には複数のフレームを含むテンソルを使用するのではなく、一度に１フレーム（フレーム値１）で実行される。
【００２９】
ＶＶＣは、ピクチャの複数のサブピクチャへの分割をサポートし、その各々は、独立して符号化され、独立して復号され得る。１つのアプローチでは、各サブピクチャが１つの「スライス」として、または符号化されたＣＴＵの連続するシーケンスとして符号化される。「タイル」メカニズムもまた、ピクチャをいくつかの独立して復号可能な領域に分割するために利用可能である。サブピクチャは、それぞれのサブピクチャとして符号化されたＣＴＵの様々な矩形セットを用いて、多少柔軟な方法で指定され得る。サブピクチャ次元の柔軟な定義は、１つのピクチャ内の異なるエリアを必要とするタイプのデータを効率的に保持することを可能にし、大きな「未使用」エリア、すなわち、テンソルデータの再構成のために使用されないフレームのエリアを回避する。
【００３０】
図１は、分散マシンタスクシステム１００の機能モジュールを示す概略ブロック図であ

る。複数のシステムにわたってマシンタスクを分配するという概念は、「協調インテリジェンス（ｃｏｌｌａｂｏｒａｔｉｖｅ　ｉｎｔｅｌｌｉｇｅｎｃｅ）」（ＣＩ）と呼ばれることがある。特定のニューラルネットワークを２つの部分に分割するには、ネットワーク内の「分割点」を指定する必要がある。入力レイヤから分割点までのネットワーク内のレイヤは、第１デバイスにおいて実行され、結果として得られる中間テンソル（複数可）が圧縮される。分割点からネットワークの終わりまでのレイヤは、第１デバイスからの解凍テンソルを使用して実行される。分割点において、圧縮される必要がある１つ以上のテンソルが存在し得る。テンソルの次元は同じであってもよいし、異なっていてもよい。「フィーチャピラミッドネットワーク（ｆｅａｔｕｒｅ　ｐｙｒａｍｉｄ　ｎｅｔｗｏｒｋ）」（ＦＰＮ）が使用されている場合、所与のレイヤが前のレイヤの幅および高さの半分であるように、レイヤが幅および高さに関連することが一般的である。ＦＰＮアーキテクチャはまた、すべての交互レイヤ上で生じる幅および高さの半減を定義し得る。いくつかのアーキテクチャでは、同じ幅および高さの複数のテンソルが見られる。レイヤ間のチャネルカウントは、変化してもよく、同じであってもよい。したがって、現代のＣＮＮにおいて使用される様々なネットワークトポロジに適用可能な圧縮方法は、広範囲のシナリオへの適用に有益である。

【００３１】

システム１００は、特徴マップを非相関化し、パッキングし、量子化して、符号化データから特徴マップを符号化し、復号するための平面フレームにするための方法を実施するために使用され得る。システム１００は、関連するオーバーヘッドデータの計算負荷が許容可能であり、復号された特徴マップについてのタスク性能がビットストリームのビットレートの変化に耐性があるように使用することができる。システム１００はまた、テンソルの量子化表現がビットを不必要に消費しないように使用され得、ビットは、タスク性能に関して相応の利益を提供しない。

【００３２】

システム１００は、符号化ビデオビットストリーム１２１の形成でＣＮＮバックボーン１１４から符号化テンソルデータ１１５ａを生成するためのソースデバイス１１０を含む。システム１００はまた、符号化ビデオビットストリーム１４３の形成でテンソルデータを復号するための宛先デバイス１４０を含む。通信チャネル１３０は、ソースデバイス１１０から宛先デバイス１４０に符号化ビデオビットストリーム１２１を通信するために使用される。いくつかの構成では、ソースデバイス１１０および宛先デバイス１４０がそれぞれの移動電話ハンドセット（たとえば、「スマートフォン」）またはネットワークカメラおよびクラウドアプリケーションのいずれかまたは両方を備え得る。通信チャネル１３０は、イーサネット（登録商標）などの有線接続、またはワイドエリアネットワーク（ＷＡＮ）を介する接続、またはアドホック接続を介する接続を含む、ＷｉＦｉまたは５Ｇなどのワイヤレス接続であり得る。さらに、ソースデバイス１１０および宛先デバイス１４０は、符号化ビデオデータがファイルサーバまたはメモリ中のハードディスクドライブなど、いくつかのコンピュータ可読記憶媒体上でキャプチャされるアプリケーションを備え得る。

【００３３】

図１に示すように、ソースデバイス１１０は、ビデオソース１１２と、ＣＮＮバックボーン１１４と、テンソルコンバイナ１６２と、主成分分析（ＰＣＡ）エンコーダ１６０と、送信機１２２とを含む。ビデオソース１１２は、典型的には画像キャプチャセンサ、非一時的記録媒体上に記憶された以前にキャプチャされたビデオシーケンス、または遠隔画像キャプチャセンサからのビデオフィードなど、キャプチャされたビデオフレームデータ（１１３として示される）のソースを備える。ビデオソース１１２はまた、コンピュータグラフィックスカードの出力であってもよく、例えば、オペレーティングシステムのビデオ出力、およびコンピューティングデバイス（例えば、タブレットコンピュータ）上で実行される様々なアプリケーションを表示する。ビデオソース１１２として画像キャプチャセンサを含み得るソースデバイス１１０の例は、スマートフォン、ビデオカムコーダ、プ

ロフェッショナルビデオカメラ、およびネットワークビデオカメラを含む。ビデオソース１１２は、独立した画像を生成してもよく、または時間的に連続した画像、すなわちビデオを生成してもよい。

【００３４】

ＣＮＮバックボーン１１４は、ビデオフレームデータ１１３を受信し、ＣＮＮの「バックボーン」に対応するレイヤなど、全体的なＣＮＮの特定のレイヤを実行し、テンソル１１５ａを出力する。ＣＮＮのバックボーンレイヤは例えば、ビデオフレームデータ１１３によって表される入力画像の異なる空間スケールに対応する複数のテンソルを出力として生成することができる。異なる空間スケールは、「特徴ピラミッドネットワーク(feature pyramid network)」（ＦＰＮ）アーキテクチャと呼ばれ得る。ＦＰＮは例えば、システム１００によって「ＹＯＬＯｖ３」ネットワーク（オブジェクト検出に使用される）が実行され、空間分解能およびチャネルカウントが変化する場合、テンソル１１５ａとしてバックボーン１１４から出力される３つのレイヤに対応する３つのテンソルをもたらすことができる。システム１００が「高速ＲＣＮＮ　Ｘ１０１－ＦＰＮ」（オブジェクト検出のために使用される）または「マスクＲＣＮＮ　Ｘ１０１－ＦＰＮ」（インスタンスセグメンテーションのために使用される）などのネットワークを実行しているとき、テンソル１１５ａは、５つのレイヤＰ２～Ｐ５のテンソルを含む。テンソルコンバイナ１６２は、１つのレイヤについてテンソルを再サンプリングし、得られたテンソルを別のレイヤ、たとえば、ＦＰＮ中の隣接するレイヤのテンソルと連結し、結合テンソル１１５を生成することによって、２つのレイヤを結合し得る。ＰＣＡエンコーダ１６０は、テンソルコンバイナ１６２から出力されるテンソル１１５を受信する。レイヤが個別に非相関化されるレイヤである場合に必要となる基底ベクトルよりも少ない基底ベクトルを使用して表される結合レイヤについて、充分なインターレイヤ相関（inter-layer correlation）が存在する場合に、レイヤの結合は適している。インターレイヤ相関の程度は、ネットワーク自体および提供される入力データの特性である。インターレイヤ相関の程度は、各レイヤのテンソルが別々に個別に非相関化される場合の基底ベクトルの数の合計と比較して、結合テンソルの基底ベクトルの総数の低減を可能にする。たとえば、通常、レイヤごとに２５個の基底ベクトルを使用し、２つのレイヤを連結する場合、連結されたレイヤによって必要とされる基底ベクトルの数は、５０未満に設定され得る。使用される基底ベクトルの数が、基底ベクトルの説明された分散（the explained variance）に適応的である構成は、そのようなインターレイヤ相関を利用することによって基底ベクトルカウントの低減を達成することができ、図２１および図２２を参照して論じられる。インターレイヤ相関が入力データによって影響を受ける限り、テンソルコンバイナ１６２は、基底ベクトルエリアが最小化されるようにレイヤを結合するかどうかを選択することができる。ネットワークがある程度のインターレイヤ相関を有すると理解される場合、固定された構成が適用されてもよく、例えば、ＦＰＮの最も分解されていないレイヤをダウンサンプリングし、その結果を、非相関ステップ（すなわち、１４８０）においてＦＰＮの第２の最も分解されていないレイヤと組み合わせる。レイヤの結合が適切または必要でない場合、テンソルコンバイナ１６２はいかなる動作も実行せず、テンソル１１５はテンソル１１５ａに等しい。

【００３５】

ＰＣＡエンコーダ１６０は、ＣＮＮバックボーン１１４によって出力される全体的なＣＮＮの１つまたは複数の内部レイヤを符号化するように動作する。ＰＣＡエンコーダ１６０は、まず、主成分分析（ＰＣＡ）などの方法を使用して、それぞれのテンソルのチャネルにわたって、テンソルのＤＣ正規化バージョンを非相関化する。テンソルのＤＣ正規化バージョンは、テンソルの各チャネルから「平均特徴」を除去することによって生成される。平均特徴は、テンソルのすべてのチャネルから均一に除去（減算）されてもよく、または平均特徴は、順変換の適用によってチャネル依存方式で除去されてもよく、その結果、テンソルの各チャネルについて対応する「平均係数」が得られる。ＰＣＡから得られる固有ベクトルのサブセットは、符号化されるテンソルのチャネルを表すために「基底ベク

トル」（または交換可能に、「構成要素」という用語が基底ベクトルを指すために使用され得る）として使用される。テンソルの各チャネルは、係数を生成するために基底ベクトルを使用して変換される。基底ベクトル、平均特徴マップ、および関連する係数は、量子化され、ビデオフレームにパックされる。ビデオフレームは、ビデオエンコーダを使用して符号化され、ビットストリーム１２１として出力される。ビットストリーム１２１は通信チャネル１３０を介した送信のために送信機１２２に供給されるか、またはビットストリーム１２１は後で使用するために記憶装置１３２に書き込まれる。

【００３６】

ソースデバイス１１０は、ＣＮＮバックボーン１１４のための特定のネットワークをサポートする。しかしながら、宛先デバイス１４０は、ヘッドＣＮＮ１５０のためのいくつかのネットワークのうちの１つを使用し得る。ヘッドＣＮＮ１５０のためのいくつかのネットワークのうちの１つを使用する際に、パックされた特徴マップの形態の部分的に処理されたデータは、ＣＮＮバックボーン１１４の動作を再度実行する必要なしに、様々なタスクを実行する際に後で使用するために記憶され得る。

【００３７】

ビットストリーム１２１は、送信機１２２によって、符号化ビデオデータ（または「符号化ビデオ情報」）として通信チャネル１３０を介して送信される。ビットストリーム１２１は、いくつかの実装形態では記憶メモリ１３２に記憶することができ、記憶装置１３２は、後に通信チャネル１３０を介して（または通信チャネル１３０を介した送信の代わりに）送信されるまで、「フラッシュ」メモリまたはハードディスクドライブなどの非一時的記憶デバイスである。例えば、符号化ビデオデータは、ビデオ分析アプリケーションのためのワイドエリアネットワーク（ＷＡＮ）を介して顧客に要求に応じて供給され得る。

【００３８】

宛先デバイス１４０は、受信機１４２と、ＰＣＡデコーダ１７０と、テンソルセパレータ１７２と、ＣＮＮヘッド１５０と、ＣＮＮタスク結果バッファ１５２と、を含む。受信機１４２は、通信チャネル１３０から符号化ビデオデータを受信し、ビデオビットストリーム１４３をＰＣＡデコーダ１７０に渡す。ＰＣＡデコーダ１７０は、復号されたテンソル１４９ａを出力し、テンソルセパレータ１７２に供給される。テンソルセパレータは、テンソルコンバイナ１６２の逆演算を実行して、抽出されたテンソル１４９を生成する。抽出されたテンソル１４９は、ＣＮＮヘッド１５０に渡される。レイヤの結合が適切でないか、または必要とされないアーキテクチャでは、テンソルセパレータ１７２がいかなる動作も実行せず、テンソル１４９はテンソル１４９ａに等しい。ＣＮＮヘッド１５０は、ＣＮＮバックボーン１１４で開始されたタスクの後のレイヤを実行して、タスク結果１５１を生成する。タスク結果１５１は、タスク結果バッファ１５２に格納される。タスク結果バッファ１５２の内容は例えば、グラフィカルユーザインターフェースを介してユーザに提示されてもよく、またはいくつかのアクションがタスク結果に基づいて決定される分析アプリケーションに提供されてもよく、タスク結果は集約されたタスク結果の要約レベルのユーザへの提示を含み得る。ソースデバイス１１０および宛先デバイス１４０の各々の機能性は、単一のデバイスにおいて実施されることも可能であり、その例は、携帯電話ハンドセット、タブレットコンピュータ、およびクラウドアプリケーションを含む。

【００３９】

上述の例示的なデバイスにもかかわらず、ソースデバイス１１０および宛先デバイス１４０の各々は、一般にハードウェア構成要素とソフトウェア構成要素との組合せを通して、汎用コンピューティングシステム内で構成され得る。図２Ａは、コンピュータモジュール２０１、キーボード２０２とマウスポインタデバイス２０３とスキャナ２２６と、ビデオソース１１２として構成され得るカメラ２２７と、マイクロフォン２８０となどの入力デバイスと、プリンタ２１５、表示デバイス２１４、およびラウドスピーカ２１７を含む出力デバイスと、を含む、そのようなコンピュータシステム２００を示す。外部変調器－復調器（Ｍｏｄｅｍ）トランシーバデバイス２１６は、コネクション２２１を介して通信

ネットワーク２２０との間で通信するためにコンピュータモジュール２０１によって使用され得る。通信チャネル１３０を表し得る通信ネットワーク２２０は、インターネット、セルラー電気通信ネットワーク、またはプライベートＷＡＮなどの（ＷＡＮ）であり得る。コネクション２２１が電話回線である場合、モデム２１６は、従来の「ダイヤルアップ」モデムであってもよい。代替として、コネクション２２１が大容量（たとえば、ケーブルまたは光）コネクションである場合、モデム２１６はブロードバンドモデムであり得る。ワイヤレスモデムはまた、通信ネットワーク２２０へのワイヤレス接続のために使用され得る。トランシーバデバイス２１６は、送信機１２２および受信機１４２の機能を提供することができ、通信チャネル１３０は、コネクション２２１において実施され得る。

【００４０】

コンピュータモジュール２０１は、典型的には少なくとも１つのプロセッサユニット２０５と、メモリユニット２０６とを含む。例えば、メモリユニット２０６は、半導体ＲＡＭ（ｒａｎｄｏｍ ａｃｃｅｓｓ ｍｅｍｏｒｙ）及び半導体ＲＯＭ（ｒｅａｄ ｏｎｌｙ ｍｅｍｏｒｙ）を有することができる。コンピュータモジュール２０１はまた、ビデオディスプレイ２１４に結合するオーディオビデオインターフェース２０７、ラウドスピーカ２１７およびマイクロフォン２８０と、キーボード２０２、マウス２０３、スキャナ２２６、カメラ２２７、およびオプションでジョイスティックまたは他のヒューマンインターフェースデバイス（図示せず）に結合するＩ／Ｏインターフェース２１３と、外部モデム２１６およびプリンタ２１５のためのインターフェース２０８とを含む、いくつかの入力／出力（Ｉ／Ｏ）インターフェースを含む。オーディオ－ビデオインターフェース２０７からコンピュータモニタ２１４への信号は一般に、コンピュータグラフィックスカードの出力である。いくつかの実装形態では、モデム２１６がコンピュータモジュール２０１内に、たとえばインターフェース２０８内に組み込まれ得る。コンピュータモジュール２０１はまた、ローカルエリアネットワーク（ＬＡＮ）として知られるローカルエリア通信ネットワーク２２２へのコネクション２２３を介してコンピュータシステム２００の結合を可能にするローカルネットワークインターフェース２１１を有する。図２Ａに示されるように、ローカル通信ネットワーク２２２はまた、コネクション２２４を介してワイドネットワーク２２０に結合することができ、これは、典型的にはいわゆる「ファイアウォール」デバイスまたは同様の機能のデバイスを含む。ローカルネットワークインターフェース２１１は、Ｅｔｈｅｒｎｅｔ™回路カード、Ｂｌｕｅｔｏｏｔｈ™ワイヤレス構成、またはＩＥＥＥ８０２．１１ワイヤレス構成を有しうるが、インターフェース２１１のために多数の他のタイプのインターフェースが実施され得る。ローカルネットワークインターフェース２１１はまた、送信機１２２および受信機１４２の機能を提供することができ、通信チャネル１３０はまた、ローカル通信ネットワーク２２２において実施され得る。

【００４１】

Ｉ／Ｏインターフェース２０８および２１３は、シリアルコネクティビティおよびパラレルコネクティビティのいずれかまたは両方を提供することができ、前者は典型的にはユニバーサルシリアルバス（ＵＳＢ）規格に従って実装され、対応するＵＳＢコネクタ（図示せず）を有する。記憶デバイス２０９が提供され、典型的にはハードディスクドライブ（ＨＤＤ）２１０を含む。また、フロッピーディスク（登録商標）ドライブや磁気テープドライブ（図示せず）などの他の記憶デバイスを用いてもよい。光ディスクドライブ２１２は、典型的にはデータの不揮発性ソースとして機能するように設けられる。例えば、光ディスク（例えば、ＣＤ－ＲＯＭ、ＤＶＤ、Ｂｌｕ ｒａｙ Ｄｉｓｃ™）、ＵＳＢ－ＲＡＭ、ポータブル、外部ハードドライブ、およびフロッピーディスクなどのポータブルメモリデバイスを、コンピュータシステム２００への適切なデータのソースとして使用することができる。典型的には、ＨＤＤ２１０、光ドライブ２１２、ネットワーク２２０および２２２のいずれも、ビデオソース１１２として、またはディスプレイ２１４を介して再生のために記憶される復号されたビデオデータの宛先として動作するように構成されてもよい。システム１００のソースデバイス１１０および宛先デバイス１４０は、コンピュ

ータシステム２００において実施され得る。
【００４２】
コンピュータモジュール２０１の構成要素２０５～２１３は、典型的には、相互接続されたバス２０４を介して、当業者に知られているコンピュータシステム２００の従来の動作モードをもたらすように通信する。たとえば、プロセッサ２０５は、コネクション２１８を使用してシステムバス２０４に結合される。同様に、メモリ２０６および光ディスクドライブ２１２は、コネクション２１９によってシステムバス２０４に結合される。説明された構成を実施することができるコンピュータの例は、ＩＢＭ－ＰＣおよび互換機、ＳｕｎＳＰＡＲＣｓｔａｔｉｏｎｓ、ＡｐｐｌｅＭａｃ™、または同様のコンピュータシステムを含む。
【００４３】
適切または望ましい場合、ＰＣＡエンコーダ１６０およびＰＣＡデコーダ１７０、ならびに以下で説明する方法は、コンピュータシステム２００を使用して実装され得る。特に、ＰＣＡエンコーダ１６０、ＰＣＡデコーダ１７０、および説明される方法は、コンピュータシステム２００内で実行可能な１つまたは複数のソフトウェアアプリケーションプログラム２３３として実装され得る。特に、ＰＣＡエンコーダ１６０、ＰＣＡデコーダ１７０、および説明される方法のステップは、コンピュータシステム２００内で実行されるソフトウェア２３３内の命令２３１（図２Ｂ参照）によって実行される。ソフトウェア命令２３１は、それぞれが１つまたは複数の特定のタスクを実行するための１つまたは複数のコードモジュールとして形成され得る。ソフトウェアはまた、２つの別個の部分に分割され得、第１部分および対応するコードモジュールは説明された方法を実行し、第２部分および対応するコードモジュールは、第１部分とユーザとの間のユーザインターフェースを管理する。
【００４４】
ソフトウェアはたとえば、以下で説明する記憶デバイスを含むコンピュータ可読媒体に記憶され得る。ソフトウェアは、コンピュータ可読媒体からコンピュータシステム２００にロードされ、次いで、コンピュータシステム２００によって実行される。コンピュータ可読媒体に記録されたそのようなコンピュータプログラムまたはソフトウェアを有するコンピュータ可読媒体は、コンピュータプログラム製品である。コンピュータシステム２００におけるコンピュータプログラム製品の使用は、好ましくはソースデバイス１１０および宛先デバイス１４０を実装するための有利な装置と、説明された方法と、をもたらす。
【００４５】
ソフトウェア２３３は、典型的には、ＨＤＤ２１０またはメモリ２０６に格納される。ソフトウェアは、コンピュータ可読媒体からコンピュータシステム２００にロードされ、コンピュータシステム２００によって実行される。したがって、たとえば、ソフトウェア２３３は光ディスクドライブ２１２によって読み取られる光学的に読み取り可能なディスク記憶媒体（たとえば、ＣＤ－ＲＯＭ）２２５に記憶され得る。
【００４６】
いくつかの例では、アプリケーションプログラム２３３が１つまたは複数のＣＤ－ＲＯＭ２２５上で符号化されてユーザに供給され、対応するドライブ２１２を介して読み出されてもよく、あるいはネットワーク２２０または２２２からユーザによって読み出されてもよい。さらに、ソフトウェアは、他のコンピュータ可読媒体からコンピュータシステム２００にロードすることもできる。コンピュータ可読記憶媒体は実行および／または処理のために、記録された命令および／またはデータをコンピュータシステム２００に提供する任意の非一時的有形記憶媒体を指す。このような記憶媒体の例は、フロッピーディスク、磁気テープ、ＣＤ－ＲＯＭ、ＤＶＤ、Ｂｌｕ－ｒａｙ　Ｄｉｓｃ™、ハードディスクドライブ、ＲＯＭ又は集積回路、ＵＳＢメモリ、光磁気ディスク、又はＰＣＭＣＩＡカード等のコンピュータ可読カード、などを含み、このようなデバイスはコンピュータモジュール２０１の内部又は外付けである。コンピュータモジュール２０１へのソフトウェア、アプリケーションプログラム、命令、および／またはビデオデータもしくは符号化ビデオ

データの提供に参加することもできる一時的または非有形のコンピュータ可読伝送媒体の例は、無線または赤外線伝送チャネル、ならびに別のコンピュータまたはネットワーク化されたデバイスへのネットワークコネクション、ならびに電子メール送信およびウェブサイトなどに記録された情報を含むインターネットまたはイントラネットを含む。

【００４７】

アプリケーションプログラム２３３の第２部分および上述の対応するコードモジュールは、ディスプレイ２１４上にレンダリングされるかまたは表される１つまたは複数のグラフィカルユーザインターフェース（ＧＵＩ）を実施するために実行され得る。典型的にはキーボード２０２およびマウス２０３の操作を通して、コンピュータシステム２００およびアプリケーションのユーザは、機能的に適応可能な方法でインターフェースを操作して、ＧＵＩに関連するアプリケーションに制御コマンドおよび／または入力を提供することができる。ラウドスピーカ２１７を介して出力されるスピーチプロンプトと、マイクロフォン２８０を介して入力されるユーザ音声コマンドとを利用するオーディオインターフェースなど、機能的に適応可能なユーザインターフェースの他の形態も実装され得る。

【００４８】

図２Ｂは、プロセッサ２０５および「メモリ」２３４の詳細な概略ブロック図である。メモリ２３４は、図２Ａのコンピュータモジュール２０１によってアクセス可能な全てのメモリモジュール（記憶デバイス２０９及び半導体メモリ２０６を含む）の論理的集合を表す。

【００４９】

コンピュータモジュール２０１が最初に電源投入されると、パワーオンセルフテスト（ＰＯＳＴ）プログラム２５０が実行される。ＰＯＳＴプログラム２５０は、典型的には図２Ａの半導体メモリ２０６のＲＯＭ２４９に格納される。ソフトウェアを記憶するＲＯＭ２４９などのハードウェアデバイスは、ファームウェアと呼ばれることがある。ＰＯＳＴプログラム２５０は適切な機能を保証するためにコンピュータモジュール２０１内のハードウェアを検査し、典型的には、正しい動作のために、プロセッサ２０５、メモリ２３４（２０９、２０６）、および同様に典型的にはＲＯＭ２４９に記憶された基本入出力システムソフトウェア（ＢＩＯＳ）モジュール２５１をチェックする。ＰＯＳＴプログラム２５０が正常に実行されると、ＢＩＯＳ２５１は、図２Ａのハードディスクドライブ２１０を起動する。ハードディスクドライブ２１０の起動により、ハードディスクドライブ２１０に常駐するブートストラップローダプログラム２５２に、プロセッサ２０５を介して実行させる。これにより、オペレーティングシステム２５３がＲＡＭメモリ２０６にロードされ、オペレーティングシステム２５３が動作を開始する。オペレーティングシステム２５３はプロセッサ２０５によって実行可能なシステムレベルアプリケーションであり、プロセッサ管理、メモリ管理、デバイス管理、ストレージ管理、ソフトウェアアプリケーションインターフェース、および汎用ユーザインターフェースを含む、様々な高レベルの機能を果たす。

【００５０】

オペレーティングシステム２５３は、メモリ２３４（２０９、２０６）を管理して、コンピュータモジュール２０１上で実行される各プロセスまたはアプリケーションが別のプロセスに割り振られたメモリと衝突することなく実行するのに十分なメモリを有することを保証する。さらに、図２Ａのコンピュータシステム２００において利用可能な異なるタイプのメモリは、各プロセスが効果的に実行され得るように、適切に使用される必要がある。したがって、集約メモリ２３４は、メモリの特定のセグメントがどのように割り振られるかを示すことを意図するものではなく（特に明記しない限り）、むしろ、コンピュータシステム２００によってアクセス可能なメモリの一般的なビュー、およびそのようなメモリがどのように使用されるかを提供することを意図するものである。

【００５１】

図２Ｂに示されるように、プロセッサ２０５は、制御部２３９と、算術論理部（ＡＬＵ）２４０と、キャッシュメモリと呼ばれることもあるローカルまたは内部メモリ２４８と

を含む、いくつかの機能モジュールを含む。キャッシュメモリ２４８は、典型的にはレジスタセクション内にいくつかの記憶レジスタ２４４～２４６を含む。１つ以上の内部バス２４１は、これらの機能モジュールを機能的に相互接続する。プロセッサ２０５はまた、典型的には、コネクション２１８を使用して、システムバス２０４を介して外部デバイスと通信するための１つまたは複数のインターフェース２４２を有する。メモリ２３４は、コネクション２１９を使用してバス２０４に結合される。

【００５２】

アプリケーションプログラム２３３は、条件付き分岐およびループ命令を含むことができる一連の命令２３１を含む。また、プログラム２３３は、プログラム２３３の実行に用いられるデータ２３２を含んでもよい。命令２３１およびデータ２３２は、それぞれ、メモリロケーション２２８、２２９、２３０および２３５、２３６、２３７に記憶される。命令２３１およびメモリロケーション２２８～２３０の相対的なサイズに応じて、特定の命令はメモリロケーション２３０に示される命令によって示されるように、単一のメモリロケーションに記憶され得る。代替的に、命令はメモリロケーション２２８および２２９に示される命令セグメントによって示されるように、それぞれが別個のメモリロケーションに記憶されるいくつかの部分にセグメント化され得る。

【００５３】

一般に、プロセッサ２０５には、その中で実行される命令のセットが与えられる。プロセッサ２０５は、プロセッサ２０５が別の命令セットを実行することによって反応する後続の入力を待つ。各入力は、すべて図２Ａに示される、入力デバイス２０２、２０３のうちの１つまたは複数によって生成されたデータ、ネットワーク２２０、２０２のうちの１つにわたって外部ソースから受信されたデータ、記憶デバイス２０６、２０９のうちの１つから取り出されたデータ、または、対応するリーダ２１２に挿入された記憶媒体２２５から取り出されたデータを含む、いくつかのソースのうちの１つまたは複数から提供され得る。命令のセットの実行は、場合によってはデータの出力をもたらし得る。実行はまた、データまたは変数をメモリ２３４に記憶することを伴い得る。

【００５４】

ＰＣＡエンコーダ１６０、ＰＣＡデコーダ１７０、および説明する方法は、メモリ２３４中の対応するメモリロケーション２５５、２５６、２５７に記憶される、入力変数２５４を使用し得る。ＰＣＡエンコーダ１６０、ＰＣＡデコーダ１７０、および説明される方法は、メモリ２３４中の対応するメモリロケーション２６２、２６３、２６４に記憶される、出力変数２６１を生成する。中間変数２５８は、メモリロケーション２５９、２６０、２６６、および２６７に記憶され得る。

【００５５】

図２Ｂのプロセッサ２０５を参照すると、レジスタ２４４、２４５、２４６、算術論理ユニット（ＡＬＵ）２４０、および制御部２３９は、プログラム２３３を構成する命令セット内の命令ごとに「フェッチ、デコード、および実行」サイクルを実行するために必要なマイクロオペレーションのシーケンスを実行するように協働する。各フェッチ、デコード、および実行サイクルは：

メモリロケーション２２８、２２９、２３０から命令２３１をフェッチまたは読み出すフェッチ動作；

制御部２３９が、どの命令がフェッチされたかを判定するデコード動作；および

制御部２３９および／またはＡＬＵ２４０が命令を実行する実行動作と

を含む。

【００５６】

その後、次の命令のためのさらなるフェッチ、デコード、および実行サイクルが実行され得る。同様に、格納サイクルは、制御部２３９が値をメモリロケーション２３２に格納または書き込むことによって実行されてもよい。

【００５７】

説明される図１５～２２の方法における各ステップまたはサブプロセスは、プログラム

２３３の１つまたは複数のセグメントに関連付けられ、典型的にはプロセッサ２０５内のレジスタセクション２４４、２４５、２４６、ＡＬＵ２４０、および制御部２３９によって実行され、プログラム２３３の言及されたセグメントのための命令セット内のすべての命令について、フェッチ、デコード、および実行サイクルを実行するように協働する。

【００５８】

図３Ａは、ＣＮＮバックボーン１１４の実装として機能し得る、ＣＮＮのバックボーン部分３１０の機能モジュールを示す概略ブロック図３００である。バックボーン部分１１４は、「ＤａｒｋＮｅｔ－５３」と呼ばれることもあるが、異なるバックボーンも可能であり、その結果、フレームごとにテンソル１１５のレイヤの数および次元が異なる。

【００５９】

図３Ａに示すように、ビデオデータ１１３は、リサイザモジュール３０４に渡される。リサイザモジュール３０４は、ビデオデータ１１３の各フレームをＣＮＮバックボーン３１０による処理に適した解像度にリサイズし、リサイズされたフレームデータ３１２を生成する。ビデオデータ１１３の解像度が既にＣＮＮバックボーン３１０に適している場合、リサイザモジュール３０４の動作は必要とされない。リサイズされたフレームデータ３１２は、畳み込みバッチ正規化リーキー修正線形（ＣＢＬ）モジュール（ａ　ｃｏｎｖｏｌｕｔｉｏｎａｌ　ｂａｔｃｈ　ｎｏｒｍａｌｉｓａｔｉｏｎ　ｌｅａｋｙ　ｒｅｃｔｉｆｉｅｄ　ｌｉｎｅａｒ　（ＣＢＬ）　ｍｏｄｕｌｅ）３１４に渡され、テンソル３１６を生成する。ＣＢＬ３１４は、図３Ｄに示されるように、ＣＢＬモジュール３６０を参照して説明されるようなモジュールを含む。

【００６０】

ＣＢＬモジュール３６０は、入力として、リサイズされたフレームデータ３１２のテンソル３６１をとる。テンソル３６１は、畳み込みレイヤ３６２に渡されてテンソル３６３を生成する。畳み込みレイヤ３６２が１のストライドを有する場合、テンソル３６３はテンソル３６１と同じ空間ディメンジョンを有する。畳み込みレイヤ３６２が２などのより大きいストライドを有する場合、テンソル３６３はテンソル３６１と比較してより小さい空間ディメンジョンを有し、例えば、２のストライドの幅および高さが半分にされる。ストライドにかかわらず、テンソル３６３のチャネルディメンジョンのサイズは、特定のＣＢＬブロックのテンソル３６１のチャネルディメンジョンと比較して変化し得る。テンソル３６３は、テンソル３６５を出力するバッチ正規化モジュール３６４に渡される。バッチ正規化モジュール３６４は、入力テンソル３６３を正規化し、スケーリング係数およびオフセット値を適用して、出力テンソル３６５を生成する。スケーリング係数およびオフセット値は、トレーニングプロセスから導出される。テンソル３６５は、リーキー修正線形活性化（「ＬｅａｋｙＲｅＬＵ」）モジュール３６６に渡され、テンソル３６７を生成する。モジュール３６６は、「リーキー」活性化関数を提供し、テンソル内の正の値が通過され、負の値の大きさが、例えば、それらの以前の値の０．１Ｘに大幅に低減される。

【００６１】

テンソル３１６は、ＣＢＬブロック３１４から、内部的に１個の残差ユニット、２個の残差ユニット、および８個の残差ユニットの連結を含む１＋２＋８モジュールなどの残差ブロックモジュール３２０に渡される。

【００６２】

残差ブロックは、図３Ｂに示されるＲｅｓＢｌｏｃｋ３４０を参照して説明される。ＲｅｓＢｌｏｃｋ３４０は、テンソル３４１を受信する。テンソル３４１は、ゼロパディングモジュール３４２によってゼロパディングされ、テンソル３４３を生成する。テンソル３４３は、ＣＢＬモジュール３４４に渡されてテンソル３４５を生成する。テンソル３４５は、残差ユニット３４６に渡される。残差ユニット３４６は、一連の連結された残差ユニットを含む。残差ユニット３４６の最後の残差ユニットは、テンソル３４７を出力する。

【００６３】

図３Ｃに示されるように、残差ユニットは、ＲｅｓＵｎｉｔ３５０を参照して説明され

る。ＲｅｓＵｎｉｔ３５０は、テンソル３５１を入力とする。テンソル３５１は、ＣＢＬモジュール３５２に渡されてテンソル３５３を生成する。テンソル３５３は、第２ＣＢＬユニット３５４に渡され、テンソル３５５を生成する。加算モジュール３５６は、テンソル３５５をテンソル３５１と合計してテンソル３５７を生成する。加算モジュール３５６は、入力テンソル３５１が出力テンソル３５７に実質的に影響を与えるので、「ショートカット」と呼ばれることもある。トレーニングされていないネットワークの場合、ＲｅｓＵｎｉｔ３５０は、パススルーテンソルに作用する。トレーニングが実行されると、ＣＢＬモジュール３５２および３５４は、トレーニングデータおよびグラウンドトゥルースデータに従ってテンソル３５７をテンソル３５１から離すように動作する。

【００６４】

図３Ａに戻って、Ｒｅｓ１１モジュール３２０は、テンソル３２２を出力する。テンソル３２２は、バックボーンモジュール３１０からレイヤの１つとして出力され、Ｒｅｓ８モジュール３２４にも提供される。Ｒｅｓ８モジュール３２４は、８個の残差ユニット（すなわち、３５０）を含む残差ブロック（すなわち、３４０）である。Ｒｅｓ８モジュール３２４は、テンソル３２６を生成する。テンソル３２６は、Ｒｅｓ４モジュール３２８に渡され、レイヤの１つとしてバックボーンモジュール３１０から出力される。Ｒｅｓ４モジュールは４つの残差ユニット（すなわち、３５０）を含む残差ブロック（すなわち、３４０）である。Ｒｅｓ４モジュール３２８は、テンソル３２９を生成する。テンソル３２９は、バックボーンモジュール３１０からレイヤの１つとして出力される。まとめて、レイヤテンソル３２２、３２６、および３２９がテンソル１１５として出力される。バックボーンＣＮＮ３１０は入力として、解像度１０８８×６０８のビデオフレームをとり、３つのレイヤに対応する３つのテンソルを生成することができ、［１、２５６、７６、１３６］、［１、５１２、３８、６８］、［１、１０２４、１９、３４］の次元を有する。３つのレイヤに対応する３つのテンソル１１５の別の例は、［１、５１２、３４、１９］、［１、２５６、６８、３８］、［１、１２８、１３６、７６］であってもよく、これらはそれぞれ、ＣＮＮ３１０内の第７５の特徴マップ、第９０の特徴マップ、および第１０５の特徴マップで分割される。分離点は、ＣＮＮ３１０に依存する。

【００６５】

Ｒｅｓ１１　３２０、Ｒｅｓ８　３２４、およびＲｅｓ４　３２８の各々は、ＲｅｓＢｌｏｃｋ３４０と同様に動作する。ＣＢＬ３１４、ＣＢＬ３４４及びＣＢＬ３５４のそれぞれは、ＣＢＬ３６０と同様に動作する。

【００６６】

図４は、ＣＮＮバックボーン１１４の実装として機能し得る、ＣＮＮの代替バックボーン部分４００の機能モジュールを示す概略ブロック図である。バックボーン部分４００は、特徴ピラミッドネットワーク（「ResNet FPN」）を有する残差ネットワークを実装し、ＣＮＮバックボーン１１４の代替バージョンである。フレームデータ１１３は入力され、テンソル４０９、４１３、４１７、４２１、４２５を介して、ステムネットワーク４０８、ｒｅｓ２モジュール４１２、ｒｅｓ３モジュール４１６、ｒｅｓ４モジュール４２０、ｒｅｓ５モジュール４２４、および最大プールモジュール４２８を通過し、最大プールモジュール４２８は、出力としてＰ６テンソル４２９を生成する。

【００６７】

ステムネットワーク４０８は、２のストライドを有する７ｘ７畳み込みと、最大プーリング演算とを含む。ｒｅｓ２モジュール４１２、ｒｅｓ３モジュール４１６、ｒｅｓ４モジュール４２０、およびｒｅｓ５モジュール４２４は、ＬｅａｋｙＲｅＬＵアクティブ化などの畳み込み演算を実行する。各モジュール４１２、４１６、４２０および４２４は、また、２のストライド設定を介して、処理されたテンソルの解像度の１つの半分を実行する。テンソル４１３、４１７、４２１、および４２５の各々は、それぞれ、１×１の横方向畳み込みモジュール４４６、４４４、４４２、および４４０のうちの１つに渡される。モジュール４４６、４４４、４４２、および４４０は、それぞれテンソル４４７、４４５、４４３、４４１を生成する。テンソル４４１は、３ｘ３出力畳み込みモジュール４７０

に渡され、出力テンソルＰ５　４７１を生成する。テンソル４４１はまた、アップサンプリングされたテンソル４５１を生成するために、アップサンプラモジュール４５０に渡される。合計モジュール４６０はテンソル４４３および４５１を合計してテンソル４６１を生成し、これは、アップサンプラモジュール４５２および３ｘ３横方向畳み込みモジュール４７２に渡される。モジュール４７２は、Ｐ４テンソル４７３を出力する。アップサンプラモジュール４５２は、アップサンプリングされたテンソル４５３を生成する。合計モジュール４６２は、テンソル４４５および４５３を合計してテンソル４６３を生成し、これは、３ｘ３横方向畳み込みモジュール４７４およびアップサンプラモジュール４５４に渡される。モジュール４７４は、Ｐ３テンソル４７５を出力する。アップサンプラモジュール４５４は、アップサンプリングされたテンソル４５５を出力する。合計モジュール４６４は、テンソル４４７および４５５を合計してテンソル４６５を生成し、これは、３ｘ３横方向畳み込みモジュール４７６に渡される。モジュール４７６は、Ｐ２テンソル４７７を出力する。アップサンプラモジュール４５０、４５２、および４５４は、低い計算量のために最近傍補間を使用する。テンソル４２９、４７１、４７３、４７５、および４７７は、ＣＮＮバックボーン４００の出力テンソル１１５を形成する。図４は高速ＲＣＮＮネットワークアーキテクチャの特定のバックボーン部分（「Ｐレイヤスプリットポイント」）を示すが、バックボーンとヘッドへの異なる分割が可能である。テンソル４１３、４１７、４２１、および４２５の結果でネットワークを分割することは、「ｒｅｓ」分割点と呼ばれる。テンソル４０９でネットワークを分割することは、「ステム」分割点と呼ばれる。テンソル４４７、４４５、４４３、および４４１でネットワークを分割することは、「Ｃレイヤ」分割点と呼ばれる。

【００６８】

図５は、分散マシンタスクシステム１００の一部としてのエンコーダ１６０（または「ＰＣＡエンコーダ」）の実装形態に対応するチャネル間非相関ベースのテンソルエンコーダを示す概略ブロック図５００である。ＰＣＡエンコーダ１６０は、入力テンソルの再構成されたバージョンがＰＣＡデコーダ１７０によって最小限の忠実度の損失で生成され得るように、入力テンソル１１５の次元数を低減するように動作する。フレームデータ１１３と比較して、テンソル１１５はより低い空間分解能を有するが、より多くのチャネルを有する。たとえば、ＲＧＢまたはＹＣｂＣｒフォーマットのフレームデータ１１３は３つのチャネルを有し、テンソルは２５６個のチャネルを有し得る。結果として、テンソル１１５のチャネルをビデオフレームに直接パックすることは、符号化される必要がある比較的大きいサイズのフレームをもたらす。テンソルのチャネルは内容は様々であるが、ある程度のチャネル間相関を示す。フレームにパックされるとき、チャネル間相関はパッキングフォーマットが特徴マップ間の特定の空間オフセットをもたらすので、ビデオエンコーダを使用して活用することが困難であり、ビデオエンコーダは一般に、活用することができない。さらに、利用のためにチャネル間相関を明らかにするビデオエンコーダであっても、そうするためのインター予測またはイントラブロックコピーの使用は、別のチャネルからの１つのチャネルの予測に対応する。チャネル間の対応は予測モードをテストし、動きベクトルまたはブロックベクトルを探索するときに、チャネルのペア間の相関がビデオエンコーダにおいて発見され得る程度まで繰り返される。

【００６９】

主成分分析（ＰＣＡ）または同等の方法を使用するチャネル間非相関のアプローチが、ＰＣＡエンコーダ１６０において使用される。ＰＣＡは、「固有空間」を定義する１組のベクトル（「主成分」または「固有ベクトル」）を生成する。ＰＣＡの出力は、一次元ベクトルである。説明される構成では、出力１次元ベクトルが、ＰＣＡが実行されるテンソルの幅および高さの次元数に対応する２次元形態に変換されると理解される。各固有ベクトルは、固有空間を定義する他の固有ベクトルに対して直交する。直交性は、各固有ベクトルが特徴マップを表現する際に行う寄与に冗長性がないことを意味する。入力テンソルからのそのようなベクトルのセットの導出は、ベクトルのセットが入力テンソルよりも小さい場合、「分解」または次元削減と呼ばれることがある。固有空間は、テンソルのチャ

ネルを主成分のスカラー倍数で表現できる空間である。主成分の順序付けは、成分における最大分散量が先行する成分によって捕捉されていないテンソルにおける残りの未説明分散の最大量を捕捉するようなものである。したがって、ベクトルのセットの最初のＮ個の成分を選択することは、ここで、Ｎはテンソルのチャネルカウントよりも小さく、オリジナルのテンソルと比較して、デコーダにおいてテンソルを再構成（または「投影」）するときに、精度の低下を最小限に抑えながら、次元が削減された空間でテンソルを表現することを可能にする。１つの空間から別の空間へのテンソルの変換はドット積演算を使用して達成され、入力は１次元アレイとして適切に配置され、必要に応じて転置される。

【００７０】

主成分分析のための方法は「特異値分解（ｓｉｎｇｕｌａｒ　ｖａｌｕｅ　ｄｅｃｏｍｐｏｓｉｔｉｏｎ）」（ＳＶＤ）を含み、ＰＣＡエンコーダ１６０の分解機能（以下のモジュール５４８に関連して説明する）として働くことができる。５４８においてＰＣＡエンコーダ１６０において使用され得る分解の他の方法は、ＱＲ分解を含み、それは、a Gram-Schmidt process、Householder transformations、またはGivens rotations、またはLU（下位）decomposition、またはCholesky decompositionを使用して実行され得る。各分解方法で使用される値Ｎは、テンソル１１５内のチャネルの数よりも実質的に小さくすることができる。たとえば、テンソル１１５内の２５６個のチャネルを用いて、最初の２５個の主成分のみを使用して、ＰＣＡデコーダ１７０はＣＮＮヘッド１５０がロスレス近くで動作するために十分に高い忠実度の再構成されたテンソルを生成することができ、すなわち、ＣＮＮバックボーン１１４とＣＮＮヘッド１５０との間に非可逆圧縮がないか、または無視できるようにすることができる。

【００７１】

主成分または基底ベクトルは非可逆圧縮を使用して符号化され、いくつかの実施形態ではエンコーダが基底ベクトルの非可逆表現を使用する必要性をもたらす。エンコーダにおける基底ベクトルの非可逆表現の使用は、係数を生成するために使用される順変換が復号されたテンソルのチャネルに基底ベクトルを投影し戻すために使用される逆変換に比べて、より反転可能であることを保証する。分解モジュール５４８はまた、一般に、ゼロセンタリングされたデータ上での動作を必要とし、またはそれから利益を得て、結果としてより正確な基底ベクトルを生成する。主成分または基底ベクトルは原点に対して相対的であるので、分解されるテンソルも原点の周りに中心を置かなければならず、すなわちゼロ中心でなければならない。ゼロ中心テンソルの使用は直交であり、各連続基底ベクトルがテンソルにおける最大量の未説明分散を正確に示す特性を有する基底ベクトルの生成を可能にする。連続する基底ベクトルがそれぞれ、テンソルにおける最大量の未説明分散を説明するとき、所与の程度までテンソルを説明するために、低減された数の基底ベクトルが必要とされる。言い換えれば、ゼロ中心テンソルの分解は最大の次元数の低減を可能にし、したがって最大の圧縮効率を可能にする。

【００７２】

テンソルエンコーダ５００はテンソル１１５を受け取り、テンソル１１５を抽出平均モジュール５１０に入力する。分解モジュール５４８がゼロ中心データに対して動作することを可能にするために、抽出平均モジュール５１０は、テンソルのチャネル次元にわたって平均を実行することによって平均特徴マップ５１１を生成するように動作する。平均特徴マップ５１１のバージョン、非可逆平均特徴マップ５２２は、導出された非可逆係数５３８を使用してドット積モジュール５４０によって順変換されて、チャネルごとの平均特徴マップ５４２を生成する。モジュール２４０で実施されるドット積は特徴マップの「平坦化された」バージョンに対して実行され、すなわち、２次元アレイから１次元アレイに再形成される。ドット積が特徴マップおよび基底ベクトルに対して実行される場合、特徴マップおよび基底ベクトルの両方が一方が水平方向に、他方が垂直方向に平坦化される。本書の残りの部分ではドット積が（２次元）特徴マップまたは他のデータに対して実行されると言われる場合、１次元表現への、および１次元表現からの再形成は暗黙的であると理解される。チャネルごとの平均特徴マップ５４２は、減算器５４４によってテンソル内

の各チャネルから差し引かれ、ＤＣ正規化テンソル５４６を生成する。

【００７３】

サンプル領域における非可逆符号化の潜在的な使用、および、整数サンプル領域への／からの浮動小数点領域からの量子化、から生じる不一致を考慮するために、ＰＣＡエンコーダ１６０は非可逆平均特徴マップ５２２を生成するために、平均特徴マップ５１１を量子化し、符号化し、復号し、逆量子化する。平均特徴マップ５１１は、量子化器モジュール５１２によって量子化されて、整数平均特徴マップ５１３を生成する。整数平均特徴マップ５１３は、サブピクチャエンコーダ５１４に渡される。サブピクチャエンコーダ５１４は、特徴マップ５１３をサブピクチャにパックする。サブピクチャは、ビットストリーム部分５１５として符号化される。ビットストリーム部分５１５は、テンソルのチャネルにわたって平均化された平均特徴マップ５１１の圧縮表現に対応する。サブピクチャエンコーダ５１４はまた、再構成された整数平均特徴マップ５１６を出力し、これは、ＰＣＡデコーダ１７０内で生成されたパックされていないデコードされた平均特徴マップ８４４に対応し、両方とも、平均特徴マップ５１１の非可逆表現である。再構成された整数平均特徴マップ５１６は、逆量子化器モジュール５２０によって浮動小数点領域に戻されて、非可逆平均特徴マップ５２２を生成する。非可逆平均特徴マップ５２２は、整数平均特徴マップ５１６の浮動小数点表現を提供する。

【００７４】

振幅がテンソル１１５内のチャネルごとに変化するので、平均特徴マップ５２２が除去される度合いは、非可逆平均特徴マップ５２２とテンソル１１５のそれぞれのチャネルとのドット積を実行するドット積モジュール５２４によって生成される「平均係数」を使用して変化させることができる。ドット積モジュールは、テンソル１１５のチャネルごとに１つの平均係数を含む平均係数のセット５２６を生成する。平均係数５２６は、量子化器モジュール５２８によってサンプル領域に量子化されて、整数平均係数５３０を生成する。平均係数５３０は、サブピクチャにパックされ、サブピクチャエンコーダ５３２によって符号化されて、ビットストリーム部分５３３および非可逆サブピクチャ表現５３４を生成する。ビットストリーム部分５３３は、平均テンソル値５１１に関する係数５２６に対応する。復号された平均係数を表す非可逆サブピクチャ表現５３４は、逆量子化器モジュール５３６に供給される。逆量子化器モジュール５３６は再構成された平均係数５３８、すなわち、平均係数５２６の非可逆浮動小数点バージョンを出力する。したがって、ゼロ平均化テンソル５４６は、ＰＣＡデコーダ１７０において導出されるゼロ平均化テンソル８５２に対応する。

【００７５】

損失の原因には、量子化および非可逆符号化が含まれる。平均係数に関する任意の損失に対するチャネルごとの平均特徴マップ５４２の生成の感度（sensitivity）に起因して、サブピクチャエンコーダ５３２は、サブピクチャエンコーダ５１４に必要とされるよりも高い品質レベル（より小さい量子化ステップサイズまたはより低い量子化パラメータ）で動作する。サブピクチャエンコーダ５３２は、ビットストリーム部分５３３においてより多くのビットが費やされるという犠牲を払って、より高い品質のために可逆モードで動作され得る。ＨＥＶＣなどの規格は、変換および量子化処理をバイパスすることによって、可逆演算を局所的に提供するための手段として、ＣＵのための「変換量子化バイパス」モードを提供する。ＶＶＣなどの規格は代わりに、量子化プロセスが可逆プロセスになるように十分に低いＱＰを使用し、変換からの損失を回避するために、変換スキップモードが独立して選択され得る。可逆符号化のそのような局所適用は「混合」非可逆／可逆符号化を可能にし、可逆符号化は、パックされた係数を含むサブピクチャの領域のためなど、ピクチャの高感度部分（highly sensitive portions）において使用される。平均またはＤＣ除去の結果は、分解モジュール５４８における基底ベクトル導出により適したゼロ中心テンソル、すなわちＤＣ正規化テンソル５４６である。

【００７６】

分解モジュールは、基底ベクトルを生成するためにテンソル５４６のチャネルにわたっ

て分解機能を実行するように動作し、基底ベクトルは、成分の数と表現の精度との間の許容可能なトレードオフを有するテンソルを表すことが可能な成分である。分解のための様々な機構が、分解モジュール５４８において使用され得る。単一値分解（ＳＶＤ）は、分解モジュール５４８での使用に適した、一般的で計算的に高速な分解方法の１つである。非可逆符号化がソースデバイス１１０から宛先デバイス１４０に伝達される必要がある異なるタイプのデータ（平均特徴、係数、基底ベクトル）のために使用され得るにもかかわらず、データのサンプル表現の使用は浮動小数点領域から整数領域への量子化を必要とし、その範囲は、ビデオ符号化および復号処理のビット深度によって制約される。量子化データのための単一のコヒーレントコンテナを生成するために、画像フレームは、複数のサブピクチャから構成され得、その各々は独立して符号化されているサブピクチャと、ＰＣＡエンコーダ１６０において利用可能にされる再構成バージョン（すなわち、対応するデコーダにおいて生成されるようなサブピクチャのバージョン）である。以下、図１３Ａ及び図１３Ｂを参照して、サブピクチャの配置について説明する。逆量子化モジュール５２０および５３６を使用する結果として、非可逆符号化による圧縮損失が、システム５００によって実施されるテンソル符号化プロセスにおいて考慮される。

【００７７】

分解モジュール５４８は、基底ベクトル５５０のセットを生成する。基底ベクトル５５０のセットのサイズは一般に、テンソル１１５のチャネルの数の一部である。基底ベクトル５５０は、テンソル１１５が忠実度の最小損失で近似的に表現され得る新しい部分空間を定義する。テンソルを分解するときにチャネルがあるより少ない基底ベクトルを生成することは、「次元削減」と呼ばれる。次元削減は、テンソル１１５のチャネルを表現することに最も重要でない寄与をする基底ベクトルを破棄するか、または生成しないことによって、非可逆圧縮を達成するための手段を提供する。例えば、２５５または２５個の基底ベクトルの入力チャネルカウントの約１０％が、分解モジュール５４８によって生成され得る。基底ベクトルはテンソル１１５内の特徴マップを表現することができる固有空間を形成し、次元、したがって、ピクチャまたはサブピクチャ内の符号化されたエリアが大幅に低減される。基底ベクトル５５０は、量子化器５５２によって量子化されて、整数基底ベクトルテンソル５５４を生成する。量子化器５５２は、基底ベクトル５５０を使用してテンソルのための（整数）係数を生成するように動作する。整数基底ベクトルテンソル５５４は、サブピクチャにパックされ、サブピクチャエンコーダ５５６によって符号化される。サブピクチャエンコーダ５５６は、ビットストリーム部分５５７を出力し、再構成されたサブピクチャをアンパックした結果として、再構成された整数テンソル５５８を出力する。ビットストリーム部分５５７は、モジュール５４８によるテンソル５４６の分解によって生成された成分５５０に対応する。逆量子化器モジュール５６０は、再構成された整数テンソル５５８を逆量子化して、再構成された基底ベクトルテンソル５６２を生成する。

【００７８】

順変換は、ドット積モジュール５６４によって実行され、ゼロ平均テンソル５４６の各チャネルと、再構成基底ベクトルテンソル５６２内の基底ベクトルの各々とのドット積を生成して、係数のセット５６６を生成する。ゼロ平均テンソル５４６内の２５６個のチャネルおよびテンソル５６２内の２５個（２５個）の成分を用いて、セット５６６として出力される２５６×２５＝６４００個の係数が存在する。係数５６６は、量子化器モジュール５６８によって浮動小数点領域から整数領域に量子化される。量子化器モジュール５６８は、係数サンプル５７０のセットを出力する。係数サンプル５７０は、サブピクチャにパックされ、サブピクチャエンコーダ５７２によってビットストリーム部分５７３に符号化される。ビットストリーム部分５７３は、成分５５０に関連する係数５６６に対応する。ビットストリーム部分５１５、５３３、５５７、および５７３は４つのサブピクチャに対応し、サブピクチャビットストリームコンバイナ５８０によって結合されて、符号化ビットストリーム１２１を生成する。ＰＣＡデコーダ１７０に見られる再構成された係数８１８の忠実度に対するタスク性能の感度（sensitivity）のために、サブピクチャエンコ

ーダ５７２において使用される品質レベルは基底ベクトルを符号化するために、すなわち、サブピクチャエンコーダ５５６によって使用される品質レベルよりも相対的に高い。

【００７９】

ＰＣＡエンコーダ１６０の構成では、分解モジュール５４８が可変数の基底ベクトル５５０を出力するように動作可能であり、その結果、少なくともテンソル１１５の最小分散量は最大数の成分Ｍの限界まで、基底ベクトル５５０によって表され、例えば、２５６個のチャネルを有するテンソルについては２５である。「固有値」は、基底ベクトル５５０内の各ベクトルに関連付けられ、それぞれの基底ベクトルを使用することによってどれだけの分散が適応されるか、または「説明される」かを示す。各連続する基底ベクトルが、テンソル１１５内の最大量の未説明の分散を説明するので、基底ベクトル５５０の固有値は基底ベクトル５５０内のベクトルにわたって進行するとき、大きさが減少している。最小分散閾値、８５％、９０％、または９５％、９８％、または他の値などを確立することができる。最大値Ｎは第１のＮ番目の固有値の累積和が最小分散閾値未満であり、Ｎが分解モジュール５４８によって生成された基底ベクトルの数以下である場合に選択され得る。あるいは、最小値Ｎが第１のＮ番目の固有値の累積和が最小分散閾値よりも大きく、Ｎが分解モジュール５４８によって生成された基底ベクトルの数以下である場合に選択され得る。Ｎ個の基底ベクトルのサブセットが基底ベクトル５５０として出力される。基底ベクトル５５０を（図１２Ａに関連して説明したように）サブピクチャ１２１６にパッキングし、ビットストリーム部分５５７に符号化するためのモジュール５５２、５５６はいくつかの構成では最初のＮ個の基底ベクトルを符号化し、残りのベクトル（存在する場合）が順変換に使用されないようにし、サブピクチャ１２１６にパッキングすることができる。その中でより小さい分散を有するテンソル１１５の場合、選択された値Ｎは、２５などの固定しきい値よりも低くなり得る。Ｎ個の基底ベクトルを選択した結果、サブピクチャ１２１６にパックする基底ベクトルのエリアおよびサブピクチャ１２１４にパックする係数の数（図１２Ａに関連して説明したように）が低減され、ビットストリーム１２１の符号化サイズが低減され得る。ネットワークの分割点が複数のテンソルを含む場合、たとえば、ＦＰＮが使用される場合、値Ｎは、ＦＰＮ中の各レイヤについて独立して導出され得、ＣＮＮバックボーン１１４からのＦＰＮの各レイヤについて生成されたテンソルに含まれる複雑さ（したがって、分散）の差への適応を可能にする。圧縮されている各テンソルについて決定された値Ｎは、ＰＣＡエンコーダ１６０によってビットストリーム１２１に符号化され、ＰＣＡデコーダ１７０が復号ピクチャから正しい数の基底ベクトルをアンパックすることを可能にする。ＰＣＡデコーダ１７０は、基底ベクトルテンソルカウントを決定するために、アンパックされた数の基底ベクトルＮを使用し、サブピクチャ１２１６におけるパッキング位置を確立するために、最悪のケースの基底ベクトルカウントＭを使用する。ＰＣＡデコーダ１７０は新しい値Ｎが復号されるときにＮを更新しながら、現在および後続のフレームのために復号された値Ｎを使用して、ビットストリーム１４３から復号された基底ベクトルとして最初のＮ個の位置における基底ベクトルをアンパックすることができる。

【００８０】

損失を低減するために、平均係数５３０および係数サンプル５７０などの感度データは、可逆的に符号化され得る。平均係数５３０および係数サンプル５７０は、ＶＶＣとは異なる符号化を使用して符号化され得る。ビデオサンプルとして符号化するために、５３０および５７０をそれぞれのサブピクチャにパッキングする代わりに、それらは、数のリストのための代替の一般的な方式を使用して符号化され得る。１つの方式は「ＤｅｅｐＣＡＢＡＣ」であり、それによって、低い大きさでクラスタ化された大きさを有する数のシーケンスと、多くのゼロ値の数（たとえば、ガウス分布または同様の分布）とが、ＶＶＣまたはＨＥＶＣなどの規格における残差サンプルに使用されるものと同様の二値化方式とともに、コンテキスト適応型バイナリ算術符号化を使用して効率的に符号化される。整数領域における係数のための可逆符号化または非可逆符号化の使用にかかわらず、浮動小数点領域から整数領域への変換は（浮動小数点値がビットストリーム１２１において表されな

い限り、あまり効率的ではない）他のやり方で不可避であるいくらかの損失を導入する。係数５３０および５７０がＤｅｅｐＣＡＢＡＣなど、ＶＶＣとは別個の方式を使用してビットストリーム１２１中で表される場合、ＤｅｅｐＣＡＢＡＣビットストリームは、「ユーザデータＳＥＩメッセージ」または他の専用ＳＥＩメッセージなど、ＳＥＩメッセージを使用してフレームに関連付けられ得る。各ピクチャ符号化基底ベクトルについて、関連するＳＥＩメッセージは係数および基底ベクトルからの特徴マップの再構成を可能にするために、ピクチャに付随する。

【００８１】

量子化器５１２、５２８、５５２、および５６８は概して、すべて浮動小数点値を、ビデオフレームのビット深度、たとえば、１０ビット符号化が使用されるときに０～１０２３によって与えられるものに対応する整数サンプル範囲に量子化する。サンプルへの量子化は閉範囲の値への量子化を意味するが、ＶＶＣに対する代替の圧縮方式が使用される場合、開（非制限）範囲への量子化が可能である。たとえば、サブピクチャエンコーダ５３２および５７２がＤｅｅｐＣＡＢＡＣなどのアルゴリズムを使用する場合、整数値を値の範囲に適合させる必要はない。浮動小数点値をサンプル範囲にフィッティングするとき、量子化器５１２、５２８、５５２、および５６８はそれぞれ、様々な粒度に設定され得るそれぞれの「量子化範囲」を使用して動作する。各それぞれの量子化器の量子化範囲は、テンソルが量子化されるたびに決定され得る。各決定された量子化範囲はＰＣＡデコーダ１７０が整数値を逆量子化して浮動小数点領域の正しい範囲に戻すことができるように、ビットストリーム１２１においてシグナリングされる必要がある。

【００８２】

ビットストリーム１１５は、ＣＮＮバックボーン１１４およびＣＮＮヘッド１５０によって形成されるニューラルネットワークのステージから取得される。したがって、テンソル５４６は、ＣＮＮバックボーン１１４およびＣＮＮヘッド１５０によって形成されるニューラルネットワークのステージからのものでもある。図５は１つのテンソル１１５のＰＣＡベースの圧縮を示しているが、特徴ピラミッドネットワークが存在する点でＣＮＮが分離される場合、圧縮を必要とする複数のテンソルが存在する。ＦＰＮが使用されるとき、ＰＣＡエンコーダ１６０は、ＦＰＮの各テンソルに対して動作する。サブピクチャエンコーダ５１４、５３２、５５６、および５７２はＦＰＮのテンソル間で共有され得、パッキングおよびアンパッキングはＦＰＮ中の各テンソルに対応するデータのためにそれぞれのサブピクチャ中のエリアを予約することが可能である。ＦＰＮが使用される場合、ＦＰＮの各レイヤを処理するとき、量子化器５１２、５２８、５５２、および５６８、ならびに逆量子化器５２０、５３６、５６０のために別個の量子化範囲が使用され得る。別個の量子化範囲は、図１３を参照して説明したＳＥＩメッセージにおいてＰＣＡデコーダ１７０によって使用するためにビットストリーム１２１において符号化される。

【００８３】

ＰＣＡエンコーダ１６０の別の構成では、量子化器５１２、５２８、５５２、および５６８によって決定された量子化範囲は、それぞれの量子化器の前の量子化範囲に設定され、現在のＰＣＡ符号化動作のために量子化されているデータ（データ５１１、５２６、５５０、および５６６）の範囲に適応するために、必要に応じて増加される。量子化範囲を設定すると、ピクチャからピクチャへのメモリ効果が生じ、これにより、テンソル圧縮動作のシーケンスにわたって量子化範囲が「拡大」する。テンソルのシーケンスにわたって増大する量子化範囲は、基底ベクトルなどのデータの大部分のサンプル値を変更する特定のテンソルに見られる外れ値に対する感度を低下させる。テンソルのシーケンスにわたって増大する量子化範囲は、連続するテンソル圧縮動作のパックされたピクチャがビデオシーケンスとして見られるときに観察され得る「ちらつき」効果を抑制することができる。パックされたピクチャにおけるちらつきの抑制はエンコーダが圧縮された形態で効率的に検出および表現することが困難である輝度シフトを低減することによって、ビデオエンコーダ（たとえば、エンコーダ５１４、５３２、５５６、および５７２）を使用する圧縮を容易にする。ビデオエンコーダがランダムアクセス構成におけるイントラピクチャなどの

周期的リフレッシュを使用する場合、量子化範囲は、そのようなピクチャの現在のテンソルのみから決定され得る。ランダムアクセスシーケンスの場合、非イントラピクチャは、前のピクチャから基底ベクトルなどのコンテンツを予測するためにインター予測符号化ツールを使用し得、連続するフレーム中の基底ベクトルを表すために使用されるサンプル値の分布の相対的安定性に起因して、より高い圧縮効率を達成する。

【００８４】

ＰＣＡエンコーダ１６０およびＰＣＡデコーダ１７０の別の構成では、ＦＰＮが使用されるとき、レイヤ間の基底ベクトルのための係数の量子化範囲（ＱＲ）間に関係が確立される。ＣＮＮバックボーン１１４およびＣＮＮヘッド１５０がＦａｓｔｅｒＲＣＮＮまたはＭａｓｋ　ＲＣＮＮネットワークを形成する場合、レイヤＮの基底ベクトル係数のＱＲは、レイヤＮ＋１のＱＲの２倍に設定され、以下同様である。ＱＲを前のレイヤのＱＲの２倍に設定することは、テンソル圧縮のために２つのレイヤが単一のレイヤに連結される場合、例えば、ダウンサンプリングされたＰ２およびＰ３が連結されてレイヤＮを形成し、Ｐ４がレイヤＮ＋１を形成する場合を含む。基底ベクトル係数について１つのＱＲがシグナリングされる場合、各レイヤについてのＱＲは、レイヤインデックスに基づいて導出される。ＣＮＮバックボーン１１４およびＣＮＮヘッド１５０がＹＯＬＯｖ３ネットワークまたは類似のものを形成するとき、解像度ベースの順序付けは、レイヤＮ＋１がレイヤＮのＱＲの２倍を有するように反転される。ビットストリーム１２１中のシグナリングがＦＰＮが使用されているときに、基底ベクトルのための係数の逆量子化のために、ＰＣＡデコーダ１７０中でどの順序付けが使用されるべきかを示す。

【００８５】

図６は、エンコーダアーキテクチャの概略ブロック図６００である。エンコーダアーキテクチャ６００は、特徴マップパッカ６１０と、パックドフレームエンコーダ６１４と、アンパッカ６２０とを含む。サブピクチャエンコーダ５１４、５３２、５５６、および５７２は、アーキテクチャ６００のインスタンスとして実装される。パッカ６１０は所与のチャネルカウント、幅、および高さ次元を有し、整数値を含む、すなわちすでに量子化されたテンソル６０８を受信し、たとえば、サブピクチャエンコーダ５１４は、整数平均特徴マップ５１３を受信する。パッカ６１０は、受信したテンソルをサンプルの２Ｄ平面アレイにパッキングする。概して、説明される構成では、テンソル６０８のそれぞれのチャネルの特徴マップが左から右および上から下の方法で、サブピクチャフレーム６１２に格納される。サブピクチャフレーム６１２は、特徴マップサイズとサブピクチャフレーム６１２の次元との間の不一致に起因するパッキングにおけるギャップの許容を含む、テンソル６０８のチャネルを保持するのに十分なサイズである必要がある。図７を参照して、一般にＶＶＣエンコーダとして実装されるパックフレームエンコーダ６１４について説明する。エンコーダ６１４は、それぞれのサブピクチャに対応する符号化ビットストリーム部分６１６を生成する。例えば、サブピクチャエンコーダ５１４は、符号化ビットストリーム部分５１５を出力する。また、エンコーダ６１４は、ビットストリーム部分６１６を復号する際に再生された非可逆バージョンに対応する、再構成（reconstructed）フレーム６１８を出力する。再構成フレーム６１８は平均特徴マップ５１３の再構成を表し、符号化に起因して引き起こされる損失は、モデル化または表現される。損失は、エンコーダ６１４で使用される特定の符号化方法によって被るものなどの符号化損失を反映する。図７に示す例では、エンコーダ６１４がＶＶＣエンコーダである。別の実装形態では、エンコーダ６１４が異なる符号化損失を被るＨＥＶＣコードであり得る。ＪＰＥＧ ＬＳまたはＤｅｐＣＡＢＡＣなどの無損失コーデックがエンコーダ６１４のために使用され得、その場合、再構成フレーム６１８はサブピクチャフレーム６１２に等しい。エンコーダ６１４のためにＨＥＶＣまたはＶＶＣが使用される場合、それらは、係数のための可逆符号化を実行するように構成され得る。

【００８６】

再構成フレーム６１８は、アンパッカ６２０に渡され、アンパッカ６２０は特徴マップを抽出して、テンソル６０８と同じ次元数を有する再構成テンソル６２２を生成する。モ

ジュール６００の動作の結果として、ＰＣＡエンコーダ１６０は、ＰＣＡデコーダ１７０に見られるバージョンに対応する特徴マップ（または係数）のバージョンを使用することが許可され、したがって、損失源が考慮されない場合よりも高いレベルの忠実度で動作する。サブピクチャエンコーダ５１４、５３２、５５６、および５７２は、ループフィルタリングが一般に、復号されたピクチャの人間の消費のために最適化されるので、内部的にもサブピクチャ境界にわたってもループフィルタリングを無効にするように構成される。システム１００は、ビデオ圧縮を手段として使用して、ＣＮＮバックボーン１１４からＣＮＮヘッド１５０に伝搬される必要がある中間テンソルデータに対して実行される次元縮小から得られるデータを効率的に表す。

【００８７】

図７は、ビデオエンコーダ６１４の機能モジュールを示す概略ブロック図７００である。ビデオエンコーダ６１４は、全体的なピクチャを定義するサブピクチャのセットの中で１つのサブピクチャを符号化する。すべてのサブピクチャが１つの符号化パスにおいて１つのピクチャとして符号化されることが可能であるが、１つの符号化パスの使用はパイプラインにおける非可逆符号化を考慮するためのＰＣＡエンコーダ１６０におけるフィードバックが可能ではないことを意味する。一般に、データは、固定サイズのサブブロックへのブロックの分割など、サンプルまたは係数のグループ内のビデオエンコーダ６１４内の機能モジュール間を通過するか、またはアレイとして通過する。ビデオエンコーダ６１４は、図２Ａおよび２Ｂに示されるように、汎用コンピュータシステム２００を使用して実装され得、様々な機能モジュールは、コンピュータシステム２００内の専用ハードウェアによって、ハードディスクドライブ２０５上に常駐し、プロセッサ２０５によってその実行において制御されるソフトウェアアプリケーションプログラム２３３の１つまたは複数のソフトウェアコードモジュールなど、コンピュータシステム２００内で実行可能なソフトウェアによって実装され得る。代替的に、ビデオエンコーダ６１４は、専用ハードウェアとコンピュータシステム２００内で実行可能なソフトウェアとの組合せによって実装され得る。ビデオエンコーダ６１４および説明された方法は代替的に、説明された方法の機能またはサブ機能を実行する１つまたは複数の集積回路などの専用ハードウェアで実装され得る。そのような専用ハードウェアは、グラフィックプロセッシングユニット（ＧＰＵ）、デジタルシグナルプロセッサ（ＤＳＰ）、特定用途向け標準製品（ＡＳＳＰ）、特定用途向け集積回路（ＡＳＩＣ）、フィールドプログラマブルゲートアレイ（ＦＰＧＡ）、または１つもしくは複数のマイクロプロセッサおよび関連するメモリを含み得る。特に、ビデオエンコーダ６１４は、各々がソフトウェアアプリケーションプログラム２３３の１つまたは複数のソフトウェアコードモジュールとして実装され得るモジュール７１０～７９０を含む。

【００８８】

図７のビデオエンコーダ６１４は汎用ビデオコーディング（ＶＶＣ）ビデオ符号化パイプラインの一例であるが、本明細書で説明する処理ステージを実行するために他のビデオコーデックも使用され得る。例えば、ＨＥＶＣを使用することができる。説明する例は、符号化データのビットストリームを生成する。他のコーデックが使用された場合、いくつかの実装形態は、フレームフォーマットなどの異なるフォーマットにデータをパックし得る。ビデオエンコーダ６１４は、サブピクチャの一連のフレームなど、サブピクチャフレームデータ６１２を受信し、各フレームは、１つまたは複数のカラーチャネルを含む。フレームデータ６１２は、使用中のプロファイルによってサポートされる任意のクロマフォーマットおよびビット深度、たとえば、８～１０ビットのサンプル精度で、ＶＶＣ規格の「Ｍａｉｎ　１０」プロファイルに対して４：０：０、４：２：０であり得る。ブロックパーティショナ７１０は最初に、フレームデータ６１２を、形状が概ね正方形であり、ＣＴＵのための特定のサイズが使用されるように構成されたＣＴＵに分割する。ＣＴＵの最大有効サイズはたとえば、「シーケンスパラメータセット」中に存在する「ｓｐｓ＿ｌｏｇ２＿ｃｔｕ＿ｓｉｚｅ＿ｍｉｎｕｓ５」シンタックス要素によって構成される、３２×３２、６４×６４、または１２８×１２８ルマサンプルであり得る。ＣＴＵサイズはまた

、さらなる分割を伴わないＣＴＵが１つのＣＵを含むので、最大ＣＵサイズを提供する。ブロックパーティショナ７１０はさらに、ルマコーディングツリーおよびクロマコーディングツリーに従って、各ＣＴＵを１つまたは複数のＣＢに分割する。ルマチャネルは、一次（primary）カラーチャネルと呼ばれることもある。各クロマチャネルは、二次（secondary）カラーチャネルとも呼ばれ得る。ＣＢは様々なサイズを有し、二乗アスペクト比と非二乗アスペクト比の両方を含み得る。しかしながら、ＶＶＣ規格では、ＣＢ、ＣＵ、ＰＵ、およびＴＵは常に２のべき乗である辺長を有する。したがって、７１２として表される現在のＣＢは、ＣＴＵのルマコーディングツリーおよびクロマコーディングツリーに従って、ＣＴＵの１つまたは複数のブロックにわたる反復に従って進行する、ブロックパーティショナ７１０から出力される。

【００８９】

フレームデータ６１４の最初の分割から得られるＣＴＵは、ラスタ走査順序で走査され得、１つまたは複数の「スライス」にグループ化され得る。スライスは、「イントラ」（または「Ｉ」）スライスであり得る。イントラスライス（Ｉスライス）は、スライス内のすべてのＣＵがイントラ予測されることを示す。一般に、符号化レイヤビデオシーケンス（ＣＬＶＳ）中の第１ピクチャはＩスライスのみを含み、「イントラピクチャ」と呼ばれる。ＣＬＶＳは「ランダムアクセスポイント」（すなわち、復号が開始され得るビデオシーケンス中の中間フレーム）を形成する、周期的なイントラピクチャを含み得る。代替として、スライスは、それぞれ、スライス中の単予測および双予測の追加の利用可能性を示す、単予測または双予測（それぞれ、「Ｐ」または「Ｂ」スライス）であり得る。

【００９０】

ビデオエンコーダ６１４は、ピクチャ構造に従ってピクチャのシーケンスを符号化する。１つのピクチャ構造は「低遅延」であり、その場合、インター予測を使用するピクチャは、シーケンス中で以前に発生したピクチャのみを参照し得る。低遅延は、後続のピクチャによる可能な参照のために記憶されることに加えて、各ピクチャが復号されるとすぐに出力されることを可能にする。別のピクチャ構造は「ランダムアクセス」であり、ピクチャの符号化順序は、表示順序とは異なる。ランダムアクセスはインター予測されたピクチャが、復号されたが、まだ出力されていない他のピクチャを参照することを可能にする。表示順序に関して将来の参照ピクチャが復号ピクチャバッファ内に存在するように、ピクチャバッファリングの度合いが必要とされ、その結果、複数フレームの遅延が生じる。

【００９１】

４：０：０以外のクロマフォーマットが使用されているとき、Ｉスライスでは、各ＣＴＵのコーディングツリーが６４×６４レベルより下で、１つはルマ用であり、別の１つはクロマ用である２つの別個のコーディングツリーに発散し得る。別個のツリーの使用は、ＣＴＵのルマ６４×６４エリア内のルマとクロマとの間に異なるブロック構造が存在することを可能にする。たとえば、大きいクロマＣＢは、多数のより小さいルマＣＢとコロケートされ得、逆もまた同様である。ＰまたはＢスライスにおいて、ＣＴＵの単一のコーディングツリーは、ルマおよびクロマに共通のブロック構造を定義する。単一ツリーの結果として生じるブロックは、イントラ予測またはインター予測され得る。

【００９２】

ピクチャのスライスへの分割に加えて、ピクチャはまた、「タイル」に分割され得る。タイルは、ピクチャの矩形領域をカバーするＣＴＵのシーケンスである。ＣＴＵ走査は、各タイル内でラスタ走査方式で行われ、１つのタイルから次のタイルに進む。スライスは、整数個のタイル、または所与のタイル内のＣＴＵの整数個の連続する行のいずれかであり得る。

【００９３】

各ＣＴＵについて、ビデオエンコーダ６１４は、２つのステージで動作する。第１ステージ（「サーチ」ステージと呼ばれる）では、ブロックパーティショナ７１０がコーディングツリーの様々な潜在的な構成をテストする。コーディングツリーの各潜在的構成は、関連する「候補」ＣＢを有する。第１ステージは比較的低い歪みで比較的高い圧縮効率を

提供するＣＢを選択するために、様々な候補ＣＢをテストすることを伴う。テストは一般に、ラグランジュ最適化を伴い、それによって、レート（すなわち、符号化コスト）および歪み（すなわち、入力フレームデータ６１２に対する誤差）の重み付けされた組合せに基づいて候補ＣＢが評価される。「最良」候補ＣＢ（すなわち、評価されたレート／歪みが最も低いＣＢ）が、ビットストリーム６１６への後続の符号化のために選択される。候補ＣＢの評価に含まれるのは、所与のエリアに対してＣＢを使用するか、または様々な分割オプションに従ってエリアをさらに分割し、より小さい結果のエリアの各々をさらなるＣＢで符号化するか、またはエリアをさらに分割するオプションである。結果として、コーディングツリーおよびＣＢ自体の両方が、サーチステージにおいて選択される。

【００９４】

ビデオエンコーダ６１４はＣＢ、たとえば、ＣＢ７１２ごとに、矢印７２０によって示される予測ブロック（ＰＢ）を生成する。ＰＢ７２０は、関連するＣＢ７１２のコンテンツの予測である。減算器モジュール７２２は、ＰＢ７２０とＣＢ７１２との間に、７２４として示される差分（または、空間領域における差分を参照する「残差」）を生成する。差分７２４は、ＰＢ７２０およびＣＢ７１２における対応するサンプル間のブロックサイズ差分である。差分７２４は、変換され、量子化され、矢印７３６によって示される変換ブロック（ＴＢ）として表される。ＰＢ７２０および関連するＴＢ７３６は典型的にはたとえば、評価されたコストまたは歪みに基づいて、多くの可能な候補ＣＢのうちの１つから選択される。

【００９５】

候補コーディングブロック（ＣＢ）は、関連付けられたＰＢおよび結果として生じる残差のためにビデオエンコーダ６１４に利用可能な予測モードのうちの１つから生じるＣＢである。ＴＢ７３６は、ビデオエンコーダ６１４中の予測されたＰＢと組み合わされると、ビットストリーム中の追加のシグナリングを犠牲にして、復号されたＣＢとオリジナルのＣＢ７１２との間の差を低減する。

【００９６】

したがって、各候補コーディングブロック（ＣＢ）、すなわち変換ブロック（ＴＢ）と組み合わせた予測ブロック（ＰＢ）は、関連する符号化コスト（または「レート」）と、関連する差分（または「歪み」）とを有する。ＣＢの歪みは、典型的には絶対差の和（ＳＡＤ）、二乗差の和（ＳＳＤ）、または差に適用されるアダマール変換など、サンプル値の差として推定される。各候補ＰＢから結果として生じる推定は予測モード７８７を決定するために、差分７２４を使用してモードセレクタ７８６によって決定され得る。予測モード７８７は、現在のＣＢのための特定の予測モード、たとえば、イントラフレーム予測またはインターフレーム予測を使用する決定を示す。各候補予測モードと対応する残差符号化とに関連する符号化コストの推定は、残差のエントロピー符号化よりも著しく低いコストで実行され得る。したがって、いくつかの候補モードが、リアルタイムビデオエンコーダにおいてさえ、レート歪み検知において最適モードを決定するために評価され得る。

【００９７】

レート歪みに関して最適モードを決定することは、典型的にはラグランジュ最適化のバリエーションを使用して達成される。

【００９８】

ラグランジアンまたは同様の最適化処理は、（ブロックパーティショナ７１０による）ＣＢへのＣＴＵの最適なパーティショニングの選択、ならびに複数の可能性からの最良の予測モードの選択の両方に使用され得る。モード選択モジュール７８６における候補モードのラグランジュ最適化プロセスの適用を通して、最も低いコスト測定を有するイントラ予測モードが「最良」モードとして選択される。最低コストモードは選択された二次変換インデックス７８８を含み、これはまた、エントロピーエンコーダ７３８によってビットストリーム６１４において符号化される。

【００９９】

ビデオエンコーダ６１４の動作の第２ステージ（「符号化」ステージと呼ばれる）では

、各ＣＴＵの決定されたコーディングツリーに対する反復がビデオエンコーダ６１４において実行される。別個のツリーを使用するＣＴＵの場合、ＣＴＵの６４×６４ルマ領域ごとに、ルマコーディングツリーが最初に符号化され、次いでクロマコーディングツリーが符号化される。ルマコーディングツリー内ではルマＣＢのみが符号化され、クロマコーディングツリー内ではクロマＣＢのみが符号化される。共有ツリーを使用するＣＴＵの場合、単一のツリーは共有ツリーの共通ブロック構造に従ってＣＵ（すなわち、ルマＣＢおよびクロマＣＢ）を記述する。

【０１００】

エントロピーエンコーダ７３８は、可変長および固定長コードワードを使用するシンタックス要素のビットワイズ符号化と、シンタックス要素のための算術符号化モードとをサポートする。「パラメータセット」などのビットストリームの一部、例えば、シーケンスパラメータセット（ＳＰＳ）およびピクチャパラメータセット（ＰＰＳ）は、固定長コードワードと可変長コードワードとの組合せを使用する。連続部分とも呼ばれるスライスは、可変長符号化を使用するスライスヘッダと、それに続く算術符号化を使用するスライスデータを有する。スライスヘッダは、スライスレベルの量子化パラメータオフセットなど、現在のスライスに固有のパラメータを定義する。スライスデータは、スライス中の各ＣＴＵのシンタックス要素を含む。可変長符号化および算術符号化の使用は、ビットストリームの各部分内での逐次的な構文解析を必要とする。部分は、「ネットワーク抽象化レイヤユニット」または「ＮＡＬユニット」を形成するための開始コードで区切られ得る。算術符号化は、コンテキスト適応型バイナリ算術符号化プロセスを使用してサポートされる。

【０１０１】

算術的に符号化されたシンタックス要素は、１つ以上の「ビン」のシーケンスからなる。ビンはビットと同様に、「０」または「１」の値を有する。しかしながら、ビンは、ディスクリートビットとしてビットストリーム６１４内に符号化されない。ビンは、関連する予測（または「可能性が高い」または「最も可能性が高い」）値と、「コンテキスト」として知られる関連する確率とを有する。符号化されるべき実際のビンが予測値と一致するとき、「最確シンボル」（ＭＰＳ）が符号化される。最も可能性の高いシンボルを符号化することは、ビットストリーム６１４内の消費ビットに関して比較的安価であり、１つ未満のディスクリートビットに相当するコストを含む。符号化される実際のビンが可能性のある値と一致しない場合、「最低確率シンボル」（ＬＰＳ）が符号化される。最も可能性の低いシンボルを符号化することは、消費されるビットに関して比較的高いコストを有する。ビン符号化技法は、「０」対「１」の確率が歪んでいるビンの効率的な符号化を可能にする。２つの可能な値（すなわち、「フラグ」）を有するシンタックス要素については、単一のビンが適切である。多くの可能な値を有するシンタックス要素の場合、ビンのシーケンスが必要とされる。

【０１０２】

シーケンス中の後のビンの存在は、シーケンス中の前のビンの値に基づいて決定され得る。さらに、各ビンは、２つ以上のコンテキストに関連付けられ得る。特定のコンテキストの選択は、シンタックス要素中の以前のビン、隣接するシンタックス要素のビン値（すなわち、隣接するブロックからのもの）などに依存し得る。文脈符号化されたビンが符号化されるたびに、そのビン（もしあれば）のために選択されたコンテキストは、新しいビン値を反映する方法で更新される。このように、バイナリ算術符号化方式は、適応的であると言われる。

【０１０３】

また、エントロピーエンコーダ７３８によってサポートされるのは、「バイパスビン」と呼ばれる、コンテキストを欠くビンである。バイパスビンは、「０」と「１」との間の等確率分布を仮定して符号化される。したがって、各ビンは、ビットストリーム６１４中の１ビットの符号化コストを有する。コンテキストがないことはメモリを節約し、複雑さを低減し、したがって、バイパスビンは特定のビンの値の分布が歪まない場合に使用され

る。コンテキストおよび適応を使用するエントロピーコーダの一例はＣＡＢＡＣ(コンテキスト適応型バイナリ算術コーダ）として当技術分野で知られており、このコーダの多くの変形例がビデオ符号化において使用されている。

【０１０４】

ＱＰコントローラ７９０は、量子化器７３４および逆量子化器７４０による使用のための量子化ステップサイズを確立するために使用される量子化パラメータ７９２を決定する。より大きい量子化ステップサイズは一次変換係数７２８がより小さい値に量子化されることをもたらし、逆変換係数７４６の忠実度の低減を犠牲にしてビットストリーム１２１のビットレートを低減する。

【０１０５】

エントロピーエンコーダ７３８は文脈符号化されたビンとバイパス符号化されたビンとの組合せを使用して、量子化パラメータ７９２を符号化し、現在のＣＢのために使用されている場合、ＬＦＮＳＴインデックス７８８を符号化する。量子化パラメータ７９２は各スライスの最初に符号化され、スライス内の量子化パラメータ７９２の変化は「デルタＱＰ」シンタックス要素を使用して符号化される。デルタＱＰシンタックス要素は、「量子化グループ」として知られる各エリアにおいて最大で１回シグナリングされる。量子化パラメータ７９２は、ルマＣＢの残差係数に適用される。調整された量子化パラメータが、コロケートされたクロマＣＢの残差係数に適用される。調整された量子化パラメータは、オフセットのリストから選択されたマッピングテーブルおよびＣＵレベルオフセットに従ってルマ量子化パラメータ７９２からマッピングすることを含み得る。二次変換インデックス７８８は、変換ブロックに関連する残差が二次変換の適用によって一次係数に変換される係数位置にのみ有意な残差係数を含むときにシグナリングされる。

【０１０６】

ＣＢに関連する各ＴＢの残差係数は、残差シンタックスを使用して符号化される。残差シンタックスは、係数の有意性を示すために主に算術的に符号化されたビンを使用し、より低い値の大きさとともに、より高い大きさの残差係数のためにバイパスビンを予約することによって、低い大きさで係数を効率的に符号化するように設計される。したがって、非常に低い大きさの値を含む残差ブロックと、有意な係数の疎な配置とが、効率的に圧縮される。さらに、２つの残差符号化方式が存在する。変換が適用されるときに見られるように、正規の残差符号化方式は、ＴＢの左上隅に主に位置する有意係数を有するＴＢに対して最適化される。変換スキップ残差符号化方式は、変換が実行されないＴＢに利用可能であり、ＴＢ全体にわたるそれらの分布にかかわらず、残差係数を効率的に符号化することができる。

【０１０７】

マルチプレクサモジュール７８４は、各候補ＣＢのテストされた予測モードから選択された、決定された最良のイントラ予測モードに従って、イントラフレーム予測モジュール７６４からＰＢ７２０を出力する。候補予測モードは、ビデオエンコーダ６１４によってサポートされるすべての考えられる予測モードを含む必要はない。イントラ予測は３つのタイプに分けられ、第１に、ＰＢに、近くの再構成されたサンプルの平均を表す単一の値をポピュレートすることを伴う「ＤＣイントラ予測」、第２に、ＰＢに、平面に従ってサンプルをポピュレートすることを伴い、ＤＣオフセットおよび垂直および水平勾配は、近くの再構成された隣接するサンプルから導出される「平面イントラ予測」。近隣の再構成されたサンプルは、典型的にはある程度までＰＢの右に延在する現在のＰＢの上の再構成されたサンプルの行と、ある程度までＰＢを越えて下方に延在する、現在のＰＢの左に再構成されたサンプルの列と、を含み、第３に、特定の方向（または「角度」）にＰＢにわたってフィルタリングされ伝播された再構成された近隣のサンプルをＰＢにポピュレートすることを含む、「角度イントラ予測」を含む。ＶＶＣでは、６５個の角度がサポートされ、矩形ブロックは合計８７個の角度を生成するために、正方形ブロックには利用できない追加の角度を利用することができる。

【０１０８】

クロマＰＢには第４のタイプのイントラ予測が利用可能であり、それによりＰＢは「クロスコンポーネント線形モデル」(ＣＣＬＭ)モードに従ってコロケートされたルマ再構成サンプルから生成される。３つの異なるＣＣＬＭモードが利用可能であり、各モードは、隣接するルマサンプルおよびクロマサンプルから導出された異なるモデルを使用する。導出されたモデルは、コロケートされたルマサンプルからクロマＰＢのためのサンプルのブロックを生成するために使用される。ルマブロックは、行列の事前定義されたセットから選択された１つの行列を使用して、参照サンプルの行列乗算を使用してイントラ予測され得る。この行列イントラ予測（ＭＩＰ）はビデオデータの大きなセットについてトレーニングされた行列を使用することによって利得を達成し、行列は、角度、平面、またはＤＣイントラ予測モードでは容易に捕捉されない予測ブロックと参照サンプルとの間の関係を表す。

【０１０９】

モジュール７６４はまた、「イントラブロックコピー」（ＩＢＣ）方法を使用して現在フレームの近くからブロックをコピーすることによって予測ユニットを生成し得る。参照ブロックの位置はＶＰＤＵとして知られる６４×６４の領域に分割された１つのＣＴＵに相当する領域に制約され、各行またはＣＴＵ内および各スライスまたはタイル内の前のＣＴＵの現在のＣＴＵおよびＶＰＤＵの処理されたＶＰＤＵをカバーする領域はビットストリームの構成されたＣＴＵサイズにかかわらず、１つの１２８×１２８ルマサンプルに対応する領域制限までである。この領域は「ＩＢＣ仮想バッファ」として知られており、ＩＢＣ参照領域を制限し、したがって、必要とされる記憶を制限する。ＩＢＣバッファは再構成されたサンプル７５４でポピュレートされ（すなわち、ループフィルタリングの前に）、したがって、フレームバッファ７７２とは別個のバッファが必要とされる。ＣＴＵサイズが１２８×１２８であるとき、仮想バッファは現在のＣＴＵに隣接し、その左にあるＣＴＵからのサンプルのみを含む。ＣＴＵサイズが３２×３２または６４×６４であるとき、仮想バッファは、現在のＣＴＵの左に４または１６個のＣＴＵまでのＣＴＵを含む。ＣＴＵサイズにかかわらず、ＩＢＣ参照ブロックのためのサンプルを取得するための隣接ＣＴＵへのアクセスは、ピクチャ、スライス、またはタイルのエッジなどの境界によって制約される。特に、より小さい寸法を有するＦＰＮレイヤの特徴マップの場合、３２×３２または６４×６４などのＣＴＵサイズの使用は、以前の特徴マップのセットをカバーするようにより整列された参照領域をもたらす。特徴マップ配置がＳＡＤ、ＳＳＥ、または他の差分メトリックに基づいて順序付けられる場合、ＩＢＣ予測のための同様の機能マップへのアクセスは、符号化効率の利点を提供する。

【０１１０】

特徴マップデータを符号化するときの予測ブロックの残差は、自然ビデオについて見られる残差とは異なる。自然ビデオは通常、オペレーティングシステムのユーザインターフェースなどに一般的に見られるように、画像センサまたはスクリーンコンテンツによってキャプチャされる。特徴マップ残差は、多くの詳細を含む傾向がある。特徴マップ残差における詳細のレベルは、主に様々な変換の低周波数係数よりも多くのスキップ符号化の変換に適している。ＤＣＴ－２、ＭＴＳ(水平方向および垂直方向のＤＳＴ－７、ＤＣＴ－８の組合せ）、およびＬＦＮＳＴ(様々なトレーニングされた分離不可能な変換）を使用して残差符号化の利点を測定するための本発明者らによる実験は、特徴マップ残差が変換符号化から利点を得るのに十分なローカル類似性を有することを示す。しかしながら、特徴マップ残差係数の分布は、変換ブロックのＤＣ(左上）係数に向かってクラスタ化されない。言い換えれば、特徴マップデータを符号化するときに、変換が利得を示すのに十分な相関が存在する。イントラブロックコピーが特徴マップデータの予測ブロックを生成するために使用されるときにも、十分な相関が存在する。したがって、アダマールコスト推定は、ＳＡＤまたはＳＳＤコスト推定のみに依存するのではなく、特徴マップデータを符号化するときに、イントラブロックコピーの候補ブロックベクトルから生じる残差を評価するときに使用され得る。ＳＡＤまたはＳＳＤコスト推定は、符号化スキップを変換するのにより適した残差を有するブロックベクトルを選択する傾向があり、変換を使用してコ

ンパクトに符号化される残差を有するブロックベクトルを見逃す可能性がある。ＶＶＣ規格の多重変換選択（ＭＴＳ）ツールはＤＣＴ－２変換に加えて、ＤＳＴ－７およびＤＣＴ－８変換の組合せが残差符号化のために水平および垂直に利用可能であるように、特徴マップデータを符号化するときに使用され得る。

【０１１１】

イントラ予測されたルマコーディングブロックは、垂直または水平のいずれかで、等しいサイズの予測ブロックのセットに区分され得、各ブロックは１６個のルマサンプルの最小エリアを有する。このイントラサブパーティション（ＩＳＰ）アプローチは、別個の変換ブロックがルマコーディングブロック中の１つのサブパーティションから次のサブパーティションへの予測ブロック生成に寄与することを可能にし、圧縮効率を改善する。

【０１１２】

以前に再構成された隣接するサンプルが利用できない場合、例えば、フレームのエッジにおいて、サンプルの範囲の半分のデフォルトハーフトーン値が使用される。例えば、１０ビットビデオの場合、５１２の値が使用される。フレームの左上位置に配置されたＣＢには以前のサンプルが利用可能ではないので、角度および平面イントラ予測モードはＤＣ予測モード（すなわち、大きさとしてハーフトーン値を有するサンプルの平面）と同じ出力を生成する。

【０１１３】

インターフレーム予測の場合、予測ブロック７８２は、動き補償モジュール７８０によってビットストリーム内の符号化順序フレームにおいて現在のフレームに先行する１つまたは２つのフレームからのサンプルを使用して生成され、マルチプレクサモジュール７８４によってＰＢ７２０として出力される。さらに、インターフレーム予測の場合、単一のコーディングツリーが典型的には、ルマチャネルとクロマチャネルの両方のために使用される。ビットストリーム中のフレームを符号化する順序は、キャプチャまたは表示されたときのフレームの順序とは異なり得る。１つのフレームが予測のために使用されるとき、ブロックは「単予測」であると言われ、１つの関連する動きベクトルを有する。２つのフレームが予測のために使用されるとき、ブロックは「双予測」であると言われ、２つの関連する動きベクトルを有する。Ｐスライスの場合、各ＣＵは、イントラ予測または単予測され得る。Ｂスライスの場合、各ＣＵは、イントラ予測、単予測、または双予測され得る。

【０１１４】

フレームは、典型的には「ピクチャのグループ」構造を使用して符号化され、フレームの時間的階層を可能にする。フレームは複数のスライスに分割され得、スライスの各々はフレームの一部分を符号化する。フレームの時間的階層は、フレームを表示する順序で、フレームが前後のピクチャを参照することを可能にする。画像は、各フレームを復号するための依存性が満たされることを保証するために必要な順序で符号化される。予測ユニットのための参照サンプルブロックを選択およびフィルタリングするために１つまたは２つの動きベクトルを使用する代わりに、予測ユニットは複数のより小さいブロックに分割され、動きフィールドが生成され、したがって、各々のより小さいブロックは別個の動きベクトルを有するアフィンインター予測モードが利用可能である。動きフィールドは、予測ユニットの近くの点の動きベクトルを「制御点」として使用する。アフィン予測は、深く分割されたコーディングツリーを使用する必要性がより少ない、変換とは異なる動きの符号化を可能にする。ＶＶＣに利用可能な双予測モードは、シグナリングされたブロックの中心からのオフセットおよび角度を有する、選択された軸に沿った２つの参照ブロックの幾何学的ブレンドを実行する。この幾何学的分割モード（「ＧＰＭ」）は、コーディングユニットのために符号化された境界のジオメトリを角度および中心オフセットとして、２つのオブジェクト間の境界に沿ってより大きいコーディングユニットを使用することを可能にする。デカルト（ｘ、ｙ）オフセットを使用する代わりに、動きベクトル差は、サポートされる２の累乗距離のセットを用いて、方向（上／下／左／右）および距離として符号化され得る。動きベクトル予測子は、オフセットが適用されないかのように、隣接ブロ

ック（「マージモード」）から取得される。現在のブロックは、選択された隣接ブロックと同じ動きベクトルを共有する。

【０１１５】

サンプルは、動きベクトル７７８および参照ピクチャインデックスに従って選択される。動きベクトル７７８および参照ピクチャインデックスは、すべてのカラーチャネルに適用され、したがって、インター予測はＰＢではなくＰｕでの動作に関して主に説明される。各ＣＴＵの１つまたは複数のインター予測ブロックへの分解は、単一のコーディングツリーを用いて説明される。インター予測方法は、動きパラメータの数およびそれらの精度において変化し得る。動きパラメータは典型的には参照フレームのリストからのどの参照フレームが使用されるべきかを示す参照フレームインデックスと、参照フレームの各々のための空間変換とを含むが、より多くのフレーム、専用のフレーム、またはスケーリングおよび回転などの複雑なアフィンパラメータを含み得る。加えて、参照されたサンプルブロックに基づいて密な動き推定を生成するために、所定の動き精緻化処理が適用され得る。

【０１１６】

ＰＢ７２０を決定して選択し、減算器７２２でオリジナルのサンプルブロックからＰＢ７２０を減算すると、７２４として表される最も低い符号化コストを有する残差が得られ、非可逆圧縮を受ける。非可逆圧縮プロセスは、変換、量子化、およびエントロピー符号化のステップを含む。順方向一次変換モジュール７２６は、差分７２４に順方向変換を適用し、差分７２４を空間領域から周波数領域に変換し、矢印７２８によって表される一次変換係数を生成する。一次元における最大の一次変換サイズは、シーケンスパラメータセット中の「ｓｐｓ＿ｍａｘ＿ｌｕｍａ＿ｔｒａｎｓｆｏｒｍ＿ｓｉｚｅ＿６４＿ｆｌａｇ」によって構成される３２ポイントＤＣＴ－２または６４ポイントＤＣＴ－２変換のいずれかである。符号化されているＣＢがブロックサイズ（たとえば、６４×６４または３２×３２）として表される最大のサポートされる一次変換サイズよりも大きい場合、一次変換７２６は、差分７２４のすべてのサンプルを変換するためにタイル方式で適用される。非正方形ＣＢが使用される場合、タイリングはまた、ＣＢの各寸法において最大の利用可能な変換サイズを使用して実行される。たとえば、３２の最大変換サイズが使用されるとき、６４×１６ＣＢは、タイル方式で構成された２つの３２×１６一次変換を使用する。ＣＢが最大サポート変換サイズよりもサイズが大きいとき、ＣＢはタイル状にＴＢで埋められる。例えば、６４－ｐｔ変換最大サイズを有する１２８×１２８ＣＢは、２×２配置で４つの６４×６４ＴＢで埋められる。３２－ｐｔ変換最大サイズを有する６４×１２８ＣＢは、２×４配置で８つの３２×３２ＴＢで埋められる。

【０１１７】

変換７２６の適用は、ＣＢのための複数のＴＢをもたらす。変換の各アプリケーションが３２×３２よりも大きい差分７２４のＴＢ、例えば６４×６４、に対して動作する場合、ＴＢの左上の３２×３２エリアの外側のすべての結果として生じる一次変換係数７２８はゼロに設定される（すなわち、破棄される）。残りの一次変換係数７２８は、量子化器モジュール７３４に渡される。一次変換係数７２８は、ＣＢに関連する量子化パラメータ７９２に従って量子化されて、一次変換係数７３２を生成する。量子化パラメータ７９２に加えて、量子化器モジュール７３４はまた、ＴＢ内のそれらの空間位置に従って残差係数をさらにスケーリングすることによって、ＴＢ内の不均一量子化を可能にするために「スケーリングリスト」を適用し得る。量子化パラメータ７９２は、ルマＣＢ対各クロマＣＢについて異なり得る。一次変換係数７３２は、順方向二次変換モジュール７３０に渡され、非分離可能二次変換（ＮＳＳＴ）動作を実行するか、または二次変換をバイパスすることによって、矢印７３６によって表される変換係数を生成する。順方向一次変換７２６は典型的には分離可能であり、行のセット、次いで、各ＴＢの列のセットを変換する。順方向一次変換モジュール７２６は、水平方向および垂直方向におけるタイプＩＩ離散コサイン変換（ＤＣＴ－２）、または水平および垂直の変換のバイパス、または幅および高さが１６サンプルを超えないルマＴＢについて水平方向または垂直方向におけるタイプＶＩ

Ｉ離散サイン変換（ＤＳＴ－７）とタイプＶＩＩＩ離散コサイン変換（ＤＣＴ－８）との組合せのいずれかを使用する。ＤＳＴ－７とＤＣＴ－８との組み合わせの使用は、ＶＶＣ規格において「多重変換選択セット」（ＭＴＳ）と呼ばれる。

【０１１８】

モジュール７３０の順方向二次変換は概して、非分離可能変換であり、それは、イントラ予測されたＣＵの残差に対してのみ適用され、それにもかかわらず、バイパスされ得る。順方向二次変換は、１６個のサンプル（一次変換係数７２８の左上の４×４サブブロックとして構成される）または４８個のサンプル（一次変換係数７２８の左上の８×８係数において３つの４×４サブブロックとして構成される）のいずれかに対して動作して、二次変換係数のセットを生成する。二次変換係数のセットは、それらが導出される一次変換係数のセットよりも数が少なくてもよい。互いに隣接し、ＤＣ係数を含む係数のセットのみに二次変換を適用することにより、二次変換は、「低周波数非分離可能二次変換」（ＬＦＮＳＴ）と呼ばれる。そのような二次変換は、トレーニングプロセスを通して、およびそれらの分離不可能な性質およびトレーニングされたオリジンによって取得され得、ＤＣＴおよびＤＳＴの変種などの分離可能な変換によって捕捉することができない残差信号における追加の冗長性を活用することができる。さらに、ＬＦＮＳＴが適用されるとき、ＴＢ中のすべての残りの係数は、一次変換領域および二次変換領域の両方において、ゼロである。

【０１１９】

量子化パラメータ７９２は、所与のＴＢについて一定であり、したがって、ＴＢについての一次変換領域における残差係数の生成のための均一なスケーリングをもたらす。量子化パラメータ７９２は、シグナリングされた「デルタ量子化パラメータ」とともに周期的に変化し得る。デルタ量子化パラメータ（デルタＱＰ）は、「量子化グループ」と呼ばれる、所与のエリア内に含まれるＣＵについて１回シグナリングされる。ＣＵが量子化グループサイズよりも大きい場合、デルタＱＰは、ＣＵのＴＢのうちの１つを用いて１回シグナリングされる。すなわち、デルタＱＰは、ＣＵの第１量子化グループについて１回エントロピーエンコーダ７３８によってシグナリングされ、ＣＵの任意の後続の量子化グループについてはシグナリングされない。「量子化行列」を適用することによって、不均一なスケーリングも可能であり、それによって、各残差係数に適用されるスケーリング係数は、量子化パラメータ７９２と、スケーリング行列における対応するエントリとの組合せから導出される。スケーリング行列はＴＢのサイズよりも小さいサイズを有することができ、ＴＢに適用されるとき、ＴＢサイズよりも小さいサイズのスケーリング行列からの各残差係数のためのスケーリング値を提供するために、最近傍アプローチが使用される。残差係数７３６は、ビットストリーム６１４における符号化のためにエントロピーエンコーダ７３８に供給される。典型的には、ＴＵの少なくとも１つの有意な残差係数を有する各ＴＢの残差係数が走査パターンに従って、値の順序付けられたリストを生成するために走査される。走査パターンは一般に、４×４「サブブロック」のシーケンスとしてＴＢを走査し、４×４セットの残差係数の粒度で規則的な走査動作を提供し、サブブロックの配置は、ＴＢのサイズに依存する。各サブブロック内の走査および１つのサブブロックから次のサブブロックへの進行は、典型的には後方斜め走査パターンに従う。さらに、量子化パラメータ７９２は、デルタＱＰシンタックス要素を使用してビットストリーム６１４に符号化され、所与のスライスまたはサブピクチャおよび二次変換インデックス７８８中の初期値のスライスＱＰがビットストリーム６１４に符号化される。

【０１２０】

上述したように、ビデオエンコーダ６１４は、ビデオデコーダに見られる復号されたフレーム表現に対応するフレーム表現にアクセスする必要がある。したがって、残差係数７３６は逆二次変換モジュール７４４を通過し、二次変換インデックス７８８に従って動作して、矢印７４２によって表される中間逆変換係数を生成する。中間逆変換係数７４２は、量子化パラメータ７９２に従って逆量子化器モジュール６４０によって逆量子化され、矢印７４６によって表される逆変換係数を生成する。逆量子化器モジュール７４０はまた

、量子化器モジュール７３４において実行される順方向スケーリングに対応する、スケーリングリストを使用して残差係数の逆不均一スケーリングを実行し得る。逆変換係数７４６は、逆一次変換モジュール７４８に渡されて、ＴＵの、矢印７５０によって表される残差サンプルを生成する。逆一次変換モジュール７４８は、順方向一次変換モジュール７２６を参照して説明したように、最大の利用可能な変換サイズによって制約され、ＤＣＴ－２変換を水平および垂直に適用する。逆二次変換モジュール７４４によって実行される逆変換のタイプは、順方向二次変換モジュール７３０によって実行される順方向変換のタイプに対応する。逆一次変換モジュール７４８によって実行される逆変換のタイプは、一次変換モジュール７２６によって実行される一次変換のタイプに対応する。加算モジュール７５２は、残差サンプル７５０およびＰＵ７２０を加算して、ＣＵの再構成されたサンプル（矢印７５４によって示される）を生成する。

【０１２１】

再構成されたサンプル７５４は、参照サンプルキャッシュ７５６およびループ内フィルタモジュール７６８に渡される。典型的には、コストのかかるオフチップメモリアクセスを回避するためにＡＳＩＣ上のスタティックＲＡＭを使用して実装される参照サンプルキャッシュ７５６は、フレーム内の後続のＣＵのためのイントラフレームＰＢを生成するための依存性を満たすために必要とされる最小限のサンプル記憶を提供する。最小依存性は、典型的にはＣＴＵの次の行による使用のためのＣＴＵの行の底部に沿ったサンプルの「ラインバッファ」と、ＣＴＵの高さによってその範囲が設定される列バッファリングとを含む。参照サンプルキャッシュ７５６は、参照サンプル（矢印７５８によって表される）を参照サンプルフィルタ７６０に供給する。サンプルフィルタ７６０は、平滑化演算を適用して、フィルタリングされた参照サンプルを生成する（矢印７６２によって示される）。フィルタリングされた参照サンプル７６２は、イントラフレーム予測モジュール７６４によって使用されて、矢印７６６によって表される、サンプルのイントラ予測されたブロックを生成する。各候補イントラ予測モードについて、イントラフレーム予測モジュール７６４は、サンプルのブロック、すなわち７６６を生成する。サンプルのブロック７６６は、ＤＣ、平面、または角度イントラ予測などの技法を使用して、モジュール７６４によって生成される。サンプルのブロック７６６はまた、ビデオエンコーダ６１４によって行列のセットから選択された行列と、入力として隣接する参照サンプルを用いる行列乗算アプローチを使用して生成され得、選択された行列は、インデックスを使用してビットストリーム６１４にシグナリングされ、行列のセットのどの行列がビデオデコーダによって使用されるべきかを識別する。

【０１２２】

ループ内フィルタモジュール７６８は、再構成されたサンプル７５４にいくつかのフィルタリングステージを適用する。フィルタリングステージは、不連続性に起因するアーチファクトを低減するためにＣＵ境界に整列された平滑化を適用する「デブロッキングフィルタ」（ＤＢＦ）を含む。ループ内フィルタモジュール７６８に存在する別のフィルタリングステージは「適応ループフィルタ」（ＡＬＦ）であり、これは、ウィナーベースの適応フィルタを適用して、歪みをさらに低減する。ループ内フィルタモジュール７６８においてさらに利用可能なフィルタリングステージは、「サンプル適応オフセット」（ＳＡＯ）フィルタである。ＳＡＯフィルタは最初に、再構成されたサンプルを１つまたは複数のカテゴリに分類し、割り当てられたカテゴリに従って、サンプルレベルでオフセットを適用することによって動作する。

【０１２３】

矢印７７０によって表されるフィルタリングされたサンプルは、ループ内フィルタモジュール７６８から出力される。フィルタリングされたサンプル７７０は、フレームバッファ７７２に格納される。フレームバッファ７７２は典型的にはいくつかの（たとえば、最大１６個の）ピクチャを記憶する容量を有し、したがって、メモリ２０６に記憶される。フレームバッファ７７２は通常、必要とされる大きなメモリ消費のために、オンチップメモリを使用して記憶されない。したがって、フレームバッファ７７２へのアクセスは、メ

モリ帯域幅の点でコストがかかる。フレームバッファ７７２は、参照フレーム（矢印７７４によって表される）を動き推定モジュール７７６および動き補償モジュール７８０に提供する。参照フレーム７７４は、対応するサブピクチャエンコーダモジュール６００（５１４、５３２、５５６、５７２）の再構成フレーム６１８として出力され、アンパッカモジュール６２０に提供される。図７の例では、再構成されたフレームがモジュール７２６～７４８および７５２～７５４の動作に起因する、非可逆ＶＶＣ符号化の動作の結果である。

【０１２４】

動き推定モジュール７７６は（７７８として示される）いくつかの「動きベクトル」を推定し、各々は現在のＣＢの位置からのデカルト空間オフセットであり、フレームバッファ７７２中の参照フレームのうちの１つの中のブロックを参照する。参照サンプルのフィルタリングされたブロック（７８２として表される）が、動きベクトルごとに生成される。フィルタリングされた参照サンプル７８２は、モードセレクタ７８６による潜在的な選択のために利用可能なさらなる候補モードを形成する。さらに、所与のＣＵについて、ＰＵ７２０は、１つの参照ブロック（「単予測された」）を使用して形成され得るか、または２つの参照ブロック（「双予測された」）を使用して形成され得る。選択された動きベクトルについて、動き補償モジュール７８０は、動きベクトルにおけるサブピクセル精度をサポートするフィルタリング処理に従ってＰＢ７２０を生成する。したがって、（多くの候補動きベクトルに対して動作する）動き推定モジュール７７６は、（選択された候補のみに対して動作する）動き補償モジュール７８０のものと比較して簡略化されたフィルタリング処理を実行して、低減された計算の複雑さを達成することができる。ビデオエンコーダ６１４がＣＵのためのインター予測を選択するとき、動きベクトル７７８はビットストリーム６１６に符号化される。

【０１２５】

図７のビデオエンコーダ６１４は、汎用ビデオコーディング（ＶＶＣ）を参照して説明されるが、他のビデオコーディング規格または実装形態はまた、モジュール７１０～７９０の処理ステージを採用し得る。フレームデータ６１２（およびビットストリーム６１６）はまた、メモリ２０６、ハードディスクドライブ２１０、ＣＤ－ＲＯＭ、ブルーレイディスク™、または他のコンピュータ可読記憶媒体から読み出される（またはこれに対して書き込まれる）ことができる。さらに、フレームデータ６１２（およびビットストリーム６１６）は、通信ネットワーク２２０または無線周波数受信機に接続されたサーバなどの外部ソースから受信（または外部ソースに送信）され得る。通信ネットワーク２２０は制限された帯域幅を提供することができ、フレームデータ６１２を圧縮することが困難なときにネットワークを飽和させることを回避するために、ビデオエンコーダ１２０におけるレート制御の使用を必要とする。

【０１２６】

ビットストリーム６１６は、ビデオエンコーダ６１４の１つまたは複数のインスタンスによって生成され、プロセッサ２０５の制御下で協調的に動作する、フレームデータ６１２の空間セクション（ＣＴＵのコレクション）を表す１つまたは複数のスライスから構築され得る。ビットストリーム６１６はまた、１つのピクチャを形成するサブピクチャの集合として出力されるべき１つのサブピクチャに対応する１つのスライスを含み得、各サブピクチャはピクチャ中の他のスライスまたはサブピクチャのいずれかに関して独立して符号化可能であり、独立して復号可能である。ピクチャ内の任意のサブピクチャを独立して符号化および復号する能力は、任意の所与のサブピクチャに含まれるパックされた特徴マップまたは係数に対する非可逆圧縮の影響を、テンソル圧縮の後のステージにおける特徴マップまたは係数の非可逆バージョンを使用することによって、ＰＣＡエンコーダ１６０において考慮に入れることを可能にする。

【０１２７】

図８は、分散マシンタスクシステム１００の一部としてのチャネル間非相関ベーステンソルデコーダ１７０の実装を示す概略ブロック図８００である。ビデオビットストリーム

１４３は、ＶＶＣビデオデコーダを実装し、ＰＣＡエンコーダ１６０によって生成されたビットストリームを復号するピクチャデコーダ８０４に渡される。ピクチャデコーダ８０４は、ビデオビットストリーム１４３中に存在するサブピクチャを復号し、各サブピクチャは、テンソル１４９を再構成するために必要とされる異なるデータタイプに対応する。各復号されたサブピクチャは、復号されたテンソルのための情報のユニットを提供し、ユニットは、平均テンソル値５１１、係数５２６、基底ベクトル５５０、および係数５６６のうちの１つに対応する。平均特徴サブピクチャ８４０は、ピクチャデコーダ８０４によって出力され、アンパッカ８４２に渡される。アンパッカ８４２は、テンソル１４９の幅および高さに基づいて、整数平均特徴マップ８４４を抽出する。整数平均特徴マップ８４４は、逆量子化器８４６に渡され、ここで、サンプル領域から浮動小数点領域への変換が例えばビットストリーム１４３から得られる適切な量子化範囲を使用して実行され、復号された平均特徴マップ８４８をもたらす。

【０１２８】

平均係数サブピクチャ８３０もピクチャデコーダ８０４によって出力され、アンパッカ８３２に渡される。アンパッカ８３２は、テンソル１４９のチャネルカウントに基づいて整数平均係数８３４を抽出する。整数平均係数８３４は、逆量子化器８３６に渡される。逆量子化器８３６は、ビットストリーム１４３から得られた量子化範囲を使用して、平均係数８３４から、整数領域から浮動ポイント領域に変換し、浮動点平均係数８３８を出力する。ドット積モジュール８５０は、平均係数８３８のうちのそれぞれの係数を有する平均特徴８４８のドット積を実行することによって、テンソル１４９の各チャネルの平均特徴マップ８５２を生成する。逆量子化器８４６、８３６、８２４、および８１６は、ビットストリームを符号化するために実装５００において使用される対応する量子化機能（５１２、５２８、５５２、および５６８）の逆である。

【０１２９】

量子化および非可逆圧縮の前に、分解モジュール５４８によって生成された（たとえば、ＰＣＡを実行する）パックされた基底ベクトルを含むサブピクチャ８２０も、ピクチャデコーダ８０４によって出力され、アンパッカ８２２に渡される。アンパッカ８２２は、整数基底ベクトル８２３を、サブピクチャ８２０に重複しないように配置された一連の特徴マップとして抽出する。整数基底ベクトル８２３は、逆量子化器８２４に渡される。逆量子化器８２４は、浮動小数点基底ベクトル８２６を構築し、ビットストリーム１４３から得られた量子化範囲を適用する。

【０１３０】

テンソル１４３の特徴マップごとに基底ベクトルごとに１つの係数を有する係数群を含むサブピクチャ８１０も、ピクチャデコーダ８０４によって出力され、アンパッカ８１２に渡される。アンパッカ８１２は、テンソル１４９の次元数に基づいて、サブピクチャ８１０から各係数を抽出する。１つの係数は特徴マップまたはチャネルごとの基底ベクトルごとに必要とされ、所与の特徴マップまたはチャネルを再構成する際の各基底ベクトルの寄与を表し、アンパッカ８１２から出力される整数係数８１４を集合的に形成する。逆量子化器８１６は、ビットストリーム１４３から得られた量子化範囲に従って、整数係数８１４を整数領域から浮動小数点領域に変換し、浮動小数点係数８１８を出力する。ゼロ中心テンソル８５４は、係数８１８と基底ベクトル８２６とに対してドット積を実行するドット積モジュール８２８によって生成される。加算モジュール８５６は、ゼロ中心テンソル８５４を平均特徴マップ８４８と加算して、ＰＣＡデコーダ１７０からの出力としてテンソル１４９を生成する。ＰＣＡデコーダ１７０における動作の流れは、圧縮されたテンソルの次元数を元の次元数に戻し、テンソル１４９が所与のマシンタスクの実行のためにＣＮＮヘッド１５０に渡されることを可能にする。

【０１３１】

図１２Ａおよび１２Ｂに関連して以下に説明するように、各復号ピクチャ８４０、８３０、８２０および８１０内の情報は、複数のサンプルの２次元アレイに配列される。復号されたピクチャ８４０、８３０、８２０、および８１０の各々は、上記の図８に関連して

説明された例では独立して復号可能である。他の実装形態では、量子化器５１２、５２８、５５２、および５６８の出力がどのようにパックされ、ビットストリームに符号化されるかに応じて、復号ピクチャ８４０、８３０、８２０、および８１０のうちの少なくとも１つは他に対して独立して復号可能である。たとえば、復号されたピクチャ８４０、８３０、８２０、および８１０の独立および／または同時復号は、単一のデコーダが復号されたピクチャ８４０、８３０、８２０、および８１０を順次デコードすることによって達成可能なものよりも高いフレームレートでの動作を容易にするために実行され得る。例えば、宛先デバイス１４０がクラウド内のサーバに実装されるとき、復号パフォーマンスは特に、ハードウェアモジュールがより計算上要求の厳しい動作をオフロードするために利用可能でないとき、プロセッサ２０５内の複数のコアにわたって分散されることから利益を得ることができる。

【０１３２】

様々な基底ベクトル（８２０）および関連する係数（８３０および８１０）の形態のテンソル１４９の縮小次元表現は、ベクトルおよび係数をチャネルごとにピクチャまたはサブピクチャにパッキングするときに、縮小されたエリアを可能にする。パッキングされたデータの削減されたエリアは、テンソル１１５のパッキングされた特徴マップまたはチャネルを含むフレームを直接圧縮しようと試みることと比較して、より効率的な圧縮表現をもたらす。

【０１３３】

図９に、ビデオデコーダとも呼ばれるピクチャデコーダ８０４の例示的な実装９００を示す。図９のビデオデコーダ８０４は汎用ビデオコーディング（ＶＶＣ）ビデオ復号パイプラインの一例であるが、本明細書で説明する処理ステージ、たとえば、ＨＥＶＣなどを実行するために、他のビデオコーデックも使用され得る。図９に示されるように、ビットストリーム９１０（ビットストリーム１４３に対応する）は、ビデオデコーダ８０４に入力される。ビットストリーム１４３は、メモリ２０６、ハードディスクドライブ２１０、ＣＤ－ＲＯＭ、ブルーレイディスク™、または他の非一時的コンピュータ可読記憶媒体から読み出され、ビットストリーム９１０として実施態様９００に提供され得る。あるいは、ビットストリーム９１０が通信ネットワーク２２０または無線周波数受信機に接続されたサーバなどの外部ソースから受信されてもよい。ビットストリーム１４３は、復号されるべきキャプチャされたフレームデータを表す符号化されたシンタックス要素を含む。サブピクチャが独立して復号される場合、各サブピクチャに対応するビットストリーム１４３の部分は、実装形態９００の別個のインスタンスに供給され得る。各サブピクチャのための実装９１０の別個のインスタンスは、スループットを改善するためのサブピクチャの並列復号を可能にする。

【０１３４】

ビットストリーム９１０は、エントロピーデコーダモジュール９２０に入力される。エントロピーデコーダモジュール９２０は「ビン」のシーケンスを復号することによってビットストリーム９１０からシンタックス要素を抽出し、シンタックス要素の値をビデオデコーダ８０４内の他のモジュールに渡す。エントロピーデコーダモジュール９２０は、可変長および固定長復号を使用して、ＳＰＳ、ＰＰＳ、またはスライスヘッダ、算術復号エンジン、を復号し、スライスデータのシンタックス要素を１つまたは複数のビンのシーケンスとして復号する。各ビンは１つまたは複数の「コンテキスト」を使用することができ、コンテキストは、ビンの「１」値および「０」値を符号化するために使用される確率レベルを記述する。所与のビンについて複数のコンテキストが利用可能である場合、「コンテキストモデリング」または「コンテキスト選択」ステップが実行されて、ビンを復号するために利用可能なコンテキストのうちの１つが選択される。ビンを復号するプロセスは順次フィードバックループを形成し、したがって、各スライスは、所与のエントロピーデコーダ９２０のインスタンスによってスライス全体において復号され得る。単一の（または少数の）高性能エントロピーデコーダ９２０インスタンスは、ビットストリーム９１０からのフレームまたはピクチャのすべてのスライスまたはサブピクチャを復号することが

できる、複数の低性能エントロピーデコーダ９２０インスタンスは、ビットストリーム９１０からのフレームのスライスを同時に復号することができる。

【０１３５】

エントロピーデコーダモジュール９２０は、算術符号化アルゴリズム、例えば「コンテキスト適応型バイナリ算術符号化」（ＣＡＢＡＣ）を適用して、ビットストリーム９１０からシンタックス要素を復号する。復号されたシンタックス要素は、ビデオデコーダ８０４内のパラメータを再構成するために使用される。パラメータは、残差係数（矢印９２４によって表される）、量子化パラメータ９７４、二次変換インデックス９７０、およびイントラ予測モード（矢印９５８によって表される）などのモード選択情報を含む。モード選択情報はまた、動きベクトル、および各ＣＴＵの１つまたは複数のＣＢへのパーティショニングなどの情報を含む。パラメータはＰＢを生成するために使用され、典型的には以前に復号されたＣＢからのサンプルデータと組み合わせられる。

【０１３６】

残差係数９２４は、逆二次変換モジュール９３６に渡され、二次変換インデックスに従って、二次変換が適用されるか、または動作が実行されない（バイパス）。逆二次変換モジュール９３６は、二次変換領域係数から、再構成された変換係数９３２、すなわち一次変換領域係数を生成する。再構成された変換係数９３２は、逆量子化器モジュール９２８に入力される。逆量子化器モジュール９２８は、量子化パラメータ９７４に従って、矢印９４０によって表される、再構成された中間変換係数を作成するために、残差係数９３２に対して、すなわち、一次変換係数領域において、逆量子化（または「スケーリング」）を実行する。逆量子化器モジュール９２８はまた、スケーリング行列を適用して、逆量子化器モジュール７４０の動作に対応する、ＴＢ内の不均一な逆量子化を提供し得る。不均一逆量子化行列を使用することがビットストリーム１４３に示される場合、ビデオデコーダ８０４はビットストリーム１４３から量子化行列をスケーリングファクタのシーケンスとして読み出し、スケーリングファクタを行列に配置する。逆スケーリングは、量子化パラメータと組み合わせて量子化行列を使用して、再構成された中間変換係数９４０を作成する。

【０１３７】

再構成された変換係数９４０は、逆一次変換モジュール９４４に渡される。モジュール９４４は、係数９４０を周波数領域から空間領域に戻すように変換する。逆一次変換モジュール９４４は、順方向一次変換モジュール７２６を参照して説明したように、最大利用可能変換サイズによって制約された、逆ＤＣＴ－２変換を水平および垂直に適用する。モジュール９４４の動作の結果は、矢印９４８によって表される残差サンプルのブロックである。残差サンプル９４８のブロックのサイズは、対応するＣＢに等しい。残差サンプル９４８は、加算モジュール９５０に供給される。

【０１３８】

加算モジュール９５０において、残差サンプル９４８は、復号されたＰＢ（９５２として表される）に加算されて、矢印９５６によって表される、再構成されたサンプルのブロックを生成する。再構成されたサンプル９５６は、再構成されたサンプルキャッシュ９６０およびループ内フィルタリングモジュール９８８に供給される。ループ内フィルタリングモジュール９８８は、９９２として表される、フレームサンプルの再構成されたブロックを生成する。フレームサンプル９９２は、フレームバッファ９９６に書き込まれる。フレームバッファ９９６は、図１のテンソル１４９に対応する画像またはビデオフレーム９１４を出力する。

【０１３９】

再構成されたサンプルキャッシュ９６０は、ビデオエンコーダ６１４の再構成されたサンプルキャッシュ７５６と同様に動作する。再構成されたサンプルキャッシュ９６０は、メモリ２０６なしで後続のＣＢをイントラ予測するのに必要な再構成されたサンプルのための記憶を提供する（例えば、典型的にはオンチップメモリであるデータ２３２を代わりに使用することによって）。矢印９６４によって表される参照サンプルは、再構成された

サンプルキャッシュ９６０から取得され、参照サンプルフィルタ９６８に供給されて、矢印９７２によって示されるフィルタリングされた参照サンプルを生成する。フィルタリングされた参照サンプル９７２は、イントラフレーム予測モジュール９７６に供給される。モジュール９７６は、ビットストリーム９１０においてシグナリングされ、エントロピーデコーダ９２０によって復号されたイントラ予測モードパラメータ９５８に従って、矢印９８０によって表されるイントラ予測サンプルのブロックを生成する。イントラ予測モジュール９７６は、ＩＢＣおよびＭＩＰを含むエンコーダサイドのモジュール７６４のモードをサポートする。サンプル９８０のブロックは、ＤＣ、平面、または角度イントラ予測などのモードを使用して生成される。

【０１４０】

ＣＢの予測モードがビットストリーム１４３においてイントラ予測を使用するように示されるとき、イントラ予測サンプル９８０は、マルチプレクサモジュール９８４を介して復号ＰＢ９５２を形成する。イントラ予測はサンプルの予測ブロック（ＰＢ）を生成し、これは、同じ色成分における「隣接サンプル」を使用して導出される、１つの色成分におけるブロックである。隣接サンプルは現在のブロックに隣接するサンプルであり、ブロック復号順序において先行しているため、すでに再構成されている。ルマブロックとクロマブロックとがコロケートされる場合、ルマブロックとクロマブロックとは異なるイントラ予測モードを使用し得る。しかしながら、２つのクロマＣＢは、同じイントラ予測モードを共有する。

【０１４１】

ＣＢの予測モードがビットストリーム９１０内のインター予測であると示されるとき、動き補償モジュール９３４は、９３８として表されるインター予測サンプルのブロックを生成する。インター予測サンプル９３８のブロックは、エントロピーデコーダ９２０によってビットストリーム１４３から復号された動きベクトルと、フレームバッファ９９６からサンプルのブロック９９８を選択およびフィルタリングするための参照フレームインデックスとを使用して生成される。サンプルのブロック９９８は、フレームバッファ９９６に記憶された以前に復号されたフレームから取得される。双予測の場合、サンプルの２つのブロックが生成され、一緒にブレンドされて、復号されたＰＢ９５２のためのサンプルが生成される。フレームバッファ９９６は、ループ内フィルタリングモジュール９８８からのフィルタリングされたブロックデータ９９２でポピュレートされる。ビデオエンコーダ６１４のループ内フィルタリングモジュール７６８と同様に、ループ内フィルタリングモジュール９８８は、ＤＢＦ、ＡＬＦ、およびＳＡＯフィルタリング動作のいずれかを適用する。一般に、動きベクトルは、ルマチャネルとクロマチャネルの両方に適用されるが、ルマチャネルとクロマチャネルにおけるサブサンプル補間のためのフィルタリング処理は異なる。

【０１４２】

図７および図９には示されていないが、各クロマチャネル内のサンプル値の範囲のより均一な使用が達成されるようにサンプル値をシフトするために、符号化前にビデオを前処理し、復号後にビデオを後処理するためのモジュールである。マルチセグメント線形モデルは、ビデオエンコーダ６１４において導出され、サンプルシフティングを取り消すためにビデオデコーダ８０４によって使用するためにビットストリームにおいてシグナリングされる。この線形モデルクロマスケーリング（ＬＭＣＳ）ツールは、量子化のアプリケーションからより高い品質損失をもたらし得るサンプル空間の利用において、ある程度の不均一性、特に限られた範囲の利用を有する特定の色空間およびコンテンツに対して圧縮利点を提供する。

【０１４３】

図１０Ａは、「ＹＯＬＯｖ３」ネットワークの「ＤａｒｋＮｅｔ－５３」バックボーン部分を除く部分に対応する、オブジェクト検出のためのＣＮＮのヘッド部分１５０の例示的な実装１０００を示す概略ブロック図である。実装１０００は、例えば図３Ａのようにバックボーンが実装される場合に使用することができる。宛先デバイス１４０にお

いて実行されるべきタスクに応じて、ＣＮＮヘッド１５０の代わりに異なるネットワークを使用することができる。入力テンソル１４９は、各レイヤのテンソル（すなわち、テンソル１０１０、１０２０、および１０３４）に分離される。テンソル１０１０は、ＣＢＬモジュール１０１２に渡されてテンソル１０１４を生成する。テンソル１０１４は、検出モジュール１０１６およびアップスケーラモジュール１０２２に渡される。検出モジュールは、検出テンソルの形成でバウンディングボックス１０１８を出力する。バウンディングボックス１０１８は、非最大抑制（ＮＭＳ）モジュール１０４８に渡される。

【０１４４】

オリジナルのビデオデータ１１３内の座標を示すバウンディングボックスを生成するために、ネットワーク１１４のバックボーン部分のサイズ変更の前に、オリジナルのビデオ幅および高さによるスケーリングが、アップスケーラモジュール１０２２において実行される。アップスケーラモジュール１０２２は、テンソル１０１４およびテンソル１０２０を受信し、ＣＢＬモジュール１０２６に渡されるアップスケールされたテンソル１０２４を生成する。ＣＢＬモジュール１０２６は、出力としてテンソル１０２８を生成する。テンソル１０２８は、検出モジュール１０３０およびアップスケーラモジュール１０３６に渡される。検出モジュール１０３０は、ＮＭＳモジュール１０４８に供給される検出テンソル１０３２を生成する。アップスケーラモジュール１０３６は、モジュール１０２０のもう１つの例である。アップスケーラモジュール１０３６は、テンソル１０２８およびテンソル１０３４を受信し、アップスケールされたテンソル１０３８を出力する。アップスケールされたテンソル１０３８は、ＣＢＬモジュール１０４０に渡され、ＣＢＬモジュール１０４０はテンソル１０４２を検出モジュール１０４４に出力する。ＣＢＬモジュール１０１２、１０２６、および１０４０はそれぞれ、５つのＣＢＬモジュール、例えば、図３Ｄに示されるＣＢＬモデル３６０の連結を含む。アップスケーラモジュール１０２２及び１０３６は、図１０Ｂに示すように、アップスケーラモジュール１０６０の各インスタンスである。モジュール１０４８は、テンソル１０１８、１０３２、および１０３６を受け取り、タスク結果１５１を出力する。

【０１４５】

図１０Ｂに示すように、アップスケーラモジュール１０６０は、テンソル１０６２（例えば、図１０Ａのテンソル１０１４）を入力として受け付ける。テンソル１０６２は、ＣＢＬモジュール１０６６に渡されてテンソル１０６８を生成する。テンソル１０６８は、アップサンプリングされたテンソル１０７２を生成するために、アップサンプラ１０７０に渡される。連結モジュール１０７４は、アップサンプリングされたテンソル１０７２を第２入力テンソル１０６４（例えば、図１０Ａのアップスケーラ１０２２へ入力されるテンソル１０２０）と連結することによって、テンソル１０７６を生成する。

【０１４６】

検出モジュール１０１６、１０３０、および１０４４は、図１０Ｃに示されるような検出モジュール１０８０のインスタンスである。検出モジュール１０８０は、テンソル１０８２を受信する。テンソル１０８２は、ＣＢＬモジュール１０８４に入力される。ＣＢＬモジュール１０８４は、テンソル１０８６を生成する。テンソル１０８６は、検出カーネルを実装する畳み込みモジュール１０８８に渡される。いくつかの構成では、検出カーネルが１×１カーネルを適用して、テンソルの３つのレイヤの各々における特徴マップ上に出力を生成する。検出カーネルは１×１×（Ｂ×（５＋Ｃ））であり、ここで、Ｂは特定のセルが予測することができるバウンディングボックスの数であり、典型的には３であり、Ｃは８０であり得るクラスの数であり、２５５の検出属性（すなわち、テンソル１０９０）のカーネルサイズをもたらす。定数「５」は、４つのバウンディングボックス属性（ボックス中心ｘ、ｙおよびサイズスケールｘ、ｙ）および１つのオブジェクト信頼レベル（「オブジェクトネス」）を表す。検出カーネルの結果は、入力特徴マップと同じ空間次元を有するが、出力の深さは検出属性に対応する。検出カーネルは、各レイヤ、典型的には３つのレイヤに適用され、多数の候補バウンディングボックスをもたらす。非最大抑制のプロセスは、ＮＭＳモジュール１０４８によって、結果として得られるバウンディング

ボックスに適用され、類似のスケールでの重複予測などの冗長ボックスを破棄し、結果として、オブジェクト検出のための出力としてバウンディングボックスの最終セットをもたらす。

【０１４７】

図１１は、ＣＮＮの代替ヘッド部分１１００を示す概略ブロック図である。ヘッド部分１１００は、ＣＮＮヘッド１５０として実装することができ、ＣＮＮバックボーン１１４は、例えばバックボーン４００として実装される。ヘッド部分１１００は「高速ＲＣＮＮ」として知られる全体的なネットワークの一部を形成し、特徴ネットワーク（すなわち、バックボーン部分４００）、リージョンプロポーザルネットワーク（a region proposal network）、および検出ネットワークを含む。ヘッド部分１１００への入力は、Ｐ２－Ｐ６レイヤテンソル１１１０、１１１２、１１１４、１１１６、および１１１８を含むテンソル１４９である。Ｐ２－Ｐ６レイヤテンソル１１１０、１１１２、１１１４、１１１６、および１１１８は、図４のＰ２～Ｐ６出力４７７、４７５、４７３、４７１、および４２９に対応する。Ｐ２－Ｐ６テンソル１１１０、１１１２、１１１４、１１１６、および１１１８は、リージョンプロポーザルネットワーク（ＲＰＮ）ヘッドモジュール１１２０に入力される。ＲＰＮヘッドモジュール１１２０は、入力テンソルに対して畳み込みを実行し、中間テンソルを生成する。中間テンソルは、２つの後続の兄弟レイヤ、（ｉ）分類のための１つと、（ｉｉ）バウンディングボックス、または「注目領域」（ＲＯＩ）回帰のための１つと、に供給される。結果として得られる出力は、分類およびバウンディングボックス１１２２である。分類およびバウンディングボックス１１２２は、ＮＭＳモジュール１１２４に渡される。ＮＭＳモジュール１１２４は、より低いスコアを有する重複ボックスを除去することによって、冗長なバウンディングボックスを枝刈りして、枝刈りされたバウンディングボックス１１２６を生成する。バウンディングボックス１１２６は、注目領域（ＲＯＩ）プーラ１１２８に入力される。ＲＯＩプーラ１１２８は、テンソル１４９のレイヤテンソル（以下でさらに説明する）およびバウンディングボックス１１２６のいくつかを使用して、最大プーリング動作を使用して様々な入力サイズマップから固定サイズ特徴マップを生成する。最大プーリング動作では、サブサンプリングが入力値の各グループ内の最大値を取得して、出力テンソル内に１つの出力値を生成する。

【０１４８】

ＣＮＮバックボーン４００およびＣＮＮヘッド１１００の構成では、「Ｐ６」レイヤテンソル４２９が（テンソル１４９として受信される）出力テンソル１１５から省略される。Ｐ６テンソルが省略される構成では、ＣＮＮヘッド１１００がＰ５テンソル１１１６に対して２に等しいストライドを有する「Ｍａｘｐｏｏｌ」動作を実行することによって、Ｐ６入力テンソル１１１８を生成する。Ｐ６レイヤは、Ｐ５レイヤから再構成することができるので、ＰＣＡエンコーダ１６０、ＰＣＡデコーダ１７０において、明示的なＦＰＮレイヤとしてＰ６レイヤを個別に符号化および復号する必要がない。

【０１４９】

ＲＯＩプーラ１１２８への入力は、Ｐ２－Ｐ５特徴マップ１１１０、１１１２、１１１４、および１１１６、ならびに注目領域提案１１２６である。１１２６からの各提案（ＲＯＩ）は、固定サイズのマップを生成するために、特徴マップ（１１１０～１１１６）の一部分に関連付けられる。固定サイズのマップは、特徴マップ１１１０～１１１６の基礎となる部分とは無関係のサイズである。特徴マップ１１１０～１１１６のうちの１つは、結果として得られるクロップされたマップが例えば、以下のルールに従って十分な詳細を有するように選択される:ｆｌｏｏｒ（４＋ｌｏｇ２（ｓｑｒｔ（ｂｏｘ＿ａｒｅａ）／２２４））、ここで２２４は基準のボックスサイズである。ＲＯＩプーラ１１２８は、テンソル１１３０を生成する提案１１２６に従って、入力特徴マップをクロップするように動作する。

【０１５０】

テンソル１１３０は、全結合（ＦＣ）ニューラルネットワークヘッド１１３２に供給される。ＦＣヘッド１１３２は、２つの全結合レイヤを実行して、クラススコアおよびバウ

ンディングボックス予測子デルタテンソル１１３４を生成する。クラススコアは一般に、８０要素テンソルであり、各要素は、対応するオブジェクトカテゴリの予測スコアに対応する。バウンディングボックス予測デルタテンソルは８０×４ ＝ ３２０要素テンソルであり、対応するオブジェクトカテゴリのバウンディングボックスを含む。最終処理は出力レイヤモジュール１１３６によって実行され、テンソル１１３４を受信し、フィルタリング演算を実行して、フィルタリングされたテンソル１１３８を生成する。低スコアリング（低分類）オブジェクトは、さらなる考慮から除外される。非最大抑制モジュール１１４０は、フィルタリングされたテンソル１１３８を受信し、より低い分類スコアを有する重複ボックスを除去することによって重複するバウンディングボックスを除去し、テンソル１５１に対応する推論出力テンソル１１４０をもたらす。

【０１５１】

図１２Ａおよび図１２Ｂはたとえば、サブピクチャビットストリームコンバイナ５８０によって実装される、ピクチャ１２００をサブピクチャ１２１０、１２１２、１２１４、および１２１６に分割することを示す概略ブロック図である。各サブピクチャは、エンコーダ５１４、５３２、５５６、および５７２のうちの対応する１つのパッカ６１０によってパックされる。各サブピクチャは、サンプルの２次元アレイに配列された情報を含む。図１２Ｂを参照すると、サブピクチャ１２１０は、ＦＰＮのテンソルにおけるテンソルの平均特徴マップ１２２０などの平均特徴マップを保持する。平均特徴マップ１２２０は、図５の構成におけるサブピクチャエンコーダ５１４のビットストリーム５１５に対応する。サブピクチャ１２１２は、図５の５３３に対応する平均係数１２２２を保持する。平均係数１２２２は、符号化レイヤまたはＦＰＮレイヤごとに１つの行と、対応するテンソルのチャネルごとに１つの値とを有する値の行に配列される。符号化レイヤの数および次元は、図１８Ａ、１８Ｂ、１９、および２０を参照して説明したように、テンソル再サンプリングおよび／または連結に基づいてＦＰＮレイヤの数および次元とは異なり得る。サブピクチャ１２１６は、図５の５５７に対応する基底ベクトルを保持する。図１２Ｂの例では、基底ベクトルがとりわけ、基底ベクトル１２２６を含み、基底ベクトルは重複しない方法でサブピクチャ１２１６のエリアにパックされる。基底ベクトルは隣接してパックされてもよく、または、例えば、基底ベクトルの各セット間の１つまたは２つのサンプルの間隔を含んでもよい。各符号化レイヤまたはＦＰＮレイヤの基底ベクトルは、サブピクチャ１２１６にパックされる。ピクチャ１２００内の、１２０８などのデータを記憶するために使用されないエリアは、サンプル値から浮動小数点領域に変換するための逆量子化の適用後に、値「０」に対応するサンプル値によって占有され得る。同様に、サブピクチャ１２１０、１２１２、１２１４、および１２１６内の、データを記憶するために使用されないエリアは、サンプル値から浮動小数点領域に変換するための逆量子化の適用後に、値「０」に対応するサンプル値によって占有され得る。サブピクチャ１２１４は、図５の５７２に対応する係数を保持し、ＦＰＮのテンソルにおけるテンソル内の各チャネルの各基底ベクトルに対して１つの係数を有する。例えば、テンソル内の２５６個のチャネルでは２５個の基底ベクトル、１つのテンソルに対する係数は２５６×２５個のサンプルアレイとして配置される。

【０１５２】

複数のテンソルがピクチャ１２００において符号化される場合、対応するサンプルアレイは、サブピクチャ１２１４において積み重ねられる。量子化された特徴マップはサンプルの２次元アレイであり、係数もまた、サンプルの１次元または２次元アレイにマッピングされるので、ピクチャ１２００のためのモノクロフォーマットの使用は十分である。ＣＮＮバックボーン１１４がフレーム１１３のためのテンソル１１５を生成したとき、ＰＣＡエンコーダ１６０は、テンソル１１５の次元に基づいてサブピクチャ１２１０、１２１２、１２１４、および１２１６のサイズを決定することができる。サブピクチャ１２１０、１２１２、および１２１６はＣＴＵの整数として、すなわち、１２８ルマサンプルの倍数として、幅および高さにおいて指定され得る。サブピクチャ１２１６の幅は、水平方向に配置されたサブピクチャ１２１０、１２１２、および１２１４の幅の合計に等しい。サ

ブピクチャ１２１０、１２１２、および１２１６の下に配置されたサブピクチャ１２１６の高さは、パックされた基底ベクトル、たとえば基底ベクトル１２２６に十分なエリアを提供するように設定される。

【０１５３】

図１２Ａおよび１２Ｂの例では、量子化係数５１３、５３０、および５６８にそれぞれ対応するサブピクチャ１２１０、１２１２、１２１４が水平に配置され、ピクチャ１２００の幅を定義する。量子化係数５５４に対応するサブピクチャ１２１６は、サブピクチャ１２１０、１２１２、１２１４の幅にわたり、ピクチャ１２００の底部まで延在する、第１、第２、および第３ユニットの下に位置する。別の実装形態では、量子化係数がサブピクチャ以外のフォーマットで構成され得る。たとえば、ＶＶＣまたはＨＥＶＣの場合、量子化係数は、スライスまたはタイルとして配置またはパックされ得る。さらに他の実装形態では、量子化係数が符号化タイプ（たとえば、ＶＶＣまたはＨＥＶＣ）および予想されるデータ特性などの要因に応じて、異なるように構成され得る。たとえば、量子化係数は、変換が使用されないときに量子化係数の順序付けられたセットに残差符号化が適用されるように、変換スキップが適用された小さいブロックの走査順序で配列され得る。たとえば、量子化係数がＨＥＶＣを使用して圧縮されるとき、変換スキップが４×４ブロックに適用され得、量子化係数は４×４ブロックへの変換の適用なしに残差符号化を活用するために、４×４ブロック中の対角走査パテン中に構成され得る。基底ベクトルの数が１６以下である場合、各特徴マップの量子化係数は、別個の４×４ブロックに記憶され得る。各４×４ブロック内で、量子化係数の圧縮効率は、４×４ブロック内の適応ライスパラメータ符号化の適用から恩恵を受ける。別個のサブピクチャの代わりに、量子化係数の各セットに対応する別個のスライスを用いて、単一のピクチャを他の実装形態で使用することができ、または各サブピクチャをピクチャ１２００とは異なるように配置することができる。

【０１５４】

ピクチャ１２００の構成では、係数１２２２のサブピクチャ１２１２へのパッキングがＮがチャネルの数であるＮ×１アレイからＷ×Ｈアレイにアレイを修正し、ここで、Ｗ×ＨはＮに等しく、サブピクチャのアスペクト比を、可能な限り正方形（１：１）または２：１比などの長方形に近くなるように調整するが、Ｎ×１の極端に細長い比を回避するように選択される。正方形に近いアスペクト比は、パックされた係数サンプルのエッジに隣接するサンプルが少なくなり、アンパッカによって使用されないが、依然として比較的高い忠実度で符号化され、したがって、テンソル再構成に何ら利益を及ぼすことなく残差符号化コストに寄与する。

【０１５５】

ピクチャ１２００の別の構成では、サブピクチャ１２１２または１２１４への係数１２２２または１２２４のパッキングが、各係数が２つ以上のサンプルにパッキングされるように実行される。たとえば、係数は、パッカ６１０によって２×２のサンプルのセットにパッキングされ得る。アンパッカ６２０または８３２または８１２は、２×２セットのサンプルを読み取り、フィルタリング動作を実行して係数を復元する。２つ以上のサンプルへのパッキングは、ビデオエンコーダ６１４における非可逆符号化の使用に起因して係数値に導入される誤差を補償するための機構を提供する。２×４、４×２、４×４を含む、サンプルのセットの他の構成も可能である。

【０１５６】

図１３は、符号化されたパッキングされた特徴マップおよび関連するメタデータを保持するビットストリーム１３００を示す概略ブロック図である。ビットストリーム１３００は、ＰＣＡエンコーダ１６０によって生成されたビットストリーム１２１またはＰＣＡデコーダ１７０によって復号されたビットストリーム１４３に対応する。ビットストリーム１３００は、「ネットワーク抽象化レイヤ」ユニットヘッダによって前置されたシンタックスのグループを含む。例えば、ＮＡＬユニットヘッダ１３０８は、シーケンスパラメータセット（ＳＰＳ）１３１０に先行する。ＳＰＳ１３１０はプロファイル（符号化ツール

のセット）およびレベルおよびティアを指定する（たとえば、最大サンプルレートおよび圧縮ビットレートに関して動作点を共同で指定する）シンタックスの「プロファイルレベルティア」（ＰＬＴ）ユニットを含み得る。シンタックスのＰＬＴユニットは、選択されたプロファイルと比較して利用可能なツールのセットをさらに制約するシンタックスの「一般制約情報」（ＧＣＩ）ユニットを含み得る。ＳＰＳ１３１０は、サブピクチャ１２１０、１２１２、１２１４、および１２１６の位置およびサイズを含む、ピクチャ１２００のレイアウトを指定する。ＧＣＩはフラグのセットを含み、各フラグは、ビットストリーム１３００において使用されない特定の符号化ツールを制約する。ＰＬＴは、ツールの特定のセットがビットストリーム１３００において使用可能であることをシグナリングし、ツールの特定のセットは「プロファイル」として知られている。プロファイルの一例は「Ｍａｉｎ　１０」であり、４：０：０または４：２：０クロマフォーマットのいずれかを有する８～１０ビットビデオを提供し、広範な展開をターゲットとする。ＧＣＩは、「サブプロファイル」として知られる、ツールのサブセットへのプロファイルのツールのセットに対するさらなる制約を示し得る。ＰＣＡエンコーダ１６０は、フレームにパッキングされた特徴マップを符号化する（すなわち、図５に従って動作する）際に、ＶＶＣ規格の特定のツールは、圧縮の利点を提供しない。パッキングされた特徴マップに圧縮利点を提供しないツールは、ビデオエンコーダ６１４によって試される必要はなく、ビットストリーム１３００において使用されないものとしてＧＣＩにおいてシグナリングされ得る。ＳＰＳ１３１０はまた、ビットストリーム１３００によって表されるフレームデータの解像度、ビット深度、クロマフォーマット、を示す。

【０１５７】

ＰＣＡエンコーダ１６０によって使用されるパッキングフォーマットは、すべての利用可能な特徴パッキングフォーマットの列挙から１つの特徴パッキングフォーマットを選択するためのインデックスを使用して、ＳＥＩメッセージ１３１３において符号化され得る。特徴マップを生成するために使用された特定のＣＮＮバックボーンは、ＳＥＩメッセージ１３１３において、インデックスを使用して、１組の所定のＣＮＮバックボーンの列挙から１つのＣＮＮバックボーンを選択することも示され得、そのうちのいくつかまたはすべてがソースデバイス１１０に利用可能である。ＣＮＮバックボーンタイプインデックスから、各レイヤにおけるチャネルの数およびレイヤの数、ならびに各レイヤにおける各特徴マップの解像度が決定され得る。基底ベクトル５５０のサイズ（すなわち、幅および高さ）、基底ベクトル５５０の最大個数（すなわち、ＦＰＮの各レイヤの最大個数、またはテンソルコンバイナ１６２から得られるＦＰＮの縮小形成）、および使用された基底ベクトル５５０の数（すなわち、方法２１００から得られる）は、付録Ａを参照して説明される、ＳＥＩメッセージ１３１３内の基底ベクトルパッキング情報１３９１として符号化される。テンソル抽出器１７２内で、テンソル１１５ａと同じ数および次元数を有するテンソル１４９を復元するために（テンソルコンバイナ１６２によって決定される）実行される動作は、レイヤマッピング１３９０として説明される。レイヤマッピング１３９０は、ＳＥＩメッセージ１３１３内に存在する。レイヤマッピング１３９０は、付録Ａを参照して説明した、ｌａｙｅｒ＿ｕｐｄａｔｅ＿ｍａｐｐｉｎｇ＿ｉｎｆｏ、ｌａｙｅｒ＿ｅｘｔｒａｃｔｉｏｎ＿ｆｌａｇ、ｌａｙｅｒ＿ｕｐｓａｍｐｌｅ＿ｆｌａｇ、ｌａｙｅｒ＿ｄｏｗｎｓａｍｐｌｉｎｇ＿ｆｌａｇ、ｒｅｓａｍｐｌｉｎｇ＿ｆｉｌｔｅｒ＿ｉｄｘ、ｒｅｓａｍｐｌｅ＿ｖｅｒｔｉｃａｌ＿ｏｎｌｙ＿ｆｌａｇ、ｒｅｓａｍｐｌｅ＿ｈｏｒｉｚｏｎｔａｌ＿ｏｎｌｙ＿ｆｌａｇ、ｓｒｃ＿ｌａｙｅｒ＿ｉｄｘ，ｓｒｃ＿ｃｈａｎｎｅｌ＿ｏｆｆｓｅｔ，ｄｓｔ＿ｌａｙｅｒ＿ｉｄｘを含む。

【０１５８】

ピクチャ１３１４は、ビットストリーム１３００において符号化される。各ピクチャは、符号化サブピクチャ１３２０、符号化サブピクチャ１２１０などの１つまたは複数のサブピクチャを含む。ビットストリームの第１ピクチャについて、一般に「ランダムアクセスポイント」アクセスユニットについて、イントラスライスは、ビットストリーム１３００中の他のアクセスユニットに対するいかなる予測依存性も回避するために使用される。

サブピクチャ１２１２を符号化する符号化サブピクチャ１３２２は、スライスヘッダ１３３０と、それに続くスライスデータ１３４０とを含む。スライスデータ１３４０は、フレームデータの符号化表現を与えるＣＴＵのシーケンスを含む。ＣＴＵは正方形であり、典型的には１２８×１２８のサイズであり、これは典型的な特徴マップサイズに十分に整合されない。特徴マップの４×４グリッドなどの最小ブロックサイズへのアライメントは、この不整合を部分的に改善する。整数またはサンプル領域から浮動小数点領域への変換を可能にするために、量子化範囲１３９２がビットストリーム１３００に存在し、浮動小数点の最小値および最大値を定義し、そこから、例えば、５２０、５３６、５６０、８４６、８３６、８２４、８１６において実行されるように、逆量子化プロセスにおいてサンプルがマッピングされる。量子化範囲は、量子化モジュール、すなわち、５１２、５２８、５５２、および５６８において決定され、ビットストリーム１２１において、ＳＥＩメッセージ１３１３における量子化範囲１３９２として符号化される。符号化サブピクチャ１３２４は、サブピクチャ１２１４を符号化し、符号化サブピクチャ１３２６は、サブピクチャ１２１６を符号化する。符号化サブピクチャ１２２０、１２２２、１２２４、および１２２６は、それぞれ、ビットストリーム部分５１５、５３３、５５７、および５７３に対応する。別個のサブピクチャの代わりに、単一のピクチャを、サブピクチャ１３２０、１３２２、１３２４、および１３２６に対応する別個のスライスとともに使用することもできる。

【０１５９】

図１４は、ＣＮＮの第１部分を実行し、ビデオデータのフレームのために結果として得られる特徴マップを符号化するための方法１４００を示す。特徴マップを符号化する際に、テンソルがビットストリームに符号化される。方法１４００は、構成されたＦＰＧＡ、ＡＳＩＣ、またはＡＳＳＰなどの装置を使用して実装され得る。代替的に、以下で説明するように、方法１４００は、プロセッサ２０５の実行下で、アプリケーションプログラム２３３の１つまたは複数のソフトウェアコードモジュールとして、ソースデバイス１１０によって実装され得る。方法１４００を実施するアプリケーションプログラム２３３のソフトウェアコードモジュールは例えば、ハードディスクドライブ２１０および／またはメモリ２０６に常駐することができる。方法１４００は、ビデオソース１１２によって生成されたビデオデータの各フレームに対して繰り返される。方法１４００は、コンピュータ可読記憶媒体および／またはメモリ２０６に記憶され得る。

【０１６０】

方法１４００は、ＣＮＮ第１部分実行ステップ１４１０で開始する。ステップ１４１０において、ＣＮＮバックボーン１１４は、プロセッサ２０５の実行下で、特定のＣＮＮのレイヤのサブセットを実行して、入力フレーム１１３を中間テンソル１１５ａに変換する。中間テンソル１１５ａは例えば、メモリ２０６および／またはハードディスクドライブ２１０に記憶され得る。ＣＮＮの例は図４に示されるように、「より速いＲ－ＣＮＮ」または「マスクＲ－ＣＮＮ」であり、レイヤのサブセットは、「Ｐレイヤ」分割点までのすべてのレイヤに対応する。例えば、ＦＰＮの使用のために、ＣＮＮバックボーン１１４から複数のテンソルが抽出されるとき、テンソル１１５ａは、複数のテンソルを含み得る。方法１４００は、ビデオソース１１２からのビデオデータの１つのフレームに対応するテンソルを符号化するように動作する。プロセッサ２０５における制御は、ステップ１４１０からテンソル低減実行ステップ１４１５に進む。

【０１６１】

ステップ１４１５において、テンソルコンバイナ１６２は、プロセッサ２０５の実行下で動作して、テンソルの数を再サンプリングする。図１９は、ステップ１４１５で実施される方法１９００を示す。ステップ１４１５の動作の結果として、ＦＰＮからのテンソル１１５ａは、（ｉ）ダウンサンプリングされ（数が削減され）、（ｉｉ）アップサンプリングされ（数が増加され）、または（ｉｉｉ）直接通過されて、テンソル１１５を生成することができる。アップサンプリングまたはダウンサンプリングは、２つのレイヤからのテンソルが互換性のある幅および高さで連結されるように実行される。テンソル１１５ａ

の所与のテンソルをアップサンプリングまたはダウンサンプリングするか、またはすべてのテンソル１１５ａを変更せずに残すかの決定は、ビットストリーム１２１において符号化される。プロセッサ２０５における制御はステップ１４１５から、平均特徴マップ決定ステップ１４２０に進む。

【０１６２】

平均特徴マップ決定ステップ１４２０において、モジュール５１０は、プロセッサ２０５の実行下で、チャネル寸法にわたってテンソル１１５を平均化して、平均特徴５１１を生成する。ステップ１４２０において、平均特徴５１１は、プロセッサ２０５の実行下で量子化器５１２によって量子化され、整数（量子化）平均特徴マップ５１３を生成する。プロセッサ２０５における制御は、ステップ１４２０から平均特徴符号化ステップ１４３０に進む。

【０１６３】

平均特徴符号化ステップ１４３０において、サブピクチャエンコーダ５１４は、プロセッサ２０５の実行下で、整数平均特徴マップ５１３をサブピクチャ１２１０にパッキングし、サブピクチャ１２１０を符号化してビットストリーム部分５１５を生成する。プロセッサ２０５における制御は、ステップ１４３０から再構成平均特徴リカバーステップ１４４０に進む。

【０１６４】

再構成平均特徴リカバーステップ１４４０において、サブピクチャエンコーダ５１４は、プロセッサ２０５の実行下で、ビデオ圧縮のためのサブピクチャ入力の非可逆バージョン、たとえば６１２に対応する、エンコーダ５１４の実装形態内の再構成されたピクチャ、たとえば６１８を出力する。プロセッサ２０５における制御は、ステップ１４４０から平均係数決定ステップ１４５０に進む。

【０１６５】

平均係数決定ステップ１４５０において、逆量子化器５２０は、プロセッサ２０５の実行下で、整数平均特徴マップ５１６を逆量子化し、マップ５１６を特徴マップ５２２として浮動小数点領域に変換して戻すように動作する。ドット積５２４は、プロセッサ２０５の実行下で、テンソル１１５においてチャネルごとに生成された１つの平均係数を用いて、ドット積関数を使用して平均係数のセット５２６を生成するためにテンソル１１５および特徴マップ５２２を使用する。プロセッサ２０５における制御は、ステップ１４５０から平均係数符号化ステップ１４６０に進む。

【０１６６】

平均係数符号化ステップ１４６０において、量子化器モジュール５２８は、プロセッサ２０５の実行下で平均係数５２６ｍを量子化して、整数平均係数５３０を生成（ｇｅｎｅｒａｔｅ）または生成（ｐｒｏｄｕｃｅ）する。サブピクチャエンコーダ５３２は、プロセッサ２０５の実行下で、整数平均係数５３０をサブピクチャ１２１２にパッキングし、サブピクチャ１２１２を符号化してビットストリーム部分５３３を生成する。プロセッサ２０５における制御は、ステップ１４６０から再構成平均係数リカバーステップ１４７０に進む。

【０１６７】

再構成平均係数リカバーステップ１４７０において、サブピクチャエンコーダ５３２は、プロセッサ２０５の実行下で、サブピクチャ１２１２の再構成されたバージョンを出力し、そこから再構成された平均係数５３４が生成される。再構成された平均係数５３４は、モジュール５３６によって逆量子化されて、リカバーされた平均係数５３８を生成する。ステップ１４７０において、ピクチャの再構成されたバージョンは、エンコーダ５３２の実装内の６１８に対応する。プロセッサ２０５における制御は、ステップ１４７０から基底ベクトル決定ステップ１４８０に進む。

【０１６８】

基底ベクトル決定ステップ１４８０において、逆量子化器５３６、ドット積モジュール５４０、減算器５４４、および分解モジュール５４８は、プロセッサ２０５の実行下で動

作して、テンソル１１５のための基底ベクトルのセットを生成する。逆量子化器５３６は、浮動小数点平均係数５３８を出力する。ドット積モジュール５４０は、リカバーされた平均特徴マップ５２２およびｈｔｅ係数５３８を用いてドット積演算を実行し、オフセットテンソル５４２を生成する。テンソル５４２はテンソル１１５と同じ次元を有し、平均（平均）特徴マップ情報のみを使用して予測される、テンソル１１５内の各チャネルの特徴マップの予測を含む。したがって、テンソル５４２は、このチャネルごとの平均特徴マップデータを減算することによって、テンソル１１５のゼロ平均バージョンの生成を可能にする。オフセット特徴マップ５４２は、モジュール５４４によってテンソル１１５から減算されて、ゼロ中心テンソル５４６を生成する。分解モジュール５４８は、テンソル５４６を入力として受け取り、基底ベクトルのセット５５０を生成する。セット５５０中の各基底ベクトルは、「成分」と呼ばれることがある。図２１を参照して、基底ベクトル５５０を生成する際の分解モジュール５４８の動作を説明する。ベクトル５５０は、テンソル１１５内のチャネルよりも少ない基底ベクトルを含み、テンソル１１５の次元数の減少に対応する。５５０内の基底ベクトルは、基底ベクトル５５０内の成分の数に対するテンソル１１５内の最大分散量を説明する部分空間内のテンソル１１５を表す。言い換えれば、基底ベクトル５５０は、所与の数の成分についての品質の最小限の劣化でテンソル１１５の表現を可能にする。プロセッサ２０５における制御は、ステップ１４８０から基底ベクトル符号化ステップ１４９０に進む。

【０１６９】

基底ベクトル符号化ステップ１４９０において、量子化器モジュール５５２は、プロセッサ２０５の実行下で、基底ベクトル５５０を整数領域に量子化するように動作する。得られた整数基底ベクトル５５４は、サブピクチャにパッキングされ、符号化されて、プロセッサ２０５の実行下でサブピクチャエンコーダ５５６によってビットストリーム部分５５７を生成する。プロセッサ２０５における制御は、ステップ１４９０から再構成基底ベクトルリカバーステップ１４１００に進む。

【０１７０】

ステップ１４１００において、サブピクチャエンコーダ５５６は、プロセッサ２０５の実行下で、再構成された整数テンソル５５８を生成し、サブピクチャ１３１６の再構成されたバージョン、たとえば、６１８を取得するように動作する。基底ベクトル５５８は、再構成されたサブピクチャからアンパックされ、プロセッサ２０５の実行下で逆量子化器５６０によって（再構成された基底ベクトル５６２として）浮動小数点領域に逆量子化される。プロセッサ２０５における制御は、ステップ１４１００から係数決定ステップ１４１１０に進む。

【０１７１】

ステップ１４４０、１４７０、および１４１００の各々は、再構成された係数をリカバーする。各ステップで生成された再構成された係数は、ステップ１４３０、１４０、および１４９０で量子化および符号化された特徴またはベクトルの再構成を提供する。各再構成は、デコーダ側で見られる特徴またはベクトルのバージョンがビットストリーム１２１を符号化する際に使用されるものを反映するように、モデル化されるサブピクチャエンコーダによって実行される符号化動作による損失をモデル化して考慮することを可能にする。実装形態５００において符号化ビットストリームを生成するために使用される量子化機能（たとえば、５１２、５２８、および５５２において）の動作によって被る損失も考慮される。

【０１７２】

ステップ１４１１０において、ドット積モジュール５６４は、プロセッサ２０５の実行下で、再構成された基底ベクトル５６２内の各ベクトルに対してテンソル５４６内の各チャネルのドット積を実行して、係数のセット、すなわち係数５６６を生成する。係数５６６は、各特徴マップのコンテンツを再生する際の各基底ベクトルの寄与を表す。プロセッサ２０５における制御は、ステップ１４１１０から係数符号化ステップ１４１２０に進む。

【０１７３】
ステップ１４１２０において、量子化器モジュール５６８は、プロセッサ２０５の実行下で、係数５６６を量子化して、整数係数５７０を生成する。整数係数５７０は、サブピクチャ１２１４にパッキングするためにサブピクチャエンコーダ５７２に渡される。サブピクチャ１２１４は、ビットストリーム部分５７３を生成するためにモジュール５７２によって符号化される。プロセッサ２０５における制御は、ステップ１４１２０からサブピクチャ合成ステップ１４１３０に進む。テンソル１１５が２つ以上のテンソルを含む場合、ステップ１４２０～１４１２０はテンソル１１５内の各テンソルについて繰り返され、最終圧縮表現はステップ１４１３０において単一のピクチャに構成される。

【０１７４】
上述したように、モジュール５５２～５７２は、モジュール５１２～５３２と同様に動作して、ビットストリームを生成する。ステップ１４１３０において、サブピクチャビットストリームコンバイナ５８０は、プロセッサ２０５の実行下で、サブピクチャ１２１０、１２１２、１２１６、および１２１４にそれぞれ対応するビットストリーム部分５１５、５３３、５５７、および５７３を単一のビットストリーム、すなわち１２１に結合する。

【０１７５】
方法１４００は完了し、プロセッサ２０５における処理は、次のフレームのために方法１４００を呼び出すように進む。

【０１７６】
図１５は、分割ＣＮＮの第２部分を実行するためにＰＣＡデコーダ８００によって実装されるような復号テンソル１４９を生成するためにビットストリーム１４３を復号するための方法１５００を示す。方法１５００は、構成されたＦＰＧＡ、ＡＳＩＣ、またはＡＳＳＰなどの装置を使用して実装され得る。代替的に、以下で説明するように、方法１５００は、プロセッサ２０５の実行下で、アプリケーションプログラム２３３の１つまたは複数のソフトウェアコードモジュールとして、ソースデバイス１１０によって実装され得る。方法１５００を実施するアプリケーションプログラム２３３のソフトウェアコードモジュールは例えば、ハードディスクドライブ２１０および／またはメモリ２０６に常駐することができる。方法１５００は、コンピュータ可読記憶媒体および／またはメモリ２０６に記憶され得る。方法１５００は、平均特徴復号ステップ１５１０で開始する。

【０１７７】
ステップ１５１０において、ピクチャデコーダ８０４は、プロセッサ２０５の実行下で、ビットストリーム１４３から（サブピクチャ８４０に対応する）サブピクチャ１２１０を復号する。プロセッサ２０５における制御は、ステップ１５１０から平均係数復号ステップ１５２０に進む。

【０１７８】
ステップ１５２０において、ピクチャデコーダ８０４は、プロセッサ２０５の実行下で、ビットストリーム１４３から（サブピクチャ８３０に対応する）サブピクチャ１２１２を復号する。プロセッサ２０５における制御は、ステップ１５２０からＤＣ成分決定ステップ１５３０に進む。

【０１７９】
ステップ１５３０において、アンパッカ８４２および８３２、逆量子化器８４６および８３６、ならびにドット積モジュール８５０は、プロセッサ２０５の実行下で、テンソル８５２を生成する。アンパッカ８４２および８３２は、サブピクチャをアンパックし、各サブピクチャは平均特徴５１１に関する量子化された特徴マップを含み、実装５００の量子化機能を使用して量子化される。逆量子化器８４６は、アンパッカ８４２によってサブピクチャ８４０からアンパックされた平均特徴マップを逆量子化して、平均特徴マップ８４８を導出する。逆量子化器８３６はアンパッカ８３２によってサブピクチャ８３０からアンパックされた平均係数を、ビットストリーム１４３においてシグナリングされるそれぞれの量子化範囲を使用して整数領域から浮動小数点領域に逆量子化し、平均係数８３８

を生成する。ドット積モジュール８５０は、平均特徴８４８および平均係数８３８に対してドット積を実行してテンソル８５２を生成する。テンソル８５２は、ゼロ中心テンソルからの非ゼロ中心テンソルの再構成のためのオフセットを表す。プロセッサ２０５における制御は、ステップ１５３０から基底ベクトル復号ステップ１５４０に進む。

【０１８０】

ステップ１５４０において、ピクチャデコーダ８０４は、プロセッサ２０５の実行下で、ビットストリーム１４３から（サブピクチャ８２０に対応する）サブピクチャ１２１６を復号する。サブピクチャ１２１６は、復号サブピクチャ８２０としてアンパッカ８２２に渡される。アンパッカ８２２は、ステップ１５６０において、逆量子化のために、復号された基底ベクトル、例えば１２２６を抽出する。図２２を参照して、ビットストリーム１４３から基底ベクトルを復号する際の宛先デバイス１４０の動作について説明する。プロセッサ２０５における制御は、ステップ１５４０から係数復号ステップ１５５０に進む。

【０１８１】

ステップ１５５０において、ピクチャデコーダ８０４は、プロセッサ２０５の実行下で、ビットストリーム１４３から（サブピクチャ８１０に対応する）サブピクチャ１２１４を復号する。サブピクチャ１２１４は、復号サブピクチャ８１０としてアンパッカ８１２に渡される。アンパッカ８１２は、ステップ１５６０において、逆量子化のために、復号された係数、例えば、１２２４を抽出する。プロセッサ２０５における制御は、ステップ１５５０からＡＣ成分決定ステップ１５６０に進む。

【０１８２】

ステップ１５６０において、アンパッカ８２２および８１２、逆量子化器８２４および８１６、ならびにドット積モジュール８２８は、プロセッサ２０５の実行下で、ゼロ中心テンソル８５４を生成するように動作する。アンパッカ８２２は、サブピクチャ８２０から基底ベクトルを抽出し、基底ベクトルを整数特徴マップ８２２として逆量子化器８２４に渡す。逆量子化器は、ビットストリーム１４３から得られた量子化範囲に従って、基底ベクトルを整数領域から浮動小数点領域に変換し、基底ベクトル８２６を生成する。アンパッカ８１２は、サブピクチャ８１０から係数を抽出し、係数を整数係数として逆量子化器８１６に渡す。逆量子化器８１６は、浮動小数点領域係数８１８を生成する。ドット積モジュール８２８は、係数８１８の各係数とベクトル８２６内のそれぞれの基底ベクトルとのドット積を実行して、ゼロ中心再構成テンソル、すなわちテンソル８５４を生成する。プロセッサ２０５における制御は、ステップ１５６０からテンソル生成ステップ１５７０に進む。

【０１８３】

ステップ１５７０において、加算モジュール８５６は、プロセッサ２０５の実行下で、ゼロ中心再構成テンソル８５４をテンソル８５２と合計して、再構成テンソル１４９を生成する。ＰＣＡプロセスの最適な、またはほぼ最適な次元削減のために、テンソル１４９はテンソル１１５と実質的に同様であり、したがって、所与のマシンタスクを実行するためのＣＮＮヘッド１５０による使用に適している。ＦＰＮが使用されている場合、ステップ１５１０～１５７０は、方法１９００から得られたＦＰＮレイヤの表現の各テンソルに対して実行される。プロセッサ２０５における制御は、ステップ１５７０からテンソル抽出ステップ１５７５に進む。

【０１８４】

ステップ１５７５において、プロセッサ２０５は、テンソル抽出器１７２の動作によって、任意の複合テンソルから別個のＦＰＮレイヤテンソルを抽出するために実行する。テンソル抽出器１７２は、図２０に示される方法２０００を実行して、テンソル１４９ａの任意の複合テンソルから別個のＦＰＮレイヤテンソルを抽出し、その結果をテンソル１４９に沿って渡す。プロセッサ２０５における制御は、ステップ１５７５からニューラルネットワーク第２部分実行ステップ１５８０に進む。

【０１８５】

ステップ１５８０において、ＣＮＮヘッド１５０は、プロセッサ２０５の実行下で、テンソル１４９を入力として使用してニューラルネットワークの後段を実行する。例えば、図１０Ａ又は図１１を参照して説明したネットワークを用いた分類は、テンソル１４９の次元数（及び、その中に含まれるテンソルの数、ＦＰＮを用いる場合）に応じて行うことができる。さらに、いくつかのネットワークがバックボーン部分にマッチングアーキテクチャおよび重みを有する場合、そのようなバックボーン部分から符号化されたテンソルを使用して、ＣＮＮヘッド１５０の異なるインスタンスを実行することによって、これらの異なるネットワークの完了を実行することができる。ステップ１５８０は、タスク結果１５１を出力するために実行される。

【０１８６】

図１６Ａおよび図１６Ｂは、固有ベクトル「不安定性」の概念を示すために、２つの連続するデータセットおよび関連する固有ベクトルを示し、これは、類似のデータセットに対して分解が繰り返し実行されるときに生じる。非相関テンソルがビデオから生成されるとき、連続するフレームは強く相関され、したがって、結果として得られる基底ベクトルも極めて類似している。図１６Ａの第１データセット１６００は、２次元データセットを示し、図１６Ｂの第２データセット１６４０は、２次元データセットの後のバージョンを示す。データセット１６００および１６４０は、類似しているが同一ではなく、ビデオデバイス１１２からのフレームキャプチャなどのサンプリングプロセスに対してＣＮＮバックボーン１１４動作を実行することによって導出され得る。特徴マップ内の各サンプルは、システム１００における分解の目的のために別個の次元を形成するが、説明の目的のために、２次元データセットが図１６Ａおよび１６Ｂに示される。単一の固有ベクトルが生成され、２次元から１次元への削減をもたらす。データセット１６００については、固有ベクトル１６１０が生成され、データセット１６４０については、固有ベクトル１６５０が生成される。データセット１６００および１６４０が類似しているので、固有ベクトル１６１０および１６５０は類似の絶対勾配を有し、「勾配」は、主成分の切片が原点を通ることに留意して、２次元ベクトルの場合におけるベクトルの方向を反映する。しかしながら、データセット１６００および１６４０におけるわずかな差異のために、分解を実行するためのプロセスは、固有ベクトルの方向を「反転（flip）」または反転（invert）させ、逆方向を有するベクトル１６１０および１６５０をもたらすことがある。言い換えれば、１６５０の反転は、ベクトル１６１０のそれぞれの要素の負である各要素（すなわち、ｘおよびｙ成分）を表す。基底ベクトルの方向の反転は、以前に生成された基底ベクトルのそれぞれの要素と同じ（または同様の）大きさを有するが符号が反対であるベクトルの各要素に対応する。基底ベクトルの場合、図１６Ａおよび１６Ｂの例に示されるよりも多くの次元が存在する。例えば、１３６×７２の特徴マップサイズは、９７９２個の要素または９７９２個の次元を有する基底ベクトルを示す。したがって、ベクトル１６５０に負の１を乗算すると、ベクトル１６１０と実質的に同じ値を有するベクトルが得られる。

【０１８７】

テンソル１１５の特徴マップにわたる分解が繰り返し実行されるとき、結果として生じる基底ベクトルの要素ごとの反転の同じ問題が生じる。個々の基底ベクトルは、たとえ基底ベクトルの内容が実質的に類似していても、時々その方向を反転させることができる。この極性反転は任意の基底ベクトルに対して発生する可能性があり、所与の基底ベクトルに対する発生は、他の基底ベクトルに対する発生とは無関係である。パッキングされた基底ベクトルを含む連続するフレームを圧縮するとき、反転された基底ベクトルに対応する反転された領域がビデオエンコーダ６１４による効率的なツール選択のために容易に認識されないので、反転は、高い符号化効率を達成するためのインター予測の能力を低減する。

【０１８８】

図１７は、ＣＮＮの第１部分からの所与のレイヤから連続する特徴マップを符号化するための方法１７００を示し、連続する特徴マップは同じ次元数を有する。方法１７００は

、低減された次元空間における表現がＰＣＡの呼び出し間の指向性に関して安定化されるように動作する。方法１７００は、ビデオデータを処理するのに特に適しており、低減された次元表現は、複数回、おそらくフレームごとに１回の頻度で再生成される必要があり、連続するビデオフレームに相関があり、連続するビデオフレームからのテンソルに対してＰＣＡを実行することによって生成される基底ベクトルに相関が生じる。そのような場合、結果として生じる基底ベクトルの方向は、基底ベクトル内の全体的な基底およびサンプルごとの大きさが比較的不変であるにもかかわらず、基底ベクトルのすべてのサンプルにわたって１つの呼び出しから次の呼び出しに反転し得る。そのような「極性反転」は、基底ベクトル生成プロセスにおける不安定性、または入力テンソルにおける小さな変化に対する感度に起因し得る。方法１７００は、構成されたＦＰＧＡ、ＡＳＩＣ、またはＡＳＳＰなどの装置によって実装され得る。方法１７００は、プロセッサ２０５の実行下で、アプリケーションプログラム２３３の１つまたは複数のソフトウェアコードモジュールとして、たとえばＰＣＡエンコーダ１６０の機能として、宛先デバイス１４０によって実装され得る。方法１７００は、ビットストリーム１４３において符号化されたビデオデータの各フレームに対して繰り返される。方法１７００を実施するアプリケーションプログラム２３３のソフトウェアコードモジュールは例えば、ハードディスクドライブ２１０および／またはメモリ２０６に記憶され得る。方法１７００は、第１ＰＣＡ実行ステップ１７１０で開始する。

【０１８９】

ステップ１７１０において、ＰＣＡエンコーダ１６０は、プロセッサ２０５の実行下で、第１テンソルの基底ベクトルの第１セットを生成し、第１テンソルは、ビデオソース１１２からの所与のフレームを処理するＣＮＮバックボーン１１４から生成される。例えば、分解モジュール５４８は、テンソル５４６を受信することができる。プロセッサ２０５における制御は、ステップ１７１０から基底ベクトル格納ステップ１７２０に進むステップから進む。

【０１９０】

ステップ１７２０において、ＰＣＡエンコーダ１６０は、プロセッサ２０５の実行下で、分解モジュール５４８の後続の呼び出しにおいて使用するために、基底ベクトルの第１セットをメモリ２０６に格納する。プロセッサ２０５における制御は、ステップ１７２０から第２ＰＣＡ実行ステップ１７３０に進む。

【０１９１】

ステップ１７３０において、ＰＣＡエンコーダ１６０は、プロセッサ２０５の実行下で、第２テンソルの基底ベクトルの第２セットを生成し、第２テンソルは、ビデオソース１１２からのステップ１７１０の所与のフレームに後続するフレームを処理するＣＮＮバックボーン１１４から生成される。ステップ１７３０は、ステップ１７２０と同様に動作し、すなわち、分解モジュール５４８を使用する。後続のフレームは、ビデオソース１１２によって提供される次のフレームであってもよく、または後のフレームであってもよい。ステップ１７３０で生成された基底ベクトルは、現在のテンソルの成分を提供し、一方、ステップ１７２０で記憶された基底ベクトルは、先行するテンソルに関連する。基底ベクトルの生成がすべてのフレームに対してより少ない頻度で実行される場合、生成は周期的（Ｎフレームごと）であり得るか、またはフレームコンテンツが有意な新しい情報を含むとみなされるときなど、フレームのコンテンツに関するいくつかの基準によって駆動され得る。例えば、基底ベクトルのさらなる生成は、ビデオソース１１２によって提供される以前のフレームと比較して大きな動きまたは他の有意な変化の検出のために実行され得る。ビデオソース１１２からのビデオフレームにおける相関のために、基底ベクトルの第１および第２セットは、ある程度の相関を有し、図１２Ａおよび１２Ｂのパッキングされた表現においてインター予測ツールを使用する圧縮に適した基底ベクトルを作成する。しかしながら、第１フレームと第２フレームとの間の雑音または他の小さな変動は、基底ベクトルの第１セットのそれぞれの基底ベクトルと比較して、反対の「方向」に生成される（または反転される）基底ベクトルの第２セットの様々な基底ベクトルをもたらし得る。プ

ロセッサ２０５における制御は、ステップ１７３０から基底ベクトル反転チェックステップ１７４０に進む。

【０１９２】

ステップ１７３０で生成された基底ベクトルは、現在のテンソルの成分を提供し、一方、ステップ１７１０で生成された基底ベクトルは、ビデオデータ１１３内の以前のフレームを表す先行テンソルの成分を提供する。ステップ１７４０では、現在の基底ベクトルが極性反転をチェックするためにテストされる。極性反転をチェックすることは、ＦＰＮの所与の符号化されたレイヤまたはレイヤについて、現在（ステップ１７３０）および前（ｅａｒｌｉｅｒ）または前（ｐｒｅｃｅｄｉｎｇ）（ステップ１７１０）に生成された基底ベクトルをテストして、基底ベクトル内の各要素の符号をスワップまたは反転することを除いて、ベクトルが実質的に類似したままであるかどうかを決定することを伴う。所与の基底ベクトル（ステップ１７１０の第１または先行ベクトルで始まる）について、合計が決定される。合計は、基底ベクトルの第１および第２セットからのそれぞれのベクトルの要素ごとの加算を最初に実行した結果である。要素ごとの加算から得られたｓｉｍｓの絶対値の合計が、類似性スコアを生成するために基底ベクトルごとに決定され、類似性スコアは、基底ベクトルごとに単一のスカラー値である。

【０１９３】

所与の基底ベクトルに対する類似性スコアの低い値は、基底ベクトルの第１セットにおける対応する基底ベクトルと実質的に類似しているが極性が反対であるセットの第２基底ベクトルを示す。したがって、類似性スコアの低い値は、分解の連続するパフォーマンス間の基底ベクトルにおける極性反転の指標である。合計が閾値を下回る場合、極性反転が検出され（ステップ１７４０において「ＹＥＳ」）、プロセッサ２０５における制御は、ステップ１７４０から基底ベクトル反転ステップ１７５０に進む。合計が閾値を下回らない場合、極性反転は検出されず（ステップ１７４０で「ＮＯ」）、プロセッサ２０５における制御は、ステップ１７４０から最後の基底ベクトルテストステップ１７６０に進む。類似性閾値は、典型的には予め決定される。いくつかの実施形態では、類似性閾値が現在のテンソルの幅および高さのうちの少なくとも１つ、例えば、幅および／または高さの０．８または０．９に線形的に関連する。他の構成では、類似性閾値がテンソルの１つのチャネルのエリアのｌｏｇ２に等しくてもよい。類似性閾値は、非反転ケースを表すフィルタリングされた類似性スコアなどの要因に基づいて決定され得る。中央値フィルタは、外れ値（例えば、極性反転を示す低スコア）を除外するために、類似性スコアに適用され得る。次いで、メジアンフィルタリングされた類似性スコアは、スケーリングされ、極性反転を検出するために所与のフレームの類似性スコアと比較され得る。フィルタリングは、基底ベクトルとレイヤとの間の異なる統計を考慮するために、基底ベクトルごとおよびレイヤごとの粒度に実行され得る。期待されるコンテンツの変化およびコンテンツの変動の結果としての類似性スコアの変動を許容することができる。閾値を設定する際の許容は、使用される分解方法などに基づいてもよい。

【０１９４】

基底ベクトル反転ステップ１７５０において、ＰＣＡエンコーダ１６０は、プロセッサ２０５の実行下で、現在処理されている基底ベクトルの極性を、例えば、ベクトルに－１（負の１）を乗算することによって反転し、更新された基底ベクトルを生成する。事実上、現在の基底ベクトルからの成分の更新されたセットが生成され、その結果、成分の更新されたセット内の成分は、ステップ１７３０で生成された基底ベクトル内のそれぞれの成分の逆である。この反転の結果として、現在の基底ベクトルは、基底ベクトルの第１セットからの対応する基底ベクトルに実質的に類似するようになる。いったん反転が行われると、次の分解動作でさらなる極性反転が行われない場合、現在の分解動作の反転された基底ベクトルとの整合極性を維持するために、基底ベクトルを後続の分解動作から反転し続ける必要がある。基底ベクトルごとに１つのフラグを有するフラグのセットは例えば、後続のフレームのための基底ベクトル反転ステップの持続性を提供するために、将来の基底ベクトル生成動作のための各基底ベクトルの反転状態を示すために使用される。現在の基

底ベクトルのためのフラグは、極性反転が起こったことを示すためにフラグのセットにおいてトグルまたは変更され、極性反転が最初に検出されたときに反転を開始し、第２極性反転が検出されたときに反転を停止する場合の処理を可能にする。フラグのセットは、任意の後続テンソルの分解機能を使用して生成された成分が反転されるべきかどうかの基底を提供する。プロセッサ２０５における制御は、ステップ１７５０から最後の基底ベクトルテストステップ１７６０に進む。

【０１９５】

ステップ１７６０において、プロセッサ２０５は、処理されている現在のベクトルが基底ベクトルのセット内の最後のベクトルであるかどうかをテストする。そうでない場合（ステップ１７６０で「ＮＯ」）、処理は基底ベクトルのセット内の次の基底ベクトルに進み、プロセッサ２０５内の制御は、ステップ１７６０からステップ１７４０に進む。その場合（ステップ１７６０で「ＹＥＳ」）、処理はステップ１７６０から基底ベクトル符号化ステップ１７７０に進む。

【０１９６】

ステップ１７７０において、エンコーダ１６０は、プロセッサ２０５の実行下で、ステップ１７５０において反転が実行されたベクトルのための更新された基底ベクトルと、反転が実行されなかったベクトルのための基底ベクトルの第２セットからの基底ベクトルとを含む最終基底ベクトルに対して動作する。基底ベクトルのこの最終セットは、方法１７００の後続の呼び出しのための基準点として、すなわち「基底ベクトルの第１セット」として記憶される。基底ベクトルの最終セットは、基底ベクトル５５０として量子化器モジュール５５２に渡され、サブピクチャエンコーダ５５６によって符号化される。さらに、反転が１回実行される場合、各コンポーネントのフラグは方法１７００の後続の呼び出しにおいて、極性が方法１７００の前回の呼び出しの極性と一致することを保証するために、それぞれのコンポーネントに対して反転が実行される必要があることを示す。反転が方法１７００の後続の呼び出しで２回目に検出される場合、フラグはトグルされ、方法１７００のさらなる呼び出しの際に、それぞれの成分の反転の停止をもたらす。次いで、方法１７００は現在のフレームについて終了し、次のフレーム、または後のフレームについて再び呼び出され得、ここで、ＰＣＡエンコーダ１６０は、すべてのフレームよりも少ない頻度で実行される。

【０１９７】

図１８Ａは、アップサンプリング動作またはダウンサンプリング動作のいずれかを使用して異なるＦＰＮレイヤからのテンソルを結合するためのモジュール１６２の例示的な実装形態１８００を示す概略ブロック図である。図１８Ａの例は、ＣＮＮバックボーン部分４００からのＰ２－Ｐ５レイヤを使用する。レイヤＰ６　４２９は、ＰＣＡデコーダ１７０が復号されたＰ５レイヤからＰ６レイヤを再構成することができるので、考慮から省略される。図１８Ａに示される例は、Ｐ２またはＰ３のうちの１つのリサンプリングに関するものであり、以下の図１９を参照して説明される。

【０１９８】

図１８Ｂは、アップサンプリング動作またはダウンサンプリング動作のいずれかを使用して、合成表現からテンソルを抽出して、ＦＰＮレイヤに対応する別個のテンソルに戻すためのモジュール１７２の実装１８９０を示す概略ブロック図である。図１８ＢのＴｅｊ例は、ＣＮＮヘッド１１００のための例示的なＰ２－Ｐ５レイヤを提供する。レイヤＰ６　１１１８は、レイヤＰ５　１１１６に対して最大プーリング演算を実行することによって導出され、したがって、圧縮または解凍される必要はない。図１８Ｂに示される例は、Ｐ２とＰ３とが組み合わされた図１８Ａに示される場合に関するものであり、以下の図２０を参照して説明される。

【０１９９】

図１９は、ＣＮＮのマルチレイヤ第１部分からの別のテンソルとの非相関のためにＣＮＮのマルチレイヤ第１部分からテンソルを再サンプリングするための方法１９００を示す。方法１９００は、構成されたＦＰＧＡ、ＡＳＩＣ、またはＡＳＳＰなどの装置によって

実施され得る。代替的に、以下で説明するように、方法１９００は、プロセッサ２０５の実行下で、アプリケーションプログラム２３３の１つまたは複数のソフトウェアコードモジュールとして、ソースデバイス１１０によって実施され得る。方法１９００は、ビットストリーム１４３において符号化されたビデオデータの各フレームについて繰り返される。方法１９００を実施するアプリケーションプログラム２３３のソフトウェアコードモジュールは例えば、ハードディスクドライブ２１０および／またはメモリ２０６に記憶され得る。方法１９００は、テンソルマッピング判断ステップ１９１０で開始する。

【０２００】

ステップ１９１０において、テンソルコンバイナ１６２は、プロセッサ２０５の実行下で、ＦＰＮの２つの異なるレイヤからの（テンソル１１５ａの）テンソルが結合されるかどうかを判断する。組合せは、典型的には、テンソル１１５ａが、第１サイズを有する特徴マップの第１セットと、第１サイズとは異なる第２サイズを有する特徴マップの第２セットと、を少なくとも有する場合に生じる。結合されない場合、テンソル１１５ａは、変更されずに通過される。テンソルが結合される場合、テンソルコンバイナ１６２は、より大きいテンソルのダウンサンプリング、またはより小さいテンソルのアップサンプリングを実行することによってテンソルが結合されるか、または再サンプリングなしにテンソルが結合されるか、を判断する。アップサンプリングおよびダウンサンプリングの選択肢はテンソル１１５ａが少なくとも２つのテンソルを含むときに利用可能であり、２つのテンソルの幅および高さは２倍の関係にあり、すなわち、含まれる２つの動作に等しいストライドを有する層状ネットワークから生成される。再サンプリングなしで連結するオプションはテンソル１１５ａが少なくとも２つのテンソルを含み、２つのテンソルが同じ幅および高さを有する場合に利用可能である。テンソル１１５ａの次元（dimensionality）はネットワークアーキテクチャおよび分割点によって設定されるので、テンソルコンバイナ１６２に利用可能なオプションはネットワークアーキテクチャおよび分割点の選択によって制約される。テンソル１１５ａが１つのテンソルのみを含む場合、テンソルを結合するための範囲はなく、テンソルコンバイナ１６２の唯一の選択肢は、テンソル１１５ａをテンソル１１５として変更せずに通過させることである。同じ幅であるが高さの半分または２倍のテンソルへのダウンサンプリングまたはアップサンプリングが評価される場合、ｒｅｓａｍｐｌｅ＿ｖｅｒｔｉｃａｌ＿ｏｎｌｙ＿ｆｌａｇシンタックス要素は、垂直方向のみにおけるリサンプリングの適用を示す。同じ高さであるが幅の半分または２倍のテンソルへのダウンサンプリングまたはアップサンプリングが評価される場合、ｒｅｓａｍｐｌｅ＿ｈｏｒｉｚｏｎｔａｌ＿ｏｎｌｙ＿ｆｌａｇシンタックス要素は、水平方向のみにおけるリサンプリングの適用を示す。垂直にのみ、または水平にのみ再サンプリングするためのオプションはいくつかのＣＮＮアーキテクチャで遭遇し得るように、２に等しいストライドが垂直または水平にのみ適用される、異なるレイヤにおけるテンソル結合を可能にする。

【０２０１】

より大きなテンソルがダウンサンプリングされる場合、より大きなテンソルにおける詳細は失われる。したがって、ダウンサンプリングは、主に大きなオブジェクトを含むフレームデータ１１３などの、詳細を欠くデータにより適している。フレームデータ１１３がより小さい詳細を含むと判定された場合、プロセッサ２０５は、より大きいテンソルをダウンサンプリングすることを選択しない。ダウンサンプリングかアップサンプリングかにかかわらず、クロスチャネル分解に基づいて圧縮を適用する前に２つの異なるレイヤからテンソルを組み合わせた結果は、レイヤにわたって冗長性を発見する能力である。各レイヤがＰＣＡエンコーダ１６０によって独立して符号化される場合、各テンソルのチャネルにわたる冗長性は発見されるが、レイヤにわたる冗長性は発見することができない。レイヤにわたる冗長性が識別される場合、２つの異なるＦＰＮレイヤを結合するテンソルを符号化するために必要とされる基底ベクトルの総数は各ＦＰＮレイヤのテンソルに割り振られる基底ベクトルの数よりも低くなり得、ここで、テンソルは単一のレイヤに結合されない。より小さいテンソルがアップサンプリングされる場合、レイヤ間冗長性を利用すると

いう点での利点が見られ、その結果、結合テンソルの基底ベクトルの数を増加させる必要がほとんどまたは全くない。２つのレイヤ間で幅および高さが同じである（チャネルカウントが異なり得る）階層化ネットワークの場合、テンソルコンバイナ１６２は、同じ幅および高さを有する２つのレイヤからテンソルを再サンプリングすることなく、クロスレイヤ冗長性を発見することを試みることができる。プロセッサ２０５は再サンプリング、アップサンプリング、またはダウンサンプリングをせずに結合し、テンソルを修正せずに残すオプションを試み、結果として得られるテンソル１１５の符号化コストが最も低くなるオプションを選択し得る。他の構成では、ステップ１９１０が再サンプリングが実施されるかどうかの所定のデフォルト動作を設定することができる。ステップ１９１０で決定されたマッピングは所定のマッピングであってもよいし、または、フレーム毎など、より定期的に決定されてもよい。マッピングが予め決められている場合、マッピングは、ビットストリームの開始時にのみシグナリングされる必要がある。ＳＥＩメッセージ１３１３中のｌａｙｅｒ＿ｕｐｄａｔｅ＿ｍａｐｐｉｎｇ＿ｉｎｆｏシンタックス要素は、マッピングが宛先デバイス１４０に通信されるべきＳＥＩメッセージ１３１３のシグナリングインスタンスを許可する。所定のマッピングの場合、ＣＬＶＳの第１ピクチャ１２００のマッピングの１回のシグナリングインスタンスで十分である。プロセッサ２０５における制御は、ステップ１９２０からテンソルマッピング符号化ステップ１９２０に進む。

【０２０２】

ステップ１９２０において、テンソルコンバイナ１６２は、プロセッサ２０５の実行下で、ステップ１９２０の動作から得られた判定をビットストリーム１２１に符号化する。符号化された判定は、テンソルコンバイナ１６２において実行される動作を逆転させるためにテンソルセパレータ１７２において実行される必要がある動作を表す。ＦＰＮレイヤをアップサンプリングする、ダウンサンプリングする、または変更されないままにする決定は、フラグまたは可変長コードワードのいずれかを使用してＳＥＩメッセージ１３１３中で符号化される。ステップ１９１０で行われた決定をビットストリーム１２１に符号化するためのシンタックスは付録Ａを参照して説明される。シンタックスがテンソル１４９がテンソル１１５ａと同じ数、次元、およびチャネルカウントを有するように、すなわち、テンソルコンバイナ１６２の動作の前に、テンソルセパレータ１７２において実行されるべき動作を符号化する。テンソルコンバイナ１６２によってテンソルがアップサンプリングされ、別のテンソルと結合された場合、シグナリングは、テンソルセパレータ１７２が復号された結合テンソルから２つのテンソルを抽出することを示す。抽出されたテンソルのうちの１つがテンソル１４９のうちの１つとして出力され、抽出されたテンソルがダウンサンプリングされて、テンソル１４９に含まれる追加のテンソルが生成される。テンソルコンバイナ１６２によってテンソルがダウンサンプリングされ、別のテンソルと結合された場合、シグナリングは、テンソルセパレータ１７２が復号された結合テンソルから２つのテンソルを抽出することを示す。抽出されたテンソルのうちの一方はテンソル１４９のうちの一方として出力され、他方はアップサンプリングされて、テンソル１４９に含まれる追加のテンソルを生成する。テンソルコンバイナ１６２によってテンソルが再サンプリングされなかったが、別のテンソルと結合された場合、シグナリングはテンソルセパレータ１７２が復号された結合テンソルから２つのテンソルを抽出することを示し、その両方がテンソル１４９のうちのテンソルとして出力される。結合テンソルを取得するためのインデックス、すなわちｓｒｃ＿ｌａｙｅｒ＿ｉｄｘ、および再サンプリングされたテンソルを挿入するためのインデックス、すなわちｄｓｔ＿ｌａｙｅｒ＿ｉｄｘ。ソースおよび宛先レイヤインデックスのシグナリングはたとえば、連結されたレイヤがテンソル１１５ａ間で隣接していない、または別のレイヤの半分の空間解像度を持つレイヤがテンソル１１５ａ間で先に出現する、またはテンソル１１５ａ間で後に出現するケースをサポートするために、柔軟である。アップサンプリングが実装される符号化は、符号化時にアップサンプリングされた特徴マップが復号時にダウンサンプリングされるべきであり、逆もまた同様であることを指定する。プロセッサ２０５における制御は、ステップ１９２０からテンソルエンコーダマッピングテストステップ１９３０に進む。

【０２０３】
ステップ１９３０において、ステップ１９１０で行われた判定が、テンソル１１５ａの異なるＦＰＮレイヤからのテンソルを結合しないことである場合、方法１９００は終了する。プロセッサ２０５における制御は方法１４００の平均特徴決定ステップ１４２０に戻り、ここで、すべてのテンソル１１５ａが、残りのステップのためのテンソル１１５として渡される。ステップ１９１０で行われる判定がテンソル１１５ａのＦＰＮレイヤからのテンソルのテンソルコンバイナ１６２におけるアップサンプリングを実行することである場合、プロセッサ２０５における制御は、ステップ１９３０から選択テンソルアップサンプルステップ１９４０に進む。ステップ１９１０で行われる判定がテンソル１１５ａのＦＰＮレイヤからのテンソルのテンソルコンバイナ１６２におけるダウンサンプリングを実行することである場合、プロセッサ２０５における制御は、ステップ１９３０から選択テンソルダウンサンプルステップ１９５０に進む。ステップ１９１０で行われた判定がリサンプリングフィルタを適用せずにテンソル１１５ａのうちのテンソルのテンソルコンバイナ１６２において連結を実行することである場合、プロセッサ２０５における制御は、ステップ１９５８から進む。

【０２０４】
図１８Ａを参照すると、実装１８００は、入力Ｐ３～Ｐ５（それぞれ１８１０～１８１３）を受信する。実装１８００は、テンソルアップサンプラ１８２２と、テンソルダウンサンプラ１８３１と、連結モジュール１８１８および１８３６と、マルチプレクサ１８４４とを含む。

【０２０５】
アップサンプリングは、特徴マップ（またはテンソル）の第２セットの第２サイズが第１のサイズより大きい、またはその逆である場合に、第１テンソルに対して行われる。ステップ１９４０において、テンソルアップサンプラ１８２２はプロセッサ２０５の実行下で、テンソル、例えば、Ｐ３テンソル１８１１をテンソル１１５ａからアップサンプリングし、アップサンプリングされたＰ３テンソル１８２４を生成する。アップサンプリングされたテンソル１８２４は、アップサンプリングされた特徴マップのセット１８２４と呼ばれることもある。Ｐ３テンソル１８１１は、幅ｗおよび高さｈを有する。アップサンプリングされたＰ３テンソル１８２４は、幅２ｄおよび高さ２ｈを有する。アップサンプリングは、アップサンプリングされたＰ３テンソル１８２４に必要な中間サンプルを生成するために、補間フィルタ、最近傍フィルタ、または他のフィルタを使用して実行され得る。プロセッサ２０５における制御は、ステップ１９４０からテンソル連結ステップ１９４５に進む。

【０２０６】
ステップ１９４５において、連結モジュール１８１８は、プロセッサ２０５の実行下で、アップサンプリングされたＰ３テンソル１８２４とＰ２テンソル１８１０とをチャネル次元に沿って連結して、連結テンソル１８４２を生成する。連結テンソル１８４２は、幅２ｗおよび高さ２ｈと、テンソル１８２４および１８１０のチャネルカウントの合計に等しいチャネルカウントとを有する。マルチプレクサ１８４４は、テンソル１８４２を連結出力テンソル１８４６として選択する。テンソル１８４６、１８１２および１８１３は、残りの符号化ステップのためのテンソル１１５として出力される。次いで、方法１９００は終了し、プロセッサ２０５における制御は、方法１４００のステップ１４２０において再開する。

【０２０７】
ステップ１９５０において、テンソルダウンサンプラ１８３１は、プロセッサ２０５の実行下で、テンソル、例えば、Ｐ２テンソル１８１０をテンソル１１５ａからダウンサンプリングして、ダウンサンプリングされたＰ２テンソル１８３２を生成する。Ｐ２テンソル１８１０は、幅２ｗおよび高さ２ｈを有する。ダウンサンプリングされたＰ２テンソル１８３２は、Ｐ２テンソル１８１０の幅の半分および高さの半分を有するように空間的にダウンサンプリングされ、寸法ｗ、ｈをもたらす。ダウンサンプリングは、サブサンプリ

ングされたバイリニア、バイキュービック、または他のフィルタリング動作を使用して実行され得る。ダウンサンプリングまたは他のフィルタリングは、Ｐ２テンソル１８１０のチャネル次元に適用されないので、ダウンサンプリングされたＰ２テンソル１８３２のチャネルカウントは変更されない。プロセッサ２０５における制御は、ステップ１９５０からテンソル連結ステップ１９５５に進む。

【０２０８】

ステップ１９５５において、連結モジュール１８３６は、プロセッサ２０５の実行下で、チャネル次元に沿ってＰ３テンソル１８１１とダウンサンプリングされたＰ２テンソル１８３２とを連結して、連結テンソル１８３９を生成する。連結テンソル１８３９は、幅ｗおよび高さｈと、テンソル１８１１および１８３２のチャネルカウントの合計に等しいチャネルカウントとを有する。マルチプレクサ１８４４は、テンソル１８３９を、連結出力テンソル１８４６として選択する。テンソル１８４６、１８１２および１８１３は、残りの符号化ステップのためのテンソル１１５として出力される。再サンプリングが実施された場合、出力テンソル１８４６はＰ２およびＰ３および連結器１８１８および１８３６の動作に関して図１８Ａに記載されているように、特徴マップの再サンプリングされたセットと、再サンプリングされていない特徴マップのセットと、を含む。次いで、方法１９００は終了し、プロセッサ２０５における制御は、方法１４００のステップ１４２０において再開する。

【０２０９】

ステップ１９５８において、連結モジュール１８２３は、プロセッサ２０５の実行下で、テンソル１８１１とテンソル１８３２（この場合、両方とも同じ幅および高さを有する）とを連結して、連結テンソル１８３８を生成する。マルチプレクサ１８４４は、連結出力テンソル１８４６としてテンソル１８３８を通す。テンソル１８４６、１８１２、および１８１３は、残りの符号化ステップのためのテンソル１１５として渡される。次いで、方法１９００は終了し、プロセッサ２０５における制御は、方法１４００のステップ１４２０において再開する。

【０２１０】

方法１４００は、ステップ１４２０～１４１２０を実行し続け、例えば、ステップ１４８０においてステップ２０７０で出力されたテンソルを使用して基底ベクトルを決定し、ステップ１４１１０において基底ベクトルを使用して連結テンソル内の両方の特徴マップに関連する係数を導出し、ステップ１４９０および１４１２０において係数および基底ベクトルを符号化する。

【０２１１】

方法１９００の動作の結果として、連結テンソルは、ＦＰＮの２つの別個のレイヤを合成するテンソル１４９のうちの１つとして渡される。連結テンソルは、ＰＣＡなどの分解法を用いて圧縮される。分解方法はまた、アップサンプリング／ダウンサンプリングが発生したＦＰＮの２つのレイヤに沿って非相関化することができ、非相関化がＦＰＮの各レイヤに限定されるのではない。さらに、詳細が保存される必要がある場合、ソースデバイス１１０は、基底ベクトル導出および圧縮の目的のために、より小さいＦＰＮレイヤ、たとえば、Ｐ３からテンソルをアップサンプリングするオプションを有する。アップサンプリングされたＰ３レイヤとＰ２レイヤとの連結のために生成された基底ベクトルの数は、Ｐ２レイヤのために純粋に生成された基底ベクトルの数と比較して増加される必要はない。したがって、Ｐ３レイヤのためのピクチャ１２００中の基底ベクトルの追加のエリアは必要とされない。

【０２１２】

図２０は、ＣＮＮヘッド１５０に提供するための出力テンソル１４９を生成するために、復号された結合表現、すなわちテンソル１４９ａからテンソルを再サンプリングするための方法２０００を示す。方法２０００は、符号化データからテンソルを復号する際に使用される。方法２０００は、構成されたＦＰＧＡ、ＡＳＩＣ、またはＡＳＳＰなどの装置によって実施され得る。代替的に、以下で説明するように、方法２０００は、プロセッサ

２０５の実行下で、アプリケーションプログラム２３３の１つまたは複数のソフトウェアコードモジュールとして、宛先デバイス１３０によって実施され得る。方法２０００は、ビットストリーム１４３において符号化されたビデオデータの各フレームについて繰り返される。方法２０００を実施するアプリケーションプログラム２３３のソフトウェアコードモジュールは例えば、ハードディスクドライブ２１０および／またはメモリ２０６に記憶され得る。方法２０００は、テンソルマッピング復号ステップ２０１０で開始する。

【０２１３】

ステップ２０１０において、エントロピーデコーダ９２０は、プロセッサ２０５の実行下で、テンソル１４９ａが結合されたリサンプリングされたテンソルを含むかどうかを示すビットストリーム１４３に含まれるＳＥＩメッセージ１３１３からシンタックス要素を復号する。デコーダはリサンプリングを実行するか否かを、例えば、符号化時にリサンプリングが発生したか否かを特定する復号情報に基づいて、又は、デフォルト設定に基づいて、決定する。すなわち、エントロピー復号器は図１９の符号化で実装されるように、テンソルのうちの１つが、別のＦＰＮレイヤからのテンソルの再サンプリング（アップサンプリングまたはダウンサンプリング）されたバージョンを有する１つのＦＰＮレイヤからのテンソルの連結であるかどうかを復号する。方法２０００は、ステップ１５４０および１５５０における基底ベクトルおよび係数の復号がすでに行われ、ステップ１５７０においてテンソルが導出されたときに実行される。結合テンソルは、同じ最終幅および高さを有するテンソルの連結であってもよく、またはテンソル１４９ａに存在する連結テンソルがなくてもよい。

【０２１４】

ＳＥＩメッセージ１３１３におけるシグナリングは、テンソル１４９を復元するために、すなわち、ＣＮＮヘッド１５０による使用に適したテンソル１１５ａと同じ数および次元を有するテンソルのセットを生成するために、テンソルセパレータ１７２によって実行される必要がある動作を示す。テンソルコンバイナ１６２が、連結されたテンソルを生成するためにテンソルをダウンサンプリングしている場合、ビットストリーム１４３におけるシグナリングはテンソルを復元するために、連結されたテンソルの一部をアップサンプリングするようにテンソルセパレータ１７２に指示する。テンソルコンバイナ１６２が、連結されたテンソルを生成するためにテンソルをアップサンプリングしている場合、ビットストリーム１４３におけるシグナリングはテンソルを復元するために、連結されたテンソルの一部をダウンサンプリングするようにテンソルセパレータ１７２に指示する。テンソルコンバイナ１６２が連結されたテンソルを生成するためにテンソルを再サンプリングしていない場合、ビットストリーム１４３におけるシグナリングは、テンソルセパレータ１７２に、再サンプリングフィルタを適用せずに、連結されたテンソルの一部を抽出してテンソルを回復するように指示する。連結されたテンソルからのテンソルの抽出は、復号されたｌａｙｅｒ＿ｅｘｔｒａｃｔｉｏｎ＿ｆｌａｇシンタックス要素から決定される。抽出されたテンソルへのダウンサンプリングフィルタの適用は、ｌａｙｅｒ＿ｄｏｗｎｓａｍｐｌｅ＿ｆｌａｇシンタックス要素を介してシグナリングされる。抽出されたテンソルへのアップサンプリングフィルタの適用は、ｌａｙｅｒ＿ｕｐｓａｍｐｌｅ＿ｆｌａｇを介してシグナリングされる。テンソル１４９ａのうちのどのテンソルが連結テンソルであるかは、ｓｒｃ＿ｌａｙｅｒ＿ｉｄｘシンタックス要素を介してシグナリングされる。テンソル１４９を生成するために抽出されたテンソルが挿入されるオフセットは、ｄｓｔ＿ｌａｙｅｒ＿ｉｄｘシンタックス要素を介してシグナリングされる。２つのレイヤ間の連結テンソル内の境界を描写するチャネルインデックスは、ｓｒｃ＿ｃｈａｎｎｅｌ＿ｏｆｆｓｅｔシンタックス要素を介してシグナリングされる。プロセッサ２０５における制御は、ステップ２０１０からテンソルデコーダマッピングテストステップ２０２０に進む。

【０２１５】

ステップ２０２０において、ステップ２０１０の結果が再マッピングが適用されないことを示す場合、プロセッサ２０５における制御は、ステップ２０１０からテンソル出力ス

テップ２０７０に進む。ステップ２０１０の結果がテンソルダウンサンプリングが適用されるべきであることを示す場合、プロセッサ２０５における制御は、ステップ２０２０からテンソル抽出ステップ２０３０に進む。ステップ２０１０の結果がテンソルアップサンプリングが適用されるべきであることを示す場合、プロセッサ２０５における制御は、ステップ２０２０からテンソル抽出ステップ２０５０に進む。ステップ２０１０の結果が再サンプリングなしのテンソル抽出が適用されることを示す場合、プロセッサ２０５における制御は、ステップ２０２０からテンソル抽出ステップ２０３５に進む。

【０２１６】

図１８Ｂの実装１８９０は、入力Ｐ２／３、Ｐ４およびＰ５（それぞれ１８５０、１８５２および１８５４）を受信する。入力Ｐ２／３は例えば、方法１９００の動作によって、符号化側でのアップサンプリングまたはダウンサンプリングによって生成される結合テンソルである。実装１８９０は、テンソルスライサ１８５６、ダウンサンプラ１８６２、アップサンプラ１８６４及びセレクタ１８７０および１８７２を含む。

【０２１７】

ステップ２０３０において、テンソルスライサ１８５６は、プロセッサ２０５の実行下で、（ｓｒｃ＿ｃｈａｎｎｅｌ＿ｏｆｆｓｅｔに従うインデックスにおいて）チャネル次元に沿って、結合されたＰ２／Ｐ３テンソル１８５０を２つのテンソルにスライスし、各々はＰ２レイヤおよびＰ３レイヤのチャネルカウントに対応するチャネルカウント、すなわち、２５６個のチャネルを有する。テンソルスライサ１８５６は、第１のスライスされたテンソル１８５８および第２のスライスされたテンソル１８６０を出力する。プロセッサ２０５における制御は、ステップ２０３０から抽出テンソルダウンサンプルステップ２０４０に進む。

【０２１８】

ステップ２０３５において、テンソルスライサ１８５６は、プロセッサ２０５の実行下で、チャネル次元に沿って（ｓｒｃ＿ｃｈａｎｎｅｌ＿ｏｆｆｓｅｔに従うインデックスで結合Ｐ２／Ｐ３テンソル１８５０を２つのテンソルにスライスする。テンソルスライサ１８５６は、第１のスライスされたテンソル１８５８をセレクタ１８７０に出力し、第２のスライスされたテンソル１８６０をセレクタ１８７２に出力する。プロセッサ２０５における制御は、ステップ２０３５からテンソル出力ステップ２０７０に進む。

【０２１９】

ステップ２０４０において、ダウンサンプラ１８６２は、プロセッサ２０５の実行下で、テンソル１８５８をダウンサンプリングして、再スケールされたテンソル１８６６を導出する。再スケーリングされたテンソルは、結合テンソルを復号するために使用される特徴マップのセットを提供し、ダウンサンプラ１８６２は、デシメーション、またはサブサンプリングされたバイリニアもしくはバイキュービックもしくは他のフィルタを使用して、再スケーリングされたテンソル１８６６を生成し得る。プロセッサ２０５における制御は、ステップ２０４０からテンソル出力ステップ２０７０に進む。

【０２２０】

ステップ２０５０において、テンソルスライサ１８５６は、プロセッサ２０５の実行下で、チャネル次元に沿って結合されたＰ２／Ｐ３テンソル１８５０を２つの半分にスライスし、各々はＰ２レイヤおよびＰ３レイヤのチャネルカウントに対応するチャネルカウント、すなわち、２５６個のチャネルを有する。テンソルスライサ１８５６は、第１のスライスされたテンソル１８５８及び第２のスライスされたテンソル１８６０を出力する。プロセッサ２０５における制御は、ステップ２０５０から抽出テンソルアップサンプルステップ２０６０に進む。

【０２２１】

ステップ２０６０において、アップサンプラ１８６４は、プロセッサ２０５の実行下で、テンソル１８６０をアップサンプリングして、再スケーリングされたテンソル１８６８を生成する。アップサンプリングは再スケーリングされたテンソル１８６８をポピュレートするために必要とされる補間されたサンプルを導出するために、バイリニア、バイキュ

ービック、最近傍、または他のフィルタ方法を使用し得る。プロセッサ２０５における制御は、ステップ２０６０からテンソル出力ステップ２０７０に進む。

【０２２２】

ステップ２０４０および２０６０は、ステップ２０３０または２０５０で導出されたテンソルの第１部分を再サンプリング（それぞれダウンサンプリングおよびアップサンプリング）して、テンソル１４９ａを復号することによって、特徴マップの第１セットを導出するように動作する。

【０２２３】

ステップ２０７０において、出力テンソル１４９が、ＣＮＮヘッド１５０による使用のために提供される。Ｐ５テンソル１８５４およびＰ４テンソル１８５２は、テンソル１４９ａからテンソル１４９に渡される。再マッピングまたは再サンプリング動作が行われなかった場合、Ｐ２レイヤおよびｐ３レイヤに対応するテンソルは、テンソル１４９ａからテンソル１４９に渡される。再マッピング動作が行われた場合、Ｐ３テンソル１８８２およびＰ２テンソル１８８０が、テンソル１４９の対応するＦＰＮレイヤとして出力される。ダウンサンプリングが行われた場合、セレクタ１８７０はダウンサンプリングされたテンソル１８６６をテンソル１８８２として出力し、セレクタ１８７２は、テンソル１８５８をテンソル１８８０として出力する。アップサンプリングが行われた場合、セレクタ１８７０はテンソル１８６０をテンソル１８８２として出力し、セレクタ１８７２は、テンソル１８６８をテンソル１８８０として出力する。リサンプリングが実施された場合、出力テンソル１８８０および１８８２のうちの１つは、Ｐ２およびＰ３およびセレクタ１８７０および１８７２の動作に関して図１８Ｂに記載されているように、特徴マップのリサンプリングされたセットと、リサンプリングされていない特徴マップのセットと、を含む。再サンプリングが行われなかった場合、セレクタ１８７０はテンソル１８６０をテンソル１８８２として出力し、セレクタ１８７２は、テンソル１８５８をテンソル１８８０として出力する。方法２０００は終了し、プロセッサ２０５における制御は方法１５００に戻る。

【０２２４】

方法１９００および２０００の構成では、ステップ１９１０はまた、Ｐ２～Ｐ５のうちのどのレイヤがアップサンプリングまたはダウンサンプリングされるべきかを決定し、隣接するレイヤの対、すなわち、Ｐ２およびＰ３、Ｐ３およびＰ４、またはＰ４およびＰ５を選択する。ステップ１９２０において、隣接レイヤの選択されたペアがビットストリーム１２１に符号化される。ステップ２０１０において、隣接レイヤの選択されたペアがビットストリーム１４３から復号され、示されたテンソルが連結テンソルから復元される。

【０２２５】

図２１は、ＣＮＮバックボーン１１４からテンソル１１５を符号化するために分解モジュール５４８によって基底ベクトルを決定するための方法２１００を示す。方法２１００は、構成されたＦＰＧＡ、ＡＳＩＣ、またはＡＳＳＰなどの装置によって実施され得る。方法２１００は、付録Ａに存在する基底ベクトルパッキングフォーマット情報ＳＥＩメッセージシンタックスを参照して説明される。あるいは以下で説明されるように、方法２１００は、プロセッサ２０５の実行下で、アプリケーションプログラム２３３の１つまたは複数のソフトウェアコードモジュールとして、ソースデバイス１１０によって実施され得る。方法２１００は、ビットストリーム１２１において符号化されたビデオデータの各フレームに対して繰り返される。方法２１００を実施するアプリケーションプログラム２３３のソフトウェアコードモジュールは例えば、ハードディスクドライブ２１０および／またはメモリ２０６に記憶され得る。方法２１００は、最大基底ベクトルカウント符号化ステップ２１１０で開始する。

【０２２６】

ステップ２１１０において、エントロピーエンコーダ７３８は、プロセッサ２０５の実行下で、値ｂｖ＿ｍａｘ＿ｃｎｔをＳＥＩメッセージ１３１３に符号化する。値ｂｖ＿ｍａｘ＿ｃｎｔは、分解モジュール５４８によってビットストリーム１２１に生成されるベ

き基底ベクトルの数に対応する基底ベクトルパッキング情報１３９１の一部としてＳＥＩメッセージ１３１３に符号化される。値ｂｖ＿ｍａｘ＿ｃｎｔはテンソル１１５のうち、テンソルの分解に用いることができる基底ベクトルの最大数を示す。テンソルコンバイナ１６２においてレイヤの減少が生じなかった場合、ｂｖ＿ｍａｘ＿ｃｎｔの１つのインスタンスが、テンソル１１５の各ＦＰＮレイヤに対して符号化される。テンソルコンバイナ１６２の結果としてレイヤの数が低減された場合、１つのｂｖ＿ｍａｘ＿ｃｎｔ値が低減されたフォーマットで得られたレイヤの各々について、すなわち、符号化されたレイヤごとに、符号化される。ステップ２１１０は、初期化ステップとして実行される必要があるだけであり、したがって、１つまたは複数のｂｖ＿ｍａｘ＿ｃｎｔ値は、ビットストリーム１２１内の初期化またはランダムアクセスポイントにおいて符号化される必要があるだけである。さらに、ステップ２１１０において、各レイヤの基底ベクトルの幅および高さが、ＳＥＩメッセージ１３１３においてｂｖ＿ｗｉｄｔｈおよびｂｖ＿ｈｅｉｇｈｔとして符号化される。フレームサイズは、ステップ２１１０で復号された基底ベクトルの最大数を使用して、符号化データについて効果的に決定され、フレームサイズは符号化データの少なくとも１つのフレームに適用可能である。プロセッサ２０５における制御は、ステップ２１１０から基底ベクトルパッキングフォーマット決定ステップ２１２０に進む。

【０２２７】

ステップ２１２０において、サブピクチャエンコーダ５５６のためのパッカ６１０は、サブピクチャ１２１６のためのパッキングフォーマットと、サブピクチャ１２１６のためのサイズとを決定する。パッキングフォーマットおよびサイズはたとえば、図１２Ｂを参照して示されるように、テンソル１１５内の各テンソルについて指定されたサイズおよびカウントｂｖ＿ｍａｘ＿ｃｎｔの基底ベクトルがサブピクチャ１２１６にパッキングされることが可能であるように決定される。パッキングされた基底ベクトルを含む後続のピクチャ（すなわち、１２００）は、所与のピクチャに対して使用される基底ベクトルがより少なくても、方法２１００の最初の呼び出しで決定されたものと同じサイズを有する。したがって、サブピクチャエンコーダ（すなわち、５１４、５３２、５５６、および５７２）は同じサイズのフレームを使用して圧縮テンソル１１５の連続セットを符号化し、ピクチャ解像度の変化をシグナリングするためにＳＰＳ１３１０の新しいインスタンスを送る必要はない。ピクチャ解像度を変更することは、フレームバッファ７７２のリセットを必要とし、それは使用されるべきインター予測方法のために必要に応じて、後続のピクチャがフレームバッファ７７２内の以前のピクチャを参照することを防ぐ。最大基底ベクトルカウントに基づいてピクチャ１２００のサイズおよびパッキングフォーマットを決定することは、フレームバッファ７７２が互換性のない解像度のピクチャを収容するためにリセットされる必要がないので、利用される基底ベクトルの数が変更するときにインター予測が実行されることを可能にする。したがって、パッキングされる基底ベクトルの数が変化する場合であっても、ピクチャ１２００の固定サイズを維持することは、結果として得られるパッキングされたピクチャにわたってインター予測を使用することができるので、圧縮効率を改善する。プロセッサ２０５における制御は、ステップ２１２０から基底ベクトルセット決定ステップ２１３０に進む。

【０２２８】

ステップ２１３０において、分解モジュール５４８は、プロセッサ２０５の実行下で、テンソル１１５の各テンソルのＭ個の基底ベクトルのセットを生成し、ここで、Ｍは、テンソル１１５のそれぞれのテンソルの値ｂｖ_ｍａｘ_ｃｎｔ（’Ｍ’）に対応する。値Ｍは、Ｐ３およびＰ４などの２５６個のチャネルを含むレイヤに対して固定値２５に設定され得る。Ｐ２およびＰ３などの２つのレイヤがコンバイナ１６２において連結された場合、値Ｍはテンソル１１５ａのレイヤごとに２５個の基底ベクトルを維持することに対応して、すなわち、テンソルコンバイナ１６２からの任意のレイヤ低減の前に、５０に設定され得る。テンソルのＭ個の基底ベクトルは主成分の順序付けられたセットであり、セットの各々は、Ｍ個の基底ベクトルの先行する主成分によってまだ捕捉されていないテンソルの最大分散量を捕捉する。分散は、テンソルのチャネル次元にわたって「捕捉」または説明

される。したがって、Ｍ個の基底ベクトルは、テンソルの各チャネルがテンソルの所与のチャネルにおけるサンプルの数（すなわち、幅ｘ高さ）よりも少ない値で表現され得る空間を定義する。各基底ベクトルによって説明される分散の量は固有値である。Ｍ個の基底ベクトルはＭ個の固有値の対応するセットを有し、固有値は、厳密に減少する数の配列を形成する。プロセッサ２０５における制御は、ステップ２１３０から固有値累積ステップ２１４０に進む。

【０２２９】

ステップ２１４０において、分解モジュール５４８は、プロセッサ２０５の実行下で、各値ｎについて累積された「説明された分散」値を生成し、ここで、ｎは１からＭまでの範囲である。累積された説明された分散値は、第１固有値から第ｎ固有値までの固有値の合計に等しい。累積された説明された分散値は、ステップ２１３０で生成された固有ベクトルまたは主成分の総セットのサブセットによってどの程度の分散が説明されるかを示す。プロセッサ２０５における制御は、ステップ２１４０から基底ベクトルサブセット決定ステップ２１５０に進む。

【０２３０】

ステップ２１５０において、分解モジュール５４８は、プロセッサ２０５の実行下で、「説明された分散閾値」に対応する基底ベクトルＭのサブセットを選択する。説明された分散閾値は、使用される基底ベクトルの数を決定し、精度、エリア、および符号化効率の間の所望のトレードオフに従ってテンソルを適切に分解するように選択され得る。説明された分散閾値は、基底ベクトル自体におけるデータに基づいて設定されるか、または所望のエリアもしくは符号化効率、または最小の所望の精度に基づいて設定されることができる。基底ベクトルのサブセットの選択は、Ｍ個の基底ベクトルの最初のＮ個の基底ベクトルを選択すること、すなわち、利用可能な（最大の）基底ベクトルの中から基底ベクトルを選択することをもたらす。説明された分散閾値は、フレームごとに、すなわち、符号化されるデータのフレームごとに決定することができる。サブセットの選択は、累積された説明された分散に対するそれらのより小さい寄与によって、テンソルの再構成にあまり寄与しない基底ベクトルを省略する。サブセットの選択は、サブピクチャ１２１６がより少ない占有エリアを有するので、圧縮効率を改善する。サブピクチャ１２１６の非占有エリアは、一定のサンプル値によって占有され得る。一定のサンプル値は、ＶＶＣの柔軟なブロック構造のために圧縮性が高い。プロセッサ２０５における制御は、ステップ２１５０から基底ベクトルサブセット符号化ステップ２１６０に進む。

【０２３１】

ステップ２１６０において、エントロピーエンコーダ７３８は、プロセッサ２０５の実行下で、テンソル１１５の各テンソルの値ｂｖ＿ｃｎｔをビットストリーム１２１に符号化する。値ｂｖ＿ｃｎｔは、ステップ２１５０の実行から得られる選択されたサブセットを示す。値ｂｖ＿ｃｎｔは、基底ベクトル０～Ｎ、すなわち、使用される基底ベクトルの数が、所与のテンソルに対して１からＭまで変化し得ることを示し得る。ステップ２１６０は、テンソルを符号化するために使用される基底ベクトルの数を符号化する。使用されるべき基底ベクトルの数を指定する情報は、ステップ２１１０で決定されるように使用されるべき基底ベクトルの最大個数と、ステップ２１５０で決定されるように使用されるべき基底ベクトルの実際の数とを含むと考えることができる。方法２１００は終了し、プロセッサ２０５における制御は、ステップ２１６０からステップ１４９０に進み、ここで、ステップ１４９０で基底ベクトルのサブセットがビットストリーム内で符号化される。

【０２３２】

図２２は、ＣＮＮの第１部分からテンソルを符号化するために使用される基底ベクトルを復号するための方法２２００を示す。方法２２０は、ＰＣＳエンコーダ１６０によって符号化されたデータからテンソルを復号する際に使用される。方法２２００は、構成されたＦＰＧＡ、ＡＳＩＣ、またはＡＳＳＰなどの装置によって実施され得る。方法２１００は、付録Ａに存在する基底ベクトルパッキングフォーマット情報ＳＥＩメッセージシンタックスを参照して説明される。あるいは以下で説明されるように、方法２２００は、プロ

セッサ２０５の実行下で、アプリケーションプログラム２３３の１つまたは複数のソフトウェアコードモジュールとして、宛先デバイス１４０によって実施され得る。方法２２００は、ビットストリーム１４３において符号化されたビデオデータの各フレームに対して繰り返される。方法２２００を実施するアプリケーションプログラム２３３のソフトウェアコードモジュールは例えば、ハードディスクドライブ２１０および／またはメモリ２０６に記憶され得る。方法２２００は、最大基底ベクトルカウント復号ステップ２２１０で開始する。

【０２３３】

ステップ２２１０において、エントロピーデコーダ９２０は、プロセッサ２０５の実行下で、ＳＥＩメッセージ１３１３からｂｖ＿ｍａｘ＿ｃｎｔシンタックス要素を復号する。ｂｖ＿ｍａｘ＿ｃｎｔシンタックス要素は、ピクチャ１２００をパッキングする際に使用されることが可能な基底ベクトルの最大数を指定または示す情報を提供する。別個の最大値は、テンソルコンバイナ１６２から得られる結合表現内の各テンソルについて復号され得る。ステップ２２１０において、各結合表現の基底ベクトルの幅および高さもＳＥＩメッセージ１３１３から復号される。ｂｖ＿ｍａｘ＿ｃｎｔシンタックス要素は、ｂｖ＿ｍａｘ＿ｃｎｔの別のインスタンスがビットストリーム１４３から復号されることが示されるまで、ピクチャのシーケンスの第１ピクチャおよび後続のピクチャによる使用のために保持される値のために復号される。ｂｖ＿ｍａｘ＿ｃｎｔの別のインスタンスは、たとえばｂａｓｉｓ＿ｖｅｃｔｏｒ＿ｄｉｍｅｎｓｉｏｎｓ＿ｕｐｄａｔｅフラグを介して示され得る。プロセッサ２０５における制御は、ステップ２２１０から基底ベクトルパッキングフォーマット決定ステップ２２２０に進む。

【０２３４】

ステップ２２２０において、アンパッカ８２２は、プロセッサ２０５の実行下で、ステップ２２１０において決定された基底ベクトル幅、高さ、および最大個数に従って、サブピクチャ１２１６においてパッキングされたときの各基底ベクトルの空間位置を決定する。基底ベクトルは一般に、まず左から右の方法でパッキングされ、次に、上から下の方法でパッキングされ、基底ベクトルの順序付けられたリストを通って進む。基底ベクトルは一般に、隣接して、重複しない方法でパッキングされる。複数のレイヤが存在する場合、たとえば、ＦＰＮの使用のため、それぞれのレイヤの基底ベクトルの空間位置がサブピクチャ１２１６について決定される。ステップ２２２０は、ｂｖ＿ｍａｘ＿ｃｎｔがビットストリーム１４３から復号されるとき、すなわち、ｂａｓｉｓ＿ｖｅｃｔｏｒ＿ｄｉｍｅｎｓｉｏｎｓ＿ｕｐｄａｔｅフラグによって示されるときはいつでも実行される必要がある。ｂｖ＿ｍａｘ＿ｃｎｔがビットストリーム１４３から復号されないとき、方法２２００のより早い呼び出しにおいてステップ２２２０を実行することから決定されたパッキング構成が再使用され得る。プロセッサ２０５における制御は、ステップ２２２０から基底ベクトルサブセット復号ステップ２２３０に進む。

【０２３５】

ステップ２２３０において、エントロピーデコーダ９２０は、プロセッサ２０５の実行下で、ＳＥＩメッセージ１３１３からｂｖ＿ｃｎｔシンタックス要素を復号する。ｂｖ＿ｃｎｔシンタックス要素は、使用された基底ベクトルカウントの更新を示すＳＥＩメッセージ１３１３からも復号されたｂａｓｉｓ＿ｖｅｃｔｏｒ＿ｃｎｔ＿ｕｐｄａｔｅが宛先デバイス１４０に伝達される必要があるときに復号される。ステップ２２３０は、復号に使用される基底ベクトルの数を示す情報を復号するように動作する。使用されるべき基底ベクトルの数を指定する情報は、２２１０において導出されるようなベクトルの最大個数と、ステップ２２３０において復号される使用されるべき実際の量との両方を含むと考えることができる。テンソル１１５中の各レイヤは、別個の基底ベクトルカウントを有し得る。シグナリングされた基底ベクトルカウントは値Ｎを示し、ここで、Ｎは、定義された閾値に従ってテンソルを適切に再構成するために選択されたＭ個の基底ベクトルのうちの最初のＮ個の基底ベクトルであるサブセットを定義する。使用される基底ベクトルの数を確立するための閾値の使用は、テンソル１１５の統計への適応を可能にする。ｂｖ＿ｃｎ

ｔシンタックス要素は、フレームごとに、すなわち符号化データのフレームごとに、決定または復号することができる。

【０２３６】

指定された品質レベル、すなわち、所与の説明された分散しきい値に到達するために必要とされる基底ベクトルがより少ない場合、追加の基底ベクトルが省略され得、圧縮効率を改善する。基底ベクトルの大部分を保持するために、９８％または９９％などの閾値を使用することができる。ほとんどの場合、すべてのＭ個の基底ベクトルが保持され、すなわち、ＮはＭに等しいが、値Ｎはデータ依存方法で低減され得る。より低い品質が許容可能である場合、より良好なビットレートまたは性能トレードオフは量子化ステップサイズをさらに増加させるよりも優先的に閾値を減少させることによって、すなわち、量子化パラメータを増加させることによって達成される。例えば、８５％の閾値はビットレートを低減し、一方で、ＱＰをより高いレベル（例えば、４５より大きいＱＰ）に増加させるよりも、タスク性能を相対的に著しく低下させることは、非常にビット圧縮されたレートが望まれる場合に好適であり得る。９５％の閾値は、達成可能なタスク性能の中程度の低減でビットレートを低減し、無損失に近い圧縮と非常に低いビットレート動作点の圧縮と間の性能を必要とするユースケースに適している。閾値およびＱＰは、「パレートフロント（Ｐａｒｅｔｏ ｆｒｏｎｔ）」、すなわち、最適な閾値およびＱＰ値からなるビットレートまたはタスク性能フロンティア曲線を生成するように、一緒に最適化され得る。このパレートフロントを導出するために様々な閾値およびＱＰ値を徹底的にテストすることは、一般に、ソースデバイス１１０にとって法外に高価であり、適切な値を示すための発見的規則が望ましい。実験は、画素当たりの下位ビット（ＢＰＰ）について、ＱＰを増加させる代わりに閾値を減少させることが好ましいことを示している。さらに、しきい値は異なる符号化レイヤ、たとえば、ＦＰＮの、またはテンソルコンバイナ１６２から得られるＦＰＮの結合表現に対して変更され得る。実験は、より多くの分解されたレイヤがわずかに高い閾値を与えられ得、ビットレートの比較的制限された増加においてタスク性能の改善を達成することを示す。たとえば、テンソル１１５のＰ２／Ｐ３連結レイヤ、Ｐ４、およびＰ５レイヤの９０％、９４％、および９８％の閾値は、すべてのレイヤにわたって適用される９５％の９０％などの固定閾値よりも良好な性能を提供する。より分解されたレイヤ（例えば、Ｐ４およびＰ５）はより分解されていないレイヤ（例えば、Ｐ２またはＰ３）と比較して、より小さいエリアを有する。したがって、サブピクチャ１２１６中のパッキングされたエリアの増分増加は、より分解されていないレイヤについての同様の追加の基底ベクトルカウントよりも小さい。最大基底ベクトルカウントは同様に、レイヤごとに変更され得る。プロセッサ２０５における制御は、ステップ２２３０から基底ベクトルサブピクチャ復号ステップ２２４０に進む。

【０２３７】

ステップ２２４０において、ピクチャデコーダ８０４は、プロセッサ２０５の実行下で、パッキングされた基底ベクトルを含むサブピクチャ１２１６を復号する。ステップ２２４０は、少なくともステップ２２３０で復号された情報に従って、符号化されたデータから基底ベクトルを効果的に導出する。プロセッサ２０５における制御は、ステップ２２４０から基底ベクトルアンパックステップ２２５０に進む。

【０２３８】

ステップ２２５０において、アンパッカ８２２は、プロセッサ２０５の実行の下で、ステップ２２３０においてピクチャ１２００において使用されるようにシグナリングされた基底ベクトルの数に従って、サブピクチャ１２１６から数ｂｖ＿ｃｎｔ基底ベクトルをアンパックする。ピクチャ１２００において使用されるようにシグナリングされる基底ベクトルの数は、ステップ２２２０において決定されたパッキング構成によって与えられる基底ベクトルの最大個数未満であり得る。ステップ２２５０は、少なくとも使用されるべき基底ベクトルの数を指定する情報に従って、符号化データから基底ベクトルを導出する。複数のレイヤがサブピクチャ１２１６にパッキングされる場合、アンパッキングは、ＳＥＩメッセージ１３１３から取得されたそれぞれの基底ベクトルカウントｂｖ_ｃｎｔ[ｌ

ａｙｅｒ＿ｉｄｘ]値に従って各レイヤについて動作する。方法２２００は終了し、プロセッサ２０５における制御は方法１５００に戻り、ステップ１５５０において処理を再開する。

【０２３９】

図２３は、ネットワーク内の分割点における異なるレイヤのテンソルと、テンソルコンバイナ１６２などのテンソルコンバイナから符号化テンソルから再生するための結果として得られるオプションと、の間の関係を示す概略ブロック図２３００である。結合テンソル２３１０は、ｓｒｃ＿ｌａｙｅｒ＿ｉｄｘによって識別されるテンソル１４９ａのテンソルである。結合テンソル２３１０から、１つまたは２つの出力テンソル、すなわちテンソル２３２０および２３３０のいずれかが、テンソル１４９の一部として出力される。テンソル２３２０は、テンソル１４９内の位置ｓｒｃ＿ｌａｙｅｒ＿ｉｄｘに留まり、テンソル２３３０は、存在する場合、テンソル１４９内の位置ｄｓｔ＿ｌａｙｅｒ＿ｉｄｘに挿入される。図２３は、仮に存在した場合、ＳＥＩメッセージから復号され、付録Ａを参照して説明されたフラグ２３１５（すなわち、layer_extraction_flag,layer_upsample_flag,layer_downsample_flag,resample_vertical_only_flag,resample_horizontal_only_flag）に基づくテンソル２３２０の幅および高さと比較した、テンソル２３３０の空間的関係を示す。フラグ２３１５は、ステップ１９１０で決定され、ステップ１９２０でレイヤマッピング１３９０として符号化される。フラグ２３１５は、ステップ２０１０において、テンソル２３２０および存在する場合にはテンソル２３３０を生成するために復号される。結合テンソル２３１０は、ｋ１＋ｋ２チャネルを有し、ここで、ｋ１はテンソル２３２０のチャネルに対応し、ｋ２はテンソル２３３０のチャネルに対応し、境界はｓｒｃ＿ｃｈａｎｎｅｌ＿ｏｆｆｓｅｔを使用してシグナリングされる。フラグ２３１５は、抽出されたテンソルのための様々な再サンプリングオプションを可能にするためのシグナリングを提供する。テンソル１４９はテンソル１１５ａと同じ次元を有する必要があるので、使用され得るフラグ２３１５の利用可能な組合せは、ネットワークおよび分割点に依存するテンソル１１５ａの次元に依存する。

【０２４０】

ソースデバイス１１０および宛先デバイス１４０の構成では、パックまたはアンパックされるべき基底ベクトルの数、すなわちｂｖ＿ｃｎｔは、パッキング構成によって与えられる基底ベクトルの最大個数に対して相対的に、すなわちｂｖ＿ｍａｘ＿ｃｎｔに対して相対的に表される。一般に、最大利用可能基底ベクトル（すなわち、ｂｖ＿ｍａｘ＿ｃｎｔ）の大部分が使用され得、したがって、ｂｖ＿ｍａｘ＿ｃｎｔとｂｖ＿ｃｎｔとの間の差は比較的小さくなり得る。ｂｖ＿ｍａｘ＿ｃｎｔとｂｖ＿ｃｎｔとの間の差が小さいとき、指数Ｇｏｌｏｍｂシンタックス要素を使用してｂｖ＿ｍａｘ＿ｃｎｔとｂｖ＿ｃｎｔとの間の差を符号化することによって、圧縮効率が改善され得る

【０２４１】

ＰＣＡエンコーダ１６０の構成では、それぞれ基底ベクトルに適用可能な平均係数および係数のためのサブピクチャエンコーダ５３２および５７２が、ＱＰコントローラ７９０によって、それぞれサブピクチャエンコーダ５１４および５５６によって平均特徴または基底ベクトルを符号化するために適用されるよりも低い量子化パラメータを適用するように構成される。サブピクチャエンコーダ５３２および５７２内のＱＰコントローラモジュール７９０のそれぞれのインスタンスは、ベースＱＰにオフセットを適用することができ、ベースＱＰは、基底ベクトルを符号化するＣＵなどの他の場所に適用される。ＱＰオフセットは、サブピクチャ１２１２および１２１４などにおいて、基底ベクトルに適用可能な平均係数または係数の残差を符号化するときに使用される量子化ステップサイズを低減するために、マイナス６またはマイナス１２、マイナス１８、またはマイナス２４などの値、または他の値を有し得る。サブピクチャ符号化係数の量子化ステップサイズを低減するためにＱＰオフセットを適用することは、ＰＣＡエンコーダ１６０中の順変換（たとえば、５２４および５６４）によって生成された係数と比較して、ＰＣＡデコーダ１７０中で生成された復号された係数間の不一致を低減する。係数は、再構成された特徴マップ上

のそれぞれの平均特徴または基底ベクトルの寄与を変調する際の係数の役割によって、特徴マップの再構成に対して乗法的な効果を有する。したがって、係数に対する非可逆符号化の影響は、再構成された特徴マップにおける忠実度の実質的な損失を引き起こし、ＣＮＮヘッド１５０におけるタスク性能に悪影響を及ぼす。したがって、係数は、基底ベクトルよりも高い品質レベルで符号化される必要がある。係数を表すことを担うサブピクチャまたはスライスに、固定の比較的低いＱＰを適用することも可能である。係数がピクチャ１２００全体の比較的小さいエリアを占めるので、低いＱＰの使用による使用されるビットレートの増加は比較的小さく、いずれの場合も、再構成された特徴マップの忠実度を改善する。ＱＰは「量子化グループ」（ＱＧ）として知られるエリアを指定することによって、スライスレベルよりも細かい粒度で変更することができ、これは、「デルタＱＰ」シンタックス要素を使用してＱＰを変更することができる各ＣＴＵ内のエリアの最大粒度に対応する。したがって、係数および基底ベクトルが同じスライス中で符号化されるべきである場合、デルタＱＰは、基底ベクトルのために必要とされるよりも高い精度で係数を符号化するための別の機構を与える。

【０２４２】

ピクチャ１２００の構成では、サブピクチャではなくタイルへの分割が使用される。タイルを用いた構成では、平均特徴マップ１２２０、平均係数１２２２、基底ベクトル１２２６、および基底ベクトル係数１２２４が別々のタイルに含まれる。タイルは、ＰＣＡエンコーダ１６０において独立して符号化され、ＰＣＡデコーダ１７０において復号され得る。

【０２４３】

ＰＣＡエンコーダ１６０およびＰＣＡデコーダ１７０の別の構成では、平均特徴がテンソル１１５のすべての特徴マップに均一に適用される。言い換えれば、すべての平均係数は、ドット積モジュール５２４によって特徴マップごとに導出されるのではなく、「１」の値が割り当てられる。平均係数が「１」に割り当てられる場合、ＰＣＡエンコーダ１６０では、再構成された平均特徴マップ５２２が平均係数によってスケーリングされる必要がなく、したがって、モジュール５２４、５２８、５３２、５３６、５４０は必要なく、５２２は５４２としてモジュール５４４に直接渡される。ＰＣＡデコーダ１７０では、モジュール８３２、８３６、および８５０は必要とされず、復号された平均特徴マップ８４８は８５２として合計モジュール８５６に渡される。平均係数が「１」の値を割り当てられるとき、サブピクチャ１２１２は、一定の平均係数値を符号化する必要がないので、ピクチャ１２００から省略され得る。サブピクチャ１２１２がピクチャ１２００から省略されるとき、基底ベクトルのための係数を含むサブピクチャ１２１４は、係数をパッキングするために必要とされる最小幅と比較して、幅が増加され得る。サブピクチャ１２１４の幅は、パッキングされた係数の幅（すなわち、パッキングされたテンソル１１５のレイヤ中のチャネルの最大数）の１．５倍に設定され得る。サブピクチャ１２１４の幅を増加させると、より広いアスペクト比を有するピクチャ１２００が得られ、これは、個々の基底ベクトルの一般的に遭遇するアスペクト比により適しており、したがって、ピクチャ１２００内の未使用エリアがより少なくなる。サブピクチャ１２１０、１２１４、および１２１５は、ピクチャ１２００の全体がサブピクチャによって占有されるように、ピクチャ構造を維持するために幅が調整され得る。一定の平均係数の使用は、また、関連する量子化および逆量子化の必要性を除去する。一定の平均係数の使用は、依然として、順変換および逆変換（射影）の可逆動作を維持するために必要な、導出された基底ベクトルに対する順変換（ドット積）の前にテンソル１１５を「ゼロセンタリング」する目的に役立つ。

【０２４４】

本明細書で提示される方法は、ＶＶＣまたはＨＥＶＣなどの現代のブロックベースの圧縮規格を使用して、圧縮に適したフォーマットでテンソルを効率的に表現することを可能にする。ブロックベースの圧縮は、特徴マップを再構成するために基底ベクトルを投影するための係数などのデータに直感的に適用可能ではないが、トレーニングされた二次変換を含む様々な変換を使用することなどによって、ブロック内の追加の予期しない冗長性を

明らかにする。本明細書に提示される方法は、「高速ＲＣＮＮ」および「ＹＯＬＯｖ３」ネットワークアーキテクチャ、ならびに「バックボーン」および「ヘッド」部分へのこれらのネットワークの特定の分割を参照して説明されるが、本方法は多次元テンソルデータについて動作する任意のニューラルネットワークに適用可能であり、そのようなネットワークの「バックボーン」および「ヘッド」部分への異なる分割に適用可能である。

【０２４５】

産業上の利用可能性

記載された構成は、コンピュータおよびデータ処理産業に適用可能であり、特に、高い圧縮効率を達成する、ビデオおよび画像信号などの信号の符号化および復号のためのデジタル信号処理に適用可能である。

【０２４６】

チャネルまたは特徴マップのグループ内の浮動小数点テンソルデータを量子化し、対数量子化領域を使用して、得られた整数値を平面フレームにパッキングするための構成も開示される。対数量子化領域を使用する量子化および逆量子化方法は、大きな大きさのテンソル値のための正確な値を符号化するのに費やされるビットがないことに起因して、より大きな圧縮効率を可能にし、そのような精度は、使用中のネットワークのためのタスク性能のさらなる改善をもたらさない。

【０２４７】

上記は本発明のいくつかの実施形態のみを説明し、本発明の範囲および趣旨から逸脱することなく、修正および／または変更を行うことができ、実施形態は例示的であり、限定的ではない。

【０２４８】

付録Ａ

ビットストリームにおける基底ベクトルパッキング（すなわち、１３９１）に関連するメタデータを表すための例示的なＳＥＩメッセージフォーマットおよび関連するセマンティクスは、以下の通りである：

【０２４９】

【表１】

basis_vector_packing_info(payloadSize) {	Descriptor
basis_vector_dimensions_update	u(1)
if(basis_vector_dimensions_update != 0) {	
layer_cnt	u(1)
for(layer_idx = 0; layer_idx < layer_cnt; layer_idx++) {	
bv_max_cnt[layer_idx]	ue(v)
bv_width[layer_idx]	ue(v)
bv_height[layer_idx]	ue(v)
}	
}	
basis_vector_cnt_update	u(1)
if(basis_vector_cnt_update)	
{	
for(layer_idx = 0; layer_idx < layer_cnt; layer_idx++) {	
bv_cnt[layer_idx]	ue(v)
}	
}	
layer_update_mapping_info	u(1)
if(layer_update_mapping_info)	
{	
layer_extraction_flag	u(1)
if(layer_extraction_flag)	
{	
layer_upsample_flag	u(1)
layer_downsampling_flag	u(1)
if(layer_upsample_flag || layer_downsampling_flag)	
{	
resampling_filter_idx	ue(v)
resample_vertical_only_flag	u(1)
resample_horizontal_only_flag	u(1)
}	
src_layer_idx	ue(v)
src_channel_offset	ue(v)
dst_layer_idx	ue(v)
}	
}	
}	

【０２５０】

基底ベクトルパッキング情報セマンティクス

上記のシンタックス構造は、基底ベクトル平面フレームをアンパックし、推論タスクを実行するために復号テンソルに投影するための基底ベクトルに変換するために必要な情報を指定する。

【０２５１】

記述子ｕ（ｎ）を有するシンタックス要素は、シンタックス要素がｎビットを使用して符号化され、符号なし整数値として解釈されることを示す。記述子ｕｅ（ｖ）を有するシンタックス要素は、シンタックス要素が指数ゴロム値として符号化され、符号なし整数値

として解釈されることを示す。
【０２５２】
基底ベクトルパッキング情報ＳＥＩメッセージの持続性は、関連するＡＵから、基底ベクトルパッキング情報ＳＥＩメッセージの次の発生またはＣＬＶＳのエンドまでである。基底ベクトルパッキング情報ＳＥＩメッセージの次の発生時に、パラメータは、更新フラグに基づいてグループ化される。更新されない（すなわち、上記のシンタックステーブルのブランチ内に含まれ、そのブランチの更新フラグがゼロに設定されている）パラメータは、それらの以前の値（存在する場合）を保持する。
【０２５３】
１に等しいｂａｓｉｓ＿ｖｅｃｔｏｒ＿ｄｉｍｅｎｓｉｏｎｓ＿ｕｐｄａｔｅは、現在および後続のフレームに適用可能な、すなわち１に等しいｂａｓｉｓ＿ｖｅｃｔｏｒ＿ｄｉｍｅｎｓｉｏｎｓ＿ｕｐｄａｔｅを有する基底ベクトルパッキング情報ＳＥＩメッセージの次の発生までの、最大基底ベクトルカウントおよび基底ベクトル次元の更新を指定する。ＣＬＶＳにおける基底ベクトルパッキング情報ＳＥＩメッセージの第１のインスタンスが１に等しいｂａｓｉｓ＿ｖｅｃｔｏｒ＿ｄｉｍｅｎｓｉｏｎｓ＿ｕｐｄａｔｅを有することは、ビットストリーム適合性の要件である。
【０２５４】
メモ：通常、１に等しいｂａｓｉｓ＿ｖｅｃｔｏｒ＿ｄｉｍｅｎｓｉｏｎｓ＿ｕｐｄａｔｅを有する基底ベクトルパッキング情報ＳＥＩメッセージは、ＣＬＶＳの第１ピクチャと共に送信される。
【０２５５】
１に等しいｂａｓｉｓ＿ｖｅｃｔｏｒ＿ｃｎｔ＿ｕｐｄａｔｅは、ＣＬＶＳ内の現在および後続のフレームに適用可能な、使用される基底ベクトルカウントの更新を指定する。使用される基底ベクトルカウントは、最大基底ベクトルの基底ベクトルが何個使用されているかを示す。
【０２５６】
ｌａｙｅｒ＿ｃｎｔは、フレーム内に存在するレイヤの数を指定する。
【０２５７】
ｂｖ＿ｍａｘ＿ｃｎｔ［ｌａｙｅｒ＿ｉｄｘ］は、ｌａｙｅｒ＿ｉｄｘについて存在する基底ベクトルの最大数の数を指定する。
【０２５８】
ｂｖ＿ｗｉｄｔｈ［ｌａｙｅｒ＿ｉｄｘ］は、ｌａｙｅｒ＿ｉｄｘの基底ベクトルの幅を指定する。
【０２５９】
ｂｖ＿ｈｅｉｇｈｔ［ｌａｙｅｒ＿ｉｄｘ］は、ｌａｙｅｒ＿ｉｄｘの基底ベクトルマップの高さを指定する。
【０２６０】
ｂｖ＿ｃｎｔ［ｌａｙｅｒ＿ｉｄｘ］は、ｌａｙｅｒ＿ｉｄｘについて存在する基底ベクトルの数の数を指定する。０からｌａｙｅｒ＿ｃｎｔ－１までの範囲における各値ｌａｙｅｒ＿ｉｄｘについて、ｂｖ＿ｃｎｔ［ｌａｙｅｒ＿ｉｄｘ］が、ｂｖ＿ｍａｘ＿ｃｎｔ［ｌａｙｅｒ＿ｉｄｘ］以下であることは、ビットストリーム適合性の要件である。ｂｖ＿ｃｎｔ［ｌａｙｅｒ＿ｉｄｘ］がビットストリームから復号されていないとき、値ｂｖ＿ｃｎｔ［ｌａｙｅｒ＿ｉｄｘ］は、０からｌａｙｅｒ＿ｃｎｔ－１までの範囲のｌａｙｅｒ＿ｉｄｘについてｂｖ＿ｍａｘ＿ｃｎｔ［ｌａｙｅｒ＿ｉｄｘ］に等しく設定される。
【０２６１】
１に等しいｌａｙｅｒ＿ｕｐｄａｔｅ＿ｍａｐｐｉｎｇ＿ｉｎｆｏは、テンソルコンバイナ１６２によって決定され、テンソルセパレータ１７２によって実行されるテンソル再マッピングの構成に対する更新を指定する。ｌａｙｅｒ＿ｕｐｄａｔｅ＿ｍａｐｐｉｎｇ＿ｉｎｆｏがゼロに等しいとき、テンソル再マッピングの構成に対する更新は行われず、

テンソル再マッピングの任意の以前の構成が適用され続ける。

【０２６２】

１に等しいｌａｙｅｒ＿ｅｘｔｒａｃｔｉｏｎ＿ｆｌａｇは、テンソル内の最後のチャネルに対するチャネルｓｒｃ＿ｃｈａｎｎｅｌ＿ｏｆｆｓｅｔに対応するテンソルの一部分が、ｄｓｔ＿ｌａｙｅｒ＿ｉｄｘにおけるテンソルのリストに挿入されることを指定し、新しいレイヤを生成するために、レイヤｓｒｃ＿ｌａｙｅｒ＿ｉｄｘからのテンソルが抽出される。ｌａｙｅｒ＿ｅｘｔｒａｃｔｉｏｎ＿ｆｌａｇがゼロに等しいとき、テンソルの抽出は実行されず、その結果、テンソル１４９はテンソル１４９ａに等しく設定される。ビットストリームｌａｙｅｒ＿ｅｘｔｒａｃｔｉｏｎ＿ｆｌａｇから以前に解析されていない場合、ゼロであると推論される。

【０２６３】

１に等しいｌａｙｅｒ＿ｕｐｓａｍｐｌｅ＿ｆｌａｇは、抽出されるテンソルの部分がアップサンプリングされて、より大きい、例えば、入力テンソルの２倍の幅および２倍の高さを有する新しいテンソルを生成することを指定する。

【０２６４】

１に等しいｌａｙｅｒ＿ｄｏｗｎｓａｍｐｌｅ＿ｆｌａｇは、抽出されるテンソルの部分がダウンサンプリングされて、より小さい、例えば、入力テンソルの幅の半分および高さの半分を有する新しいテンソルを生成することを指定する。

【０２６５】

ｌａｙｅｒ＿ｕｐｓａｍｐｌｅ＿ｆｌａｇおよびｌａｙｅｒ＿ｄｏｗｎｓａｍｐｌｅ＿ｆｌａｇがともに、基底ベクトルパッキング情報ＳＥＩメッセージの１つのインスタンスにおける１つに等しくないことは、ビットストリーム適合性の要件である。

【０２６６】

ｒｅｓａｍｐｌｉｎｇ＿ｆｉｌｔｅｒ＿ｉｄｘは、新しいテンソルを生成するときにテンソルの部分を再サンプリングする際に適用されるフィルタリングのタイプを指定する。再サンプリングは、ｒｅｓａｍｐｌｅ＿ｖｅｒｔｉｃａｌ＿ｏｎｌｙ＿ｆｌａｇおよびｒｅｓａｍｐｌｅ＿ｈｏｒｉｚｏｎｔａｌ＿ｏｎｌｙ＿ｆｌａｇに従って、水平または垂直または水平および垂直に適用される。次の表に、使用可能なフィルタを列挙する：

【０２６７】

【表２】

resampling_filter_idx	ダウンサンプリング時	アップサンプリング時
0	デシメーション	バイリニアフィルタ
1	アンチエイリアシング、デシメーション	バイキュービックフィルタ
2	将来使用するため予約	最近傍
3-	将来使用するため予約	将来使用するため予約

【０２６８】

１に等しいｒｅｓａｍｐｌｅ＿ｖｅｒｔｉｃａｌ＿ｏｎｌｙ＿ｆｌａｇは、リサンプリング動作が垂直方向にのみ適用されることを指定する。

【０２６９】

１に等しいｒｅｓａｍｐｌｅ＿ｈｏｒｉｚｏｎｔａｌ＿ｏｎｌｙ＿ｆｌａｇは、リサンプリング動作が水平方向にのみ適用されることを指定する。

【０２７０】

ｒｅｓａｍｐｌｅ＿ｖｅｒｔｉｃａｌ＿ｏｎｌｙ＿ｆｌａｇおよびｒｅｓａｍｐｌｅ＿ｈｏｒｉｚｏｎｔａｌ＿ｏｎｌｙ＿ｆｌａｇは両方とも、基底ベクトルマッピング情報ＳＥＩメッセージのいずれのインスタンスにおいても１に等しくないことが、ビットストリ

ーム適合の要件である。

【０２７１】

ｓｒｃ＿ｌａｙｅｒ＿ｉｄｘは、テンソル１４９に含まれる新しいテンソルを生成するために、再サンプリングのために、テンソル１４９ａの部分のどのテンソルから抽出されるべきかを指定する。

【０２７２】

ｓｒｃ＿ｃｈａｎｎｅｌ＿ｏｆｆｓｅｔは、ｓｒｃ＿ｌａｙｅｒ＿ｉｄｘによって識別されるテンソルが分割されるチャネルオフセットを指定する。ｓｒｃ＿ｌａｙｅｒ＿ｉｄｘによって指定されたテンソルのチャネル０～ｓｒｃ＿ｃｈａｎｎｅｌ＿ｏｆｆｓｅｔ－１が保持され、テンソル１４９の対応するテンソルとして出力される。ｓｒｃ＿ｌａｙｅｒ＿ｉｄｘによって指定されるテンソルの最後のチャネルのチャネルｓｒｃ＿ｃｈａｎｎｅｌ＿ｏｆｆｓｅｔは、再サンプリングの対象となり、その結果は、テンソル１４９内の追加のテンソルとして出力される。

【０２７３】

ｄｓｔ＿ｌａｙｅｒ＿ｉｄｘは、リサンプリングされたテンソルがテンソル１４９ａに挿入されてテンソル１４９が生成されるインデックスを指定する。インデックス値０は開始時の挿入を指定し、すなわち、新しい最初の場所を作成し、ｌａｙｅｒ＿ｃｎｔに等しいインデックス値は、最後の場所の後の挿入を指定し、すなわち、新しいテンソルを追加して、新しい最後の場所を作成する。１からｌａｙｅｒ＿ｃｎｔ－１までのインデックス値は、テンソル１４９を作成するために、位置ｄｓｔ＿ｌａｙｅｒ＿ｉｄｘおよびｄｓｔ＿ｌａｙｅｒ＿ｉｄｘ＋１におけるテンソル１４９ａの間の位置への再サンプリングされたテンソルの挿入を示す。

【図面】

【図１】

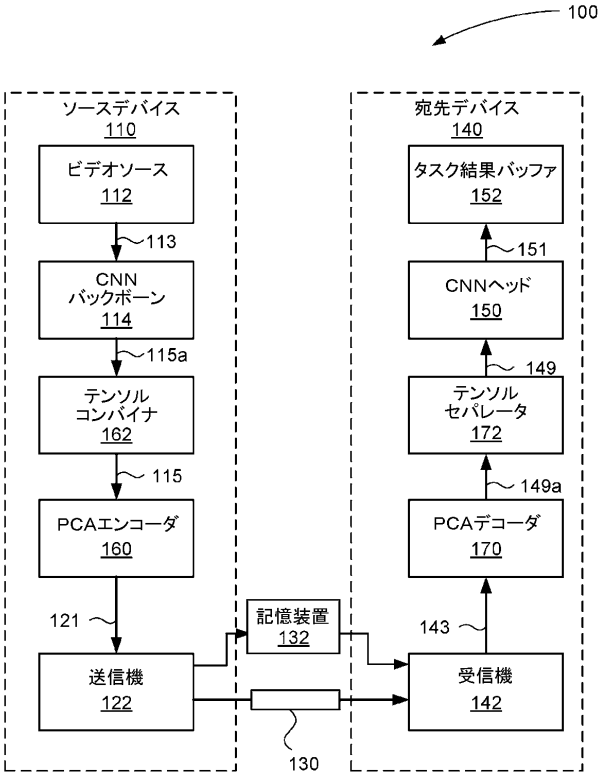


Fig. 1

【図２Ａ】

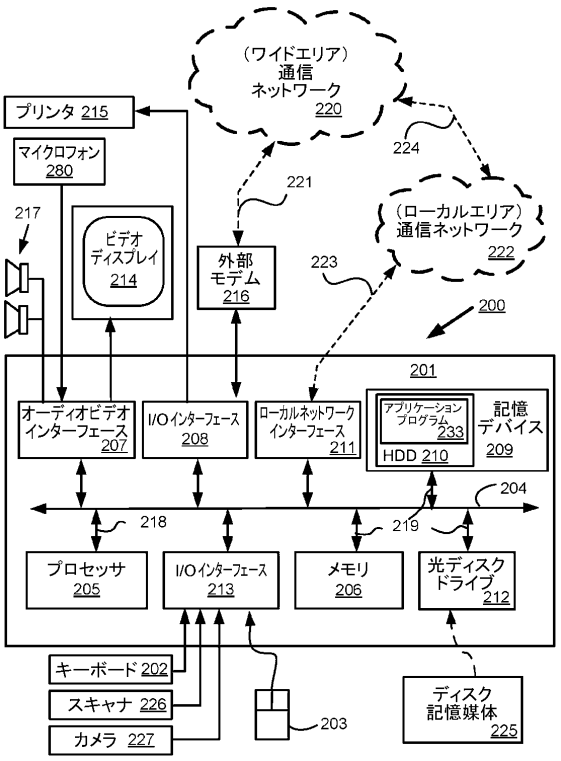


Fig. 2A

【図２Ｂ】

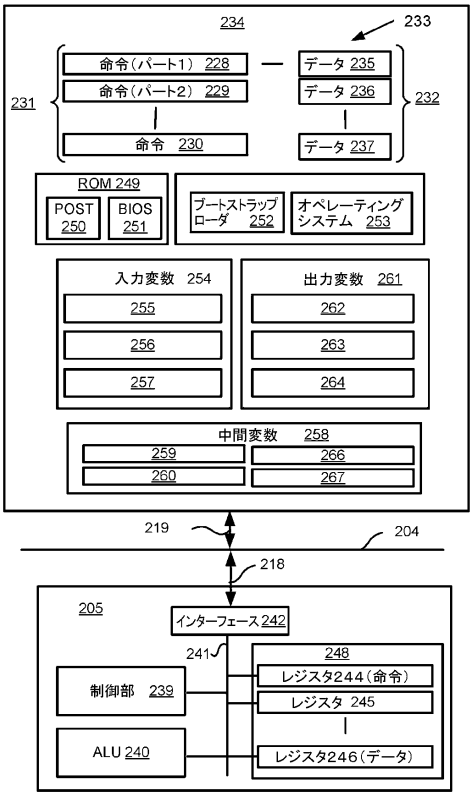


Fig. 2B

【図３Ａ】

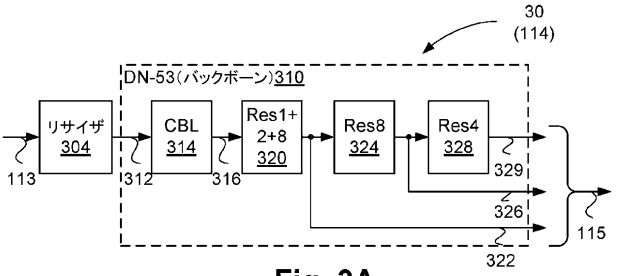


Fig. 3A

【図３Ｂ】

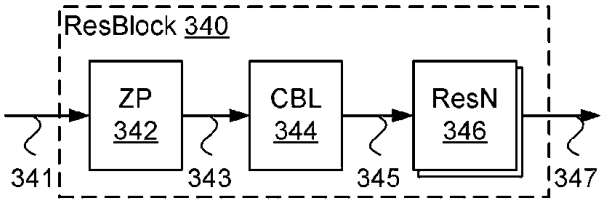


Fig. 3B

【図３Ｃ】

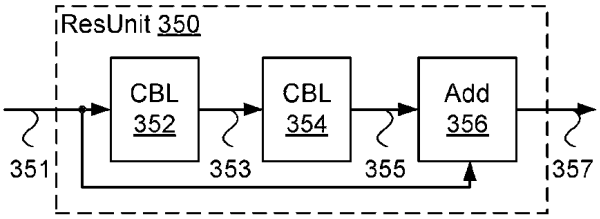


Fig. 3C

【図３Ｄ】

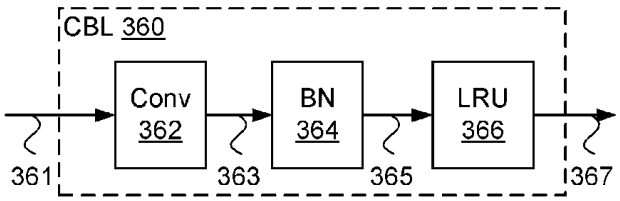


Fig. 3D

【図４】

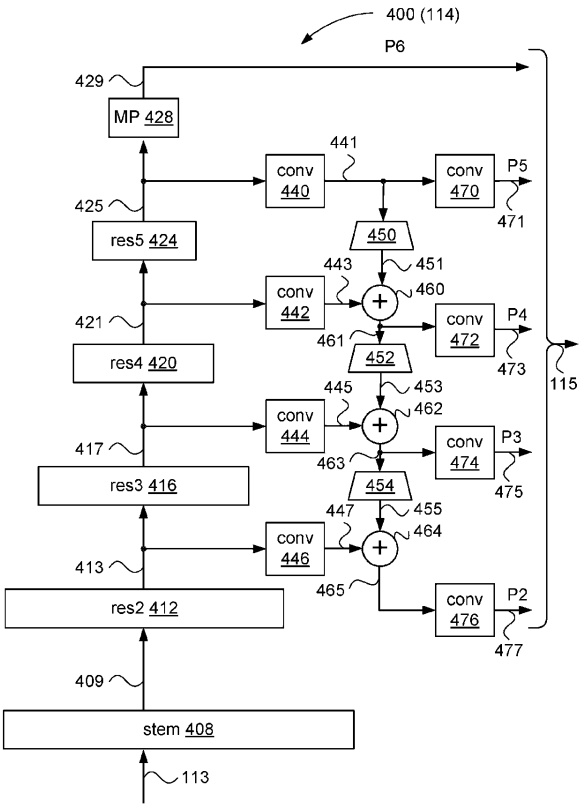


Fig. 4

【図５】

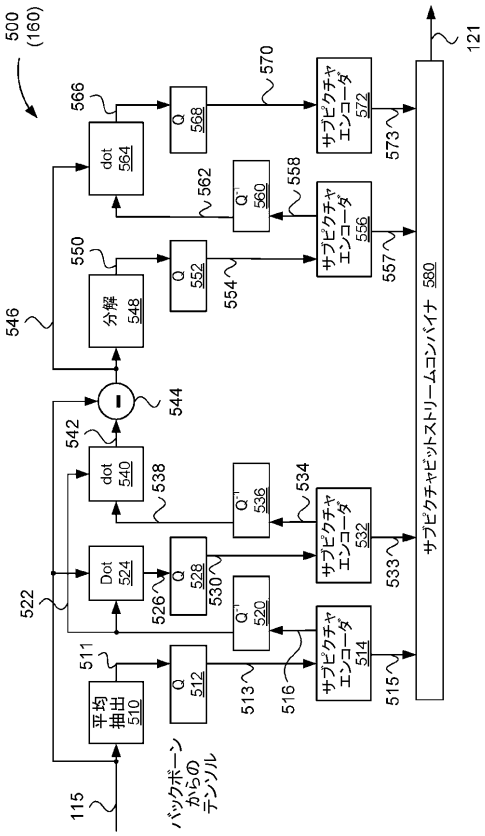


Fig. 5

【図６】

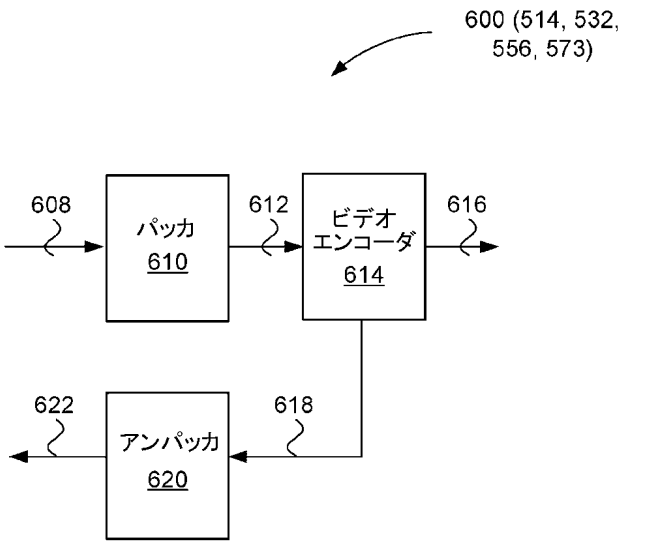


Fig. 6

【図７】

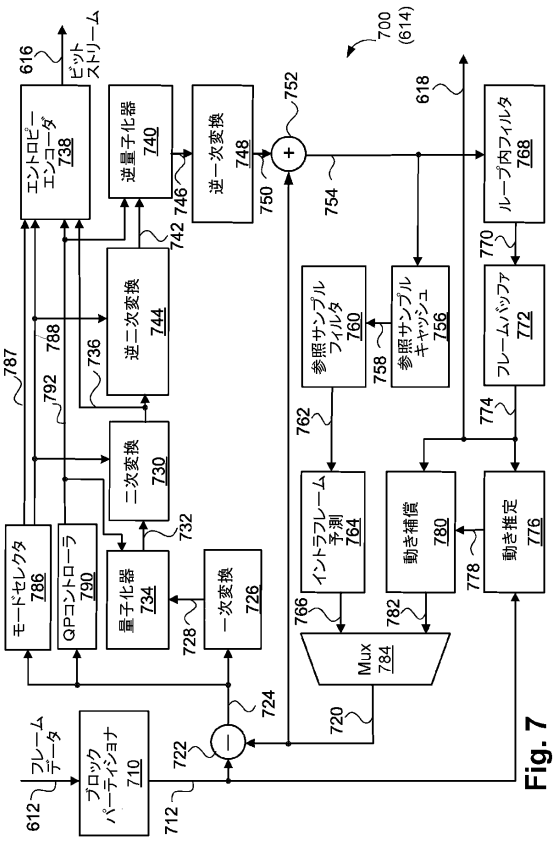


Fig. 7

【図８】

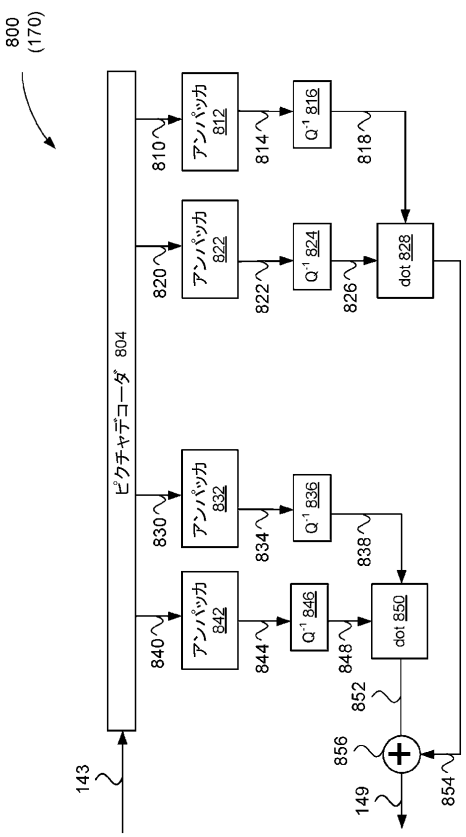


Fig. 8

【図９】

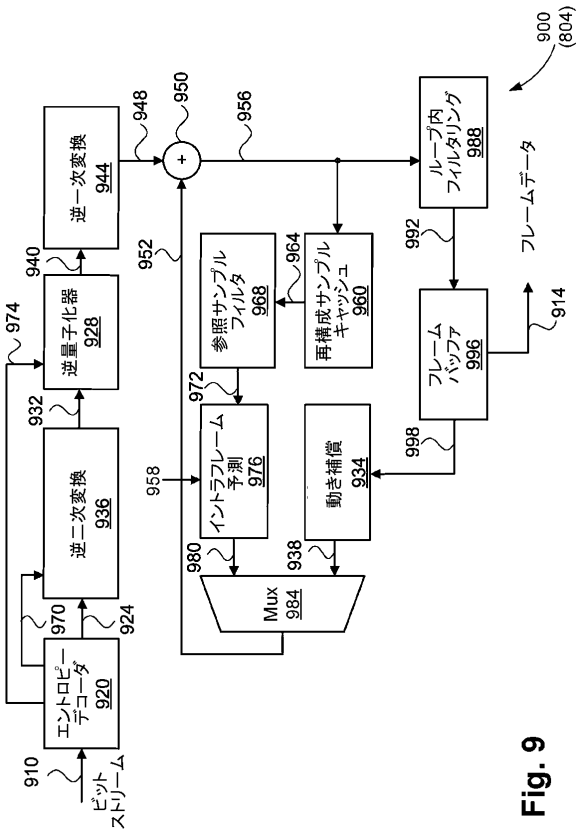


Fig. 9

【図１０Ａ】

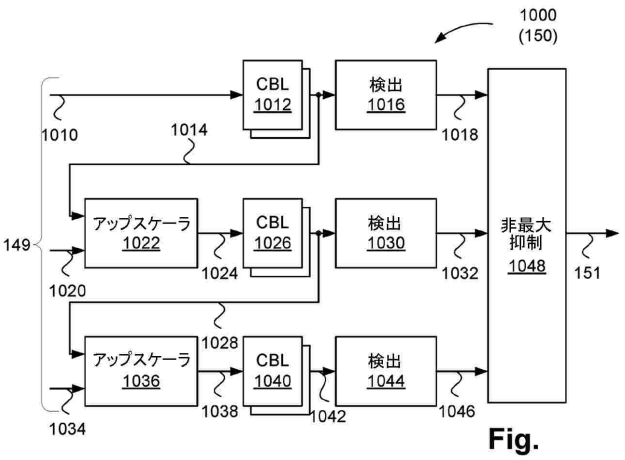


Fig. 10A

【図１０Ｂ】

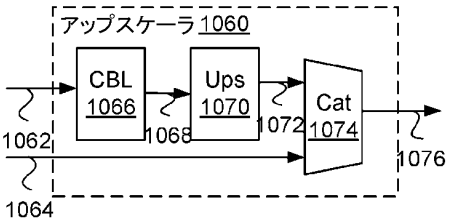


Fig. 10B

【図１０Ｃ】

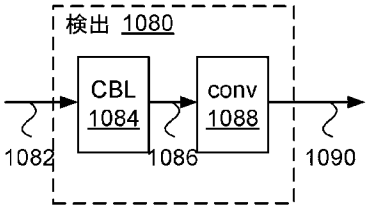


Fig. 10C

【図１１】

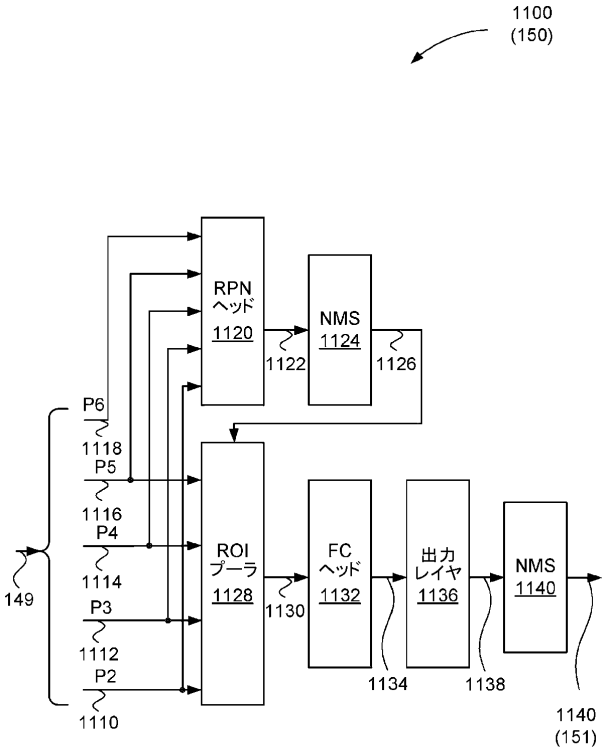


Fig. 11

【図１２Ａ】

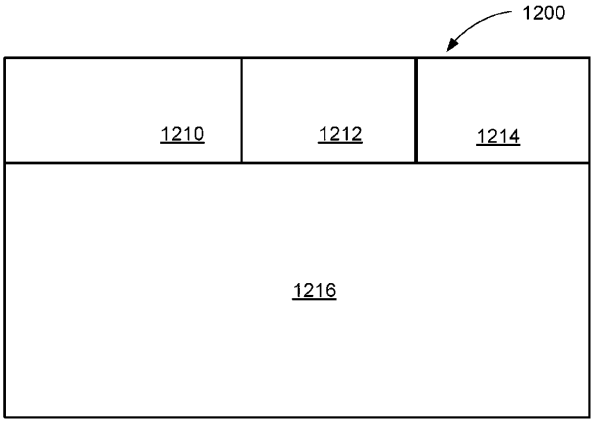


Fig. 12A

【図１２Ｂ】

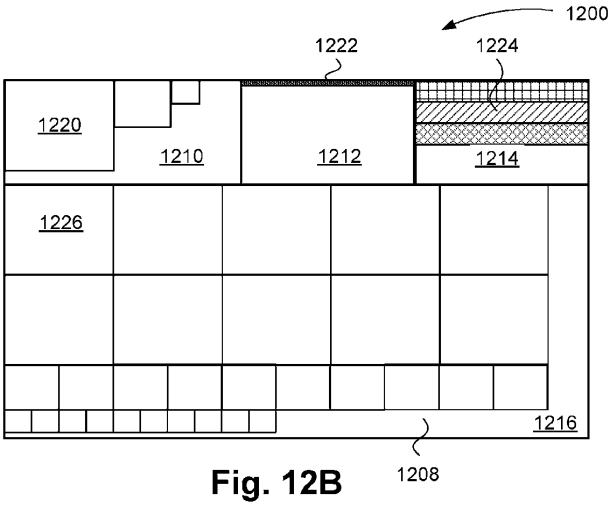


Fig. 12B

【図１３】

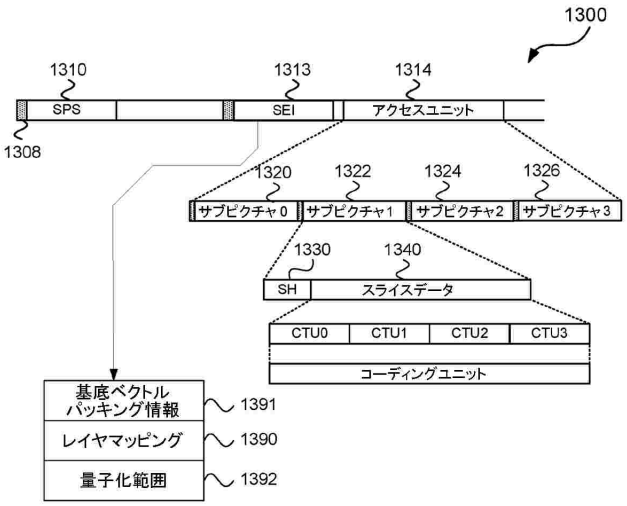


Fig. 13

【図１４】

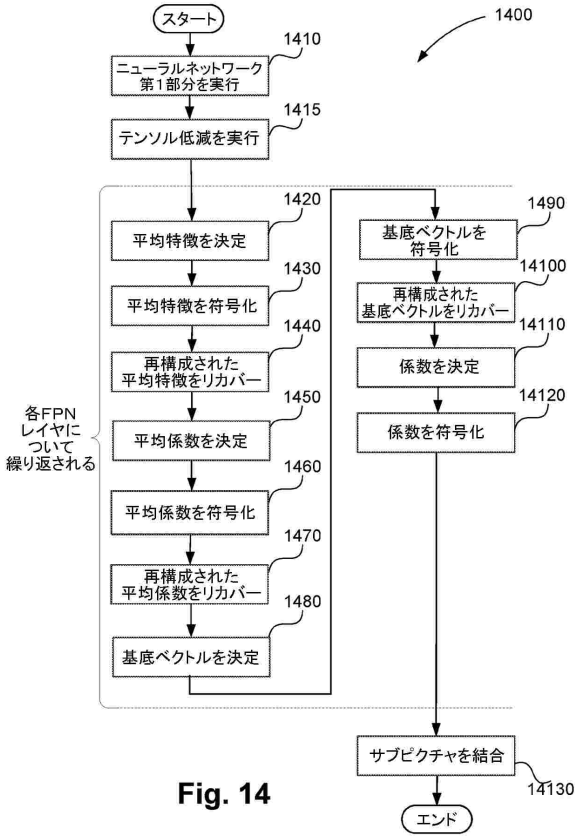


Fig. 14

【図１５】

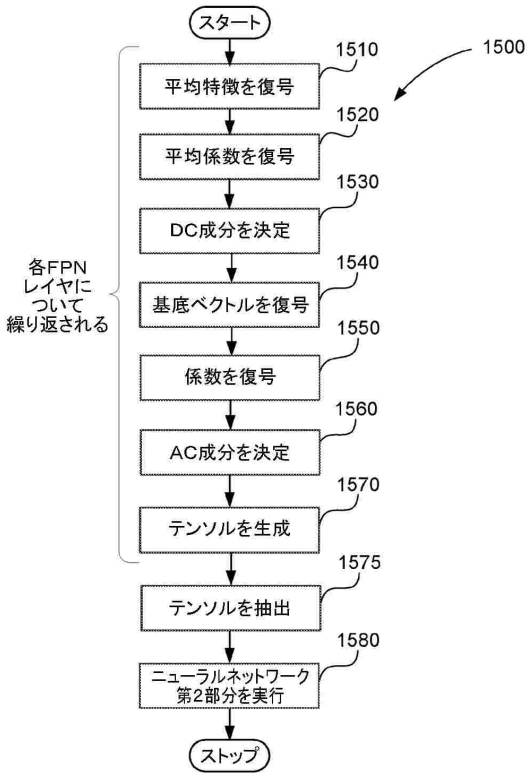


Fig. 15

【図１６Ａ】

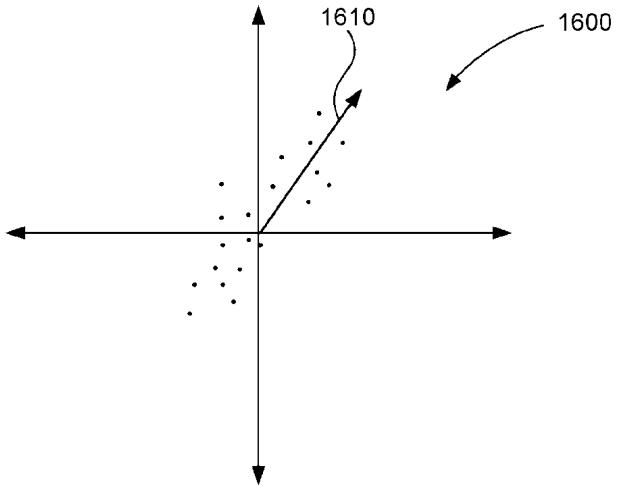


Fig. 16A

【図１６Ｂ】

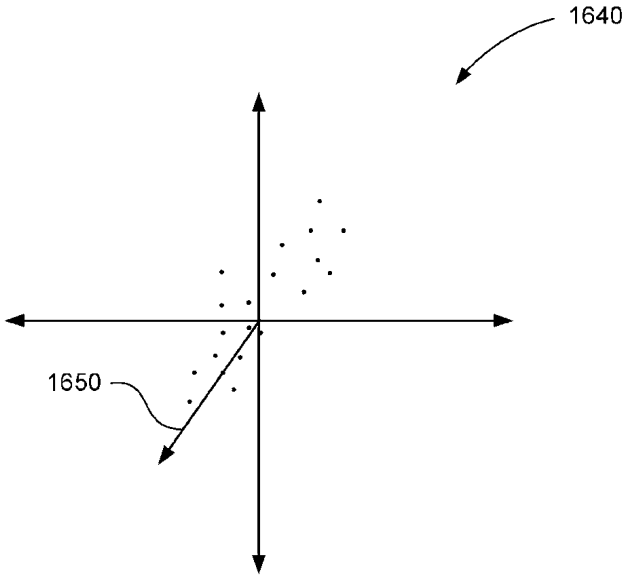


Fig. 16B

【図１７】

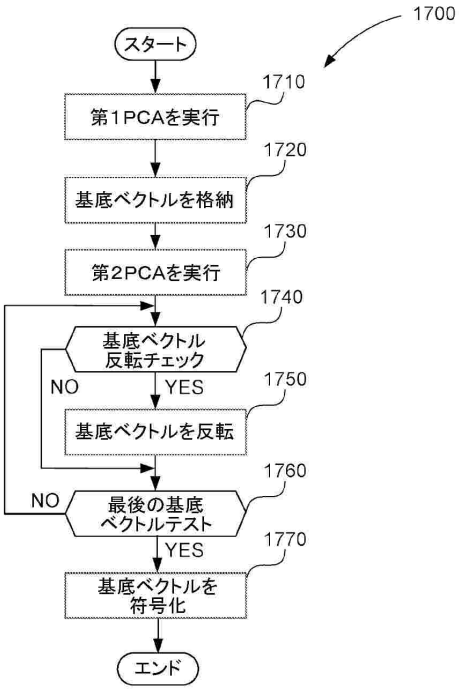


Fig. 17

【図１８Ａ】

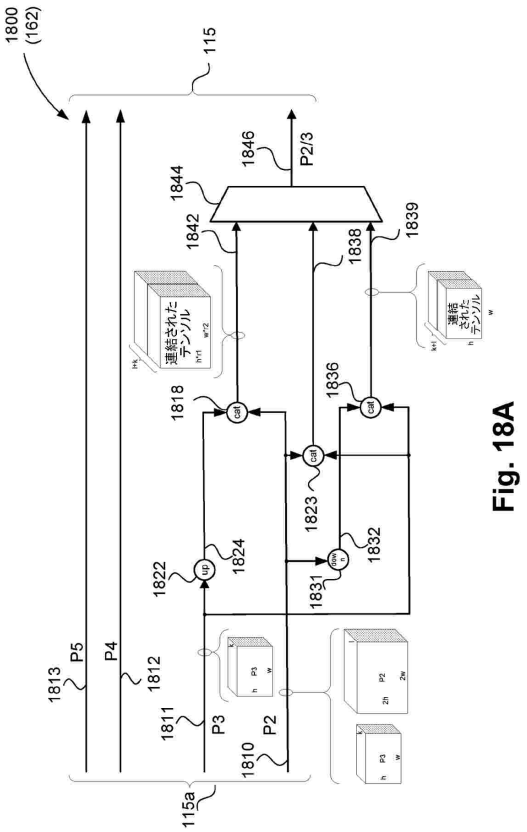


Fig. 18A

【図１８Ｂ】

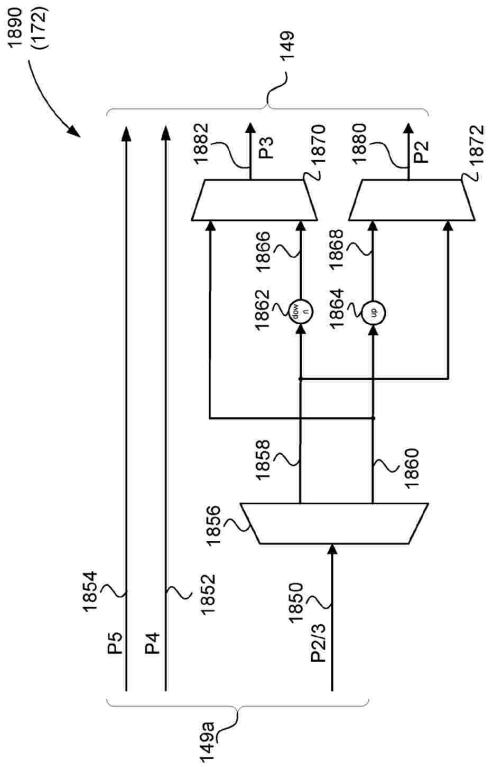


Fig. 18B

【図１９】

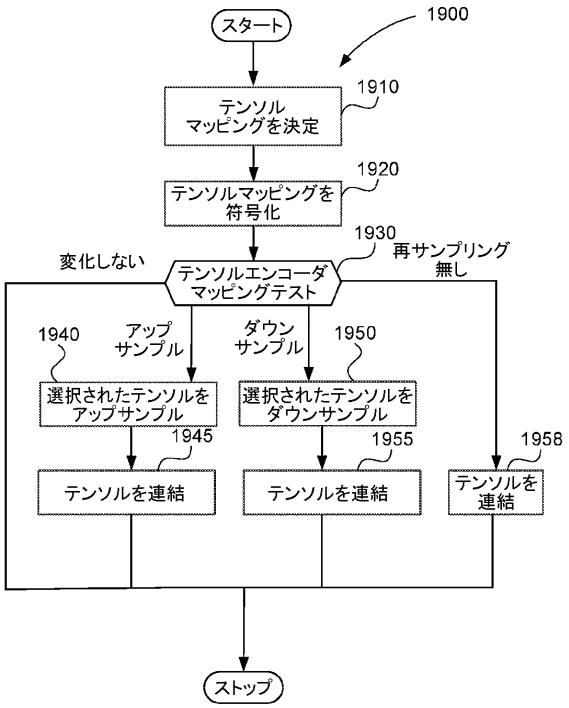


Fig. 19

【図２０】

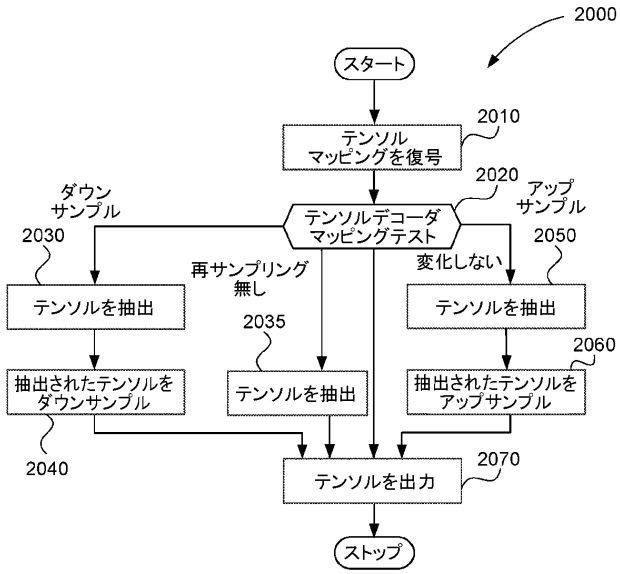


Fig. 20

【図２１】

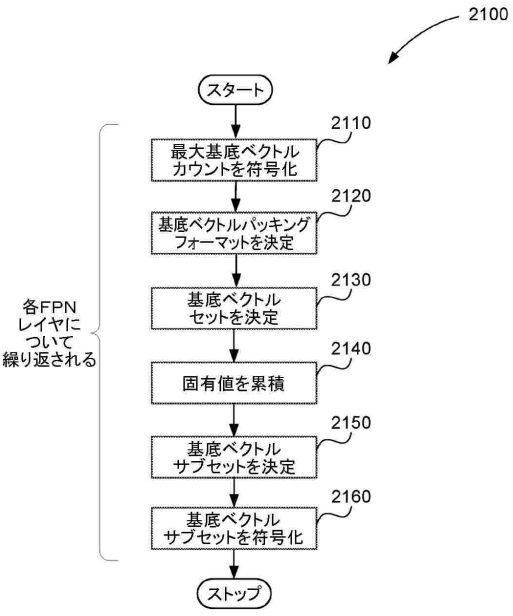


Fig. 21

【図２２】

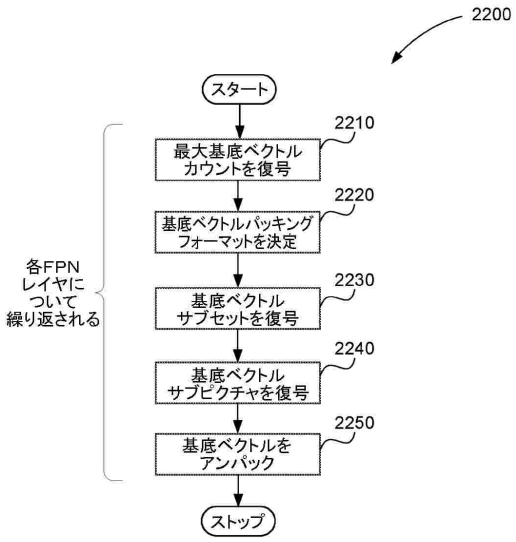


Fig. 22

【図２３】

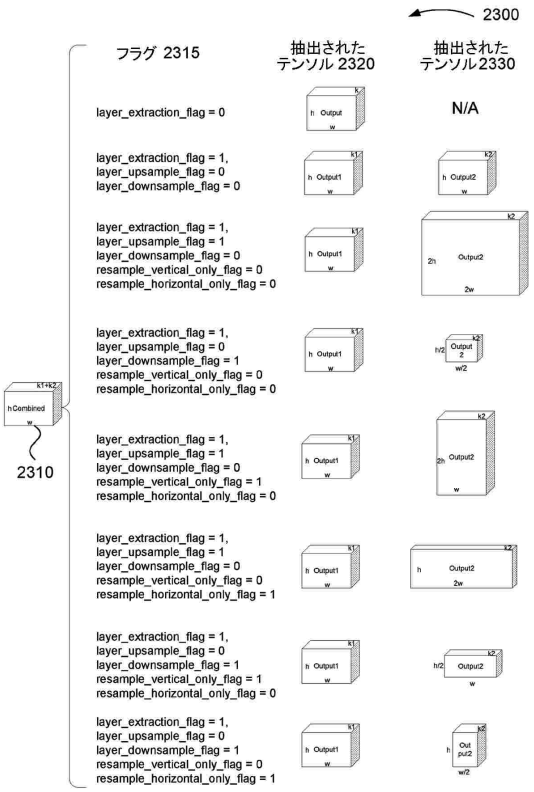


Fig. 23

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/AU2023/050289

A. CLASSIFICATION OF SUBJECT MATTER

H04N 19/169 (2014.01) H04N 19/103 (2014.01) H04N 19/159 (2014.01) H04N 19/172 (2014.01) H04N 19/18 (2014.01) H04N 19/50 (2014.01) H04N 19/91 (2014.01) H04N 19/96 (2014.01)

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
JVET documents (https://jvet-experts.org)

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

Databases: PATENW (EPODOC, WPIAP, TXTE), Google Patents, IPC/CPC – G06F17/16, G06N3/02, G06N3/0464, G06V10/77, H04N19, H04N19/174, H04N19/46, H04N19/85, H04N19/90 and keywords: average feature map, multiply, independently decodable, horizontally arranged, aspect ratio, VVC, HEVC, decode, encode, tensor, coefficient, decomposition, PCA, SVD, neural network, generating tensor, tensor decomposition, convolutional layer, encode layer tensor, subtract mean feature, decorrelate tensor channel, basis tensor channel, transform tensor coefficient, basis vector and similar terms.

Applicant/Inventor(s) name search was conducted in external databases DOCDB, DWPI and IP Australia internal databases.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	Documents are listed in the continuation of Box C	

[X] Further documents are listed in the continuation of Box C [X] See patent family annex

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"D" document cited by the applicant in the international application
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
21 June 2023	21 June 2023
Name and mailing address of the ISA/AU AUSTRALIAN PATENT OFFICE PO BOX 200, WODEN ACT 2606, AUSTRALIA Email address: pct@ipaustralia.gov.au	**Authorised officer** Jae-Hak Kim AUSTRALIAN PATENT OFFICE (ISO 9001 Quality Certified Service) Telephone No. +61 2 6283 2251

Form PCT/ISA/210 (fifth sheet) (July 2019)

INTERNATIONAL SEARCH REPORT C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		International application No. PCT/AU2023/050289
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JP 2015129785 A (NIPPON TELEGRAPH & TELEPHONE) 16 July 2015, English machine-generated translation from Espacenet Title, Abstract, Figs. 2 & 4A-5C, Paragraphs [0006], [0007], [0011], [0012], [0018], [0021], [0022] and [0033]	1, 2, 5, 11, 12, 15 and 21-24
X	WO 2021/254855 A1 (INTERDIGITAL VC HOLDINGS FRANCE, SAS) 23 December 2021 Title, Abstract, Fig. 4, Page 1 lines 10-17, Page 2 lines 8-20; Page 5 line 21 to Page 6 line 8, Page 6 lines 29-36, Page 8 lines 6-11 and Page 25 lines 14-26	1-24
A	US 10970629 B1 (AMAZON TECHNOLOGIES, INC.) 06 April 2021 See the whole content	
A	CN 110059796 B (HANGZHOU HIKVISION DIGITAL TECHNOLOGY CO LTD) 21 September 2021, English machine-generated translation from Espacenet See the whole content	

Form PCT/ISA/210 (fifth sheet) (July 2019)

INTERNATIONAL SEARCH REPORT Information on patent family members	International application No. PCT/AU2023/050289

This Annex lists known patent family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

Patent Document/s Cited in Search Report		Patent Family Member/s	
Publication Number	**Publication Date**	**Publication Number**	**Publication Date**
JP 2015129785 A	16 July 2015	JP 2015129785 A	16 Jul 2015
		JP 6139419 B2	31 May 2017
WO 2021/254855 A1	23 December 2021	WO 2021254855 A1	23 Dec 2021
		CN 116018757 A	25 Apr 2023
		EP 4168940 A1	26 Apr 2023
		IL 299171 A	01 Feb 2023
		KR 20230027152 A	27 Feb 2023
US 10970629 B1	06 April 2021	US 10970629 B1	06 Apr 2021
CN 110059796 B	21 September 2021	CN 110059796 A	26 Jul 2019
		CN 110059796 B	21 Sep 2021

End of Annex

Due to data integration issues this family listing may not include 10 digit Australian applications filed since May 2001.

フロントページの続き

,MC,ME,MK,MT,NL,NO,PL,PT,RO,RS,SE,SI,SK,SM,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GQ,GW,KM,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AO,AT,AU,AZ,BA,BB,BG,BH,BN,BR,BW,BY,BZ,CA,CH,CL,CN,CO,CR,CU,CV,CZ,DE,DJ,DK,DM,DO,DZ,EC,EE,EG,ES,FI,GB,GD,GE,GH,GM,GT,HN,HR,HU,ID,IL,IN,IQ,IR,IS,IT,JM,JO,JP,KE,KG,KH,KN,KP,KR,KW,KZ,LA,LC,LK,LR,LS,LU,LY,MA,MD,MG,MK,MN,MU,MW,MX,MY,MZ,NA,NG,NI,NO,NZ,OM,PA,PE,PG,PH,PL,PT,QA,RO,RS,RU,RW,SA,SC,SC,SD,SE,SG,SK,SL,ST,SV,SY,TH,TJ,TM,TN,TR,TT,TZ,UA,UG,US,UZ,VC,VN,WS,ZA,ZM,ZW
ノン オーストラリア プロプライエタリー リミテッド内

Fターム（参考） 5C159 MA04 MA05 MA21 MC11 ME01 PP04 TA01 UA02 UA05