(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2016-174384
(P2016-174384A)
(43)公開日　平成28年9月29日(2016.9.29)

(51)Int.Cl.		F I		テーマコード（参考）
H04N 19/70	(2014.01)	H04N	19/70	5C159
H04N 19/30	(2014.01)	H04N	19/30	

審査請求 有　請求項の数 8　OL　（全 41 頁）

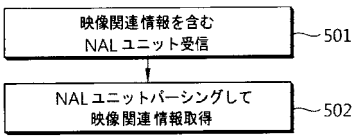
(21)出願番号	特願2016-92743 (P2016-92743)
(22)出願日	平成28年5月2日 (2016.5.2)
(62)分割の表示	特願2015-506892 (P2015-506892)の分割
原出願日	平成25年4月16日 (2013.4.16)
(31)優先権主張番号	10-2012-0038870
(32)優先日	平成24年4月16日 (2012.4.16)
(33)優先権主張国	韓国 (KR)
(31)優先権主張番号	10-2012-0066606
(32)優先日	平成24年6月21日 (2012.6.21)
(33)優先権主張国	韓国 (KR)
(31)優先権主張番号	10-2012-0067925
(32)優先日	平成24年6月25日 (2012.6.25)
(33)優先権主張国	韓国 (KR)

(71)出願人　596099882
エレクトロニクス　アンド　テレコミュニケーションズ　リサーチ　インスチチュート
ＥＬＥＣＴＲＯＮＩＣＳ　ＡＮＤ　ＴＥＬＥＣＯＭＭＵＮＩＣＡＴＩＯＮＳ　ＲＥＳＥＡＲＣＨ　ＩＮＳＴＩＴＵＴＥ
大韓民国　３０５－７００　デジョン　ユソン－グ　ガジョン－ロ　２１８
(74)代理人　100091982
弁理士　永井　浩之
(74)代理人　100091487
弁理士　中村　行孝
(74)代理人　100082991
弁理士　佐藤　泰和

最終頁に続く

(54)【発明の名称】映像情報デコーディング方法、映像デコーディング方法及びそれを利用する装置

(57)【要約】　（修正有）
【課題】映像情報デコーディング方法を提供する。
【解決手段】デコーディング方法は、エンコーディングされた映像に関連した情報を含むＮＡＬ(Ｎｅｔｗｏｒｋ Ａｂｓｔｒａｃｔｉｏｎ Ｌａｙｅｒ)ユニットを含むビットストリームを受信するステップと、前記ＮＡＬユニットのＮＡＬユニットヘッダをパーシングするステップとを含む。前記ＮＡＬユニットヘッダは、エンコーディング時、全体ビットストリームで非参照ピクチャであるか、または参照ピクチャであるかを示す１ビットのフラグ情報を含まない。
【選択図】図５



【特許請求の範囲】
【請求項１】
受信されたピクチャのスライスヘッダをパーシングするパーシング部;
前記受信されたピクチャをデコードするデコーディング部;
前記デコードされたピクチャを格納するDPB(decoded picture buffer);を含み
前記デコードされたピクチャの次のピクチャに対するスライスヘッダに含まれる参照ピクチャ情報に基づいて、前記デコードされたピクチャが参照ピクチャであるか又は非参照ピクチャであるかが表示 されて、
前記パーシング部は、エンコードされた映像に関連した情報を含むNAL(Network Abstraction Layer)ユニットのNALユニットヘッダをパーシングし、前記NALユニットヘッダは、映像のエンコーディング時に、前記NALユニットがビットストリームで非参照ピクチャであるか、または参照ピクチャであるかを示す情報を含まないことを特徴とする映像デコーディング装置。
【請求項２】
前記エンコードされた映像は、前記デコードされたピクチャの次のピクチャに対応する請求項1に記載のビデオデコード装置。
【請求項３】
前記パーシング部は、前記次のピクチャのスライスヘッダをパーシングする請求項1に記載のビデオデコード装置。
【請求項４】
前記パーシング部は、前記デコードされたピクチャが参照ピクチャであることを非参照ピクチャであるかどうかに基づいて、参照ピクチャリストを生成する請求項1に記載のビデオデコード装置。
【請求項５】
エンコードされた映像に関連した情報を含むNAL(Network Abstraction Layer)ユニットのNALユニットヘッダをパーシングする段階;
受信されたピクチャをデコードする段階;
前記デコードされたピクチャの次のピクチャに対するスライスヘッダに含まれる参照ピクチャ情報に基づいて、前記デコードされたピクチャが参照ピクチャであるか又は非参照ピクチャであるかが表示する段階
を含み、
前記NALユニットヘッダは、映像のエンコーディング時に、前記NALユニットがビットストリームで非参照ピクチャであるか、または参照ピクチャであるかを示す情報を含まないことを特徴とする映像デコーディング方法。
【請求項６】
前記エンコードされた映像は、前記デコードされたピクチャの次のピクチャに対応する請求項5に記載のビデオデコード装置。
【請求項７】
前記次のピクチャのスライスヘッダパーシングする段階
さらに含まれている、請求項5に記載のビデオデコード装置。
【請求項８】
前記パーシング部は、前記デコードされたピクチャが参照ピクチャであることを非参照ピクチャであるかどうかに基づいて、参照ピクチャリストを生成する段階
さらに含まれている、請求項5に記載のビデオデコード装置。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、ビデオ符号化及び復号化処理に関し、より詳しくは、ビットストリーム内のビデオの情報をデコーディングする方法及び装置に関する。
【背景技術】

【０００２】
最近、ＨＤ(Ｈｉｇｈ Ｄｅｆｉｎｉｔｉｏｎ)解像度を有する放送サービスが韓国内だけではなく、世界的に拡大されるにつれて、多くのユーザが高解像度、高画質の映像に慣れており、それによって、多くの機関が次世代映像機器に対する開発に拍車を掛けている。また、ＨＤＴＶと共にＨＤＴＶの４倍以上の解像度を有するＵＨＤ(Ｕｌｔｒａ Ｈｉｇｈ Ｄｅｆｉｎｉｔｉｏｎ)に対する関心が増大しながら、より高い解像度、高画質の映像に対する圧縮機術が要求されている。
【０００３】
映像圧縮のために、時間的に以前及び/または以後のピクチャから現在ピクチャに含まれている画素値を予測するインター(ｉｎｔｅｒ)予測技術、現在ピクチャ内の画素情報を利用して現在ピクチャに含まれている画素値を予測するイントラ(ｉｎｔｒａ)予測技術、出現頻度が高いシンボル(ｓｙｍｂｏｌ)に短い符号を割り当て、出現頻度が低いシンボルに長い符号を割り当てるエントロピー符号化技術などが使われることができる。
【０００４】
映像圧縮技術には、流動的なネットワーク環境を考慮せずにハードウェアの制限的な動作環境下で一定のネットワーク帯域幅を提供する技術がある。しかし、随時帯域幅が変化するネットワーク環境に適用される映像データを圧縮するためには新たな圧縮技術が要求され、そのために、スケーラブル(ｓｃａｌａｂｌｅ)ビデオ符号化/復号化方法が使われることができる。
【発明の概要】
【発明が解決しようとする課題】
【０００５】
本発明の技術的課題は、階層的ビットストリーム内の抽出及びスケーラビリティ情報を記述する方法及び装置を提供することである。
【０００６】
本発明の他の技術的課題は、柔軟な方式で多様な種類のビットストリームのスケーラビリティ情報を表現する方法及び装置を提供することである。
【０００７】
本発明の他の技術的課題は、階層的ビットストリーム内の抽出及びスケーラビリティ情報をパケットレベルでの適応的に変換可能に提供する方法及び装置を提供することである。
【課題を解決するための手段】
【０００８】
本発明の一実施例に係る映像情報デコーディング方法は、エンコーディングされた映像に関連した情報を含むＮＡＬ(Ｎｅｔｗｏｒｋ Ａｂｓｔｒａｃｔｉｏｎ Ｌａｙｅｒ)ユニットを含むビットストリームを受信するステップと、前記ＮＡＬユニットのＮＡＬユニットヘッダをパーシングするステップとを含み、前記ＮＡＬユニットヘッダは、前記ＮＡＬユニットのエンコーディング時、全体ビットストリームで非参照ピクチャであるか、または参照ピクチャであるかを示す１ビットのフラグ情報を含まない。
【０００９】
本発明の一実施例に係る映像デコーディング方法は、受信されたピクチャをデコーディングするステップと、デコーディングされたピクチャをＤＰＢ(ｄｅｃｏｄｅｄ ｐｉｃｔｕｒｅ ｂｕｆｆｅｒ)に参照ピクチャで表示するステップと、前記デコーディングされたピクチャの次のピクチャに対するスライスヘッダをパーシングするステップと、前記スライスヘッダに含まれている参照ピクチャ情報に基づき、前記デコーディングされたピクチャが参照ピクチャであるか、または非参照ピクチャであるかを表示するステップとを含む。
【００１０】
本発明の他の実施例として、活性化されるパラメータセットに対する情報を含むＳＥＩ(Ｓｕｐｐｌｅｍｅｎｔａｌ ｅｎｈａｎｃｅｍｅｎｔ ｉｎｆｏｒｍａｔｉｏｎ)メッセー

ジを受信し、前記パラメータセットに対する情報をパーシングする。
【発明の効果】
【００１１】
本発明の一実施例によると、階層的ビットストリーム内の抽出及びスケーラビリティ情報を記述する方法及び装置が提供される。
【００１２】
本発明の一実施例によると、柔軟な方式で多様な種類のビットストリームのスケーラビリティ情報を表現する方法及び装置が提供される。
【００１３】
本発明の他の実施例によると、階層的ビットストリーム内の抽出及びスケーラビリティ情報をパケットレベルでの適応的に変換可能に提供する方法及び装置が提供される。
【図面の簡単な説明】
【００１４】
【図１】映像符号化装置の一実施例に係る構成を示すブロック図である。
【図２】映像復号化装置の一実施例に係る構成を示すブロック図である。
【図３】本発明が適用されることができる、複数階層を利用したスケーラブルビデオコーディング構造の一実施例を概略的に示す概念図である。
【図４】本発明による映像情報のエンコーディング方法を説明するための制御流れ図である。
【図５】本発明による映像情報のデコーディング方法を説明するための制御流れ図である。
【発明を実施するための形態】
【００１５】
以下、図面を参照して本発明の実施形態に対して具体的に説明する。本明細書の実施例を説明するにあたって、関連した公知構成または機能に対する具体的な説明が本明細書の要旨を不明にすると判断される場合、その詳細な説明は省略する。
【００１６】
一構成要素が他の構成要素に“連結されている”または“接続されている”と言及された場合、該当他の構成要素に直接的に連結されている、または接続されていることもあるが、中間に他の構成要素が存在することもあると理解しなければならない。また、本発明において、特定構成を“含む”と記述する内容は、該当構成以外の構成を排除するものではなく、追加的な構成が本発明の実施または本発明の技術的思想の範囲に含まれることができることを意味する。
【００１７】
第１、第２などの用語は、多様な構成要素の説明に使われることができるが、前記構成要素は、前記用語により限定されてはならない。前記用語は、一つの構成要素を他の構成要素から区別する目的にのみ使われる。例えば、本発明の権利範囲を外れない限り、第１の構成要素は第２の構成要素と命名することができ、同様に、第２の構成要素も第１の構成要素と命名することができる。
【００１８】
また、本発明の実施例に示す構成部は、互いに異なる特徴的な機能を示すために独立的に図示されるものであり、各構成部が分離されたハードウェアや一つのソフトウェア構成単位に構成されることを意味しない。即ち、各構成部は、説明の便宜上、各々の構成部として羅列して含むものであり、各構成部のうち少なくとも２個の構成部が統合されて一つの構成部からなり、または一つの構成部が複数個の構成部に分けられて機能を遂行することができ、このような各構成部の統合された実施例及び分離された実施例も本発明の本質から外れない限り、本発明の権利範囲に含まれる。
【００１９】
また、一部の構成要素は、本発明で本質的な機能を遂行する必須な構成要素ではなく、単に性能を向上させるための選択的構成要素である。本発明は、単に性能向上のために使

われる構成要素を除いた本発明の本質を具現するのに必須な構成部のみを含んで具現されることができ、単に性能向上のために使われる選択的構成要素を除いた必須構成要素のみを含む構造も本発明の権利範囲に含まれる。

【００２０】

図１は、映像符号化装置の一実施例に係る構成を示すブロック図である。スケーラブル(ｓｃａｌａｂｌｅ)ビデオ符号化/復号化方法または装置は、スケーラビリティ(ｓｃａｌａｂｉｌｉｔｙ)を提供しない一般的な映像符号化/復号化方法または装置の拡張(ｅｘｔｅｎｓｉｏｎ)により具現されることができ、図１のブロック図は、スケーラブルビデオ符号化装置のベースになることができる映像符号化装置の一実施例を示す。

【００２１】

図１を参照すると、前記映像符号化装置１００は、動き予測部１１１、動き補償部１１２、イントラ予測部１２０、スイッチ１１５、減算器１２５、変換部１３０、量子化部１４０、エントロピー符号化部１５０、逆量子化部１６０、逆変換部１７０、加算器１７５、フィルタ部１８０及び参照映像バッファ１９０を含む。

【００２２】

映像符号化装置１００は、入力映像に対してイントラ(ｉｎｔｒａ)モードまたはインター(ｉｎｔｅｒ)モードに符号化を実行することで、ビットストリーム(ｂｉｔ ｓｔｒｅａｍ)を出力することができる。イントラ予測は画面内予測を意味し、インター予測は画面間予測を意味する。イントラモードである場合、スイッチ１１５がイントラに切り替えられ、インターモードである場合、スイッチ１１５がインターに切り替えられる。映像符号化装置１００は、入力映像の入力ブロックに対する予測ブロックを生成した後、入力ブロックと予測ブロックとの差分を符号化することができる。

【００２３】

イントラモードである場合、イントラ予測部１２０は、現在ブロック周辺の既に符号化されたブロックの画素値を利用して空間的予測を実行することで、予測ブロックを生成することができる。

【００２４】

インターモードである場合、動き予測部１１１は、動き予測過程で参照映像バッファ１９０に格納されている参照映像で入力ブロックと最もよくマッチされる領域を探して動きベクトルを求めることができる。動き補償部１１２は、動きベクトルと参照映像バッファ１９０に格納されている参照映像を利用して動き補償を実行することによって、予測ブロックを生成することができる。

【００２５】

減算器１２５は、入力ブロックと生成された予測ブロックとの差分により残余ブロック(ｒｅｓｉｄｕａｌ ｂｌｏｃｋ)を生成することができる。変換部１３０は、残余ブロックに対して変換(ｔｒａｎｓｆｏｒｍ)を実行することで、変換係数(ｔｒａｎｓｆｏｒｍ ｃｏｅｆｆｉｃｉｅｎｔ)を出力することができる。そして、量子化部１４０は、入力された変換係数を量子化パラメータによって量子化し、量子化された係数(ｑｕａｎｔｉｚｅｄ ｃｏｅｆｆｉｃｉｅｎｔ)を出力することができる。

【００２６】

エントロピー符号化部１５０は、量子化部１４０で算出された値または符号化過程で算出された符号化パラメータ値などに基づき、シンボル(ｓｙｍｂｏｌ)を確率分布によってエントロピー符号化することで、ビットストリーム(ｂｉｔ ｓｔｒｅａｍ)を出力することができる。エントロピー符号化方法は、多様な値を有するシンボルの入力を受け、統計的な重複性を除去すると共に、復号可能な２進数の列で表現する方法である。

【００２７】

ここで、シンボルとは、符号化/復号化対象構文要素(ｓｙｎｔａｘ ｅｌｅｍｅｎｔ)及び符号化パラメータ(ｃｏｄｉｎｇ ｐａｒａｍｅｔｅｒ)、残余信号(ｒｅｓｉｄｕａｌ ｓｉｇｎａｌ)の値などを意味する。符号化パラメータは、符号化及び復号化に必要な媒介変数であって、構文要素のように符号化器で符号化されて復号化器に伝達される情報だ

けでなく、符号化または復号化過程で類推されることができる情報を含むことができ、映像を符号化または復号化する時に必要な情報を意味する。符号化パラメータは、例えば、イントラ/インター予測モード、移動/動きベクトル、参照映像索引、符号化ブロックパターン、残余信号有無、変換係数、量子化された変換係数、量子化パラメータ、ブロックの大きさ、ブロック分割情報などの値または統計を含むことができる。また、残余信号は、原信号と予測信号との差を意味し、また、原信号と予測信号との差が変換(ｔｒａｎｓｆｏｒｍ)された形態の信号または原信号と予測信号との差が変換されて量子化された形態の信号を意味する。残余信号は、ブロック単位では残余ブロックという。

【００２８】

エントロピー符号化が適用される場合、高い発生確率を有するシンボルに少ない数のビットが割り当てられ、低い発生確率を有するシンボルに多い数のビットが割り当てられてシンボルが表現されることによって、符号化対象シンボルに対するビット列の大きさが減少されることができる。したがって、エントロピー符号化を介して映像符号化の圧縮性能が高まることができる。

【００２９】

エントロピー符号化のために、指数ゴロム(ｅｘｐｏｎｅｎｔｉａｌ ｇｏｌｏｍｂ)、ＣＡＶＬＣ(Ｃｏｎｔｅｘｔ－Ａｄａｐｔｉｖｅ Ｖａｒｉａｂｌｅ Ｌｅｎｇｔｈ Ｃｏｄｉｎｇ)、ＣＡＢＡＣ(Ｃｏｎｔｅｘｔ－Ａｄａｐｔｉｖｅ Ｂｉｎａｒｙ Ａｒｉｔｈｍｅｔｉｃ Ｃｏｄｉｎｇ)のような符号化方法が使われることができる。例えば、エントロピー符号化部１５０には、可変長さ符号化(ＶＬＣ：Ｖａｒｉａｂｌｅ Ｌｅｎｇｈｔｈ Ｃｏｄｉｎｇ/Ｃｏｄｅ)テーブルのようなエントロピー符号化を実行するためのテーブルが格納されることができ、エントロピー符号化部１５０は、格納された可変長さ符号化(ＶＬＣ)テーブルを使用してエントロピー符号化を実行することができる。また、エントロピー符号化部１５０は、対象シンボルの２進化(ｂｉｎａｒｉｚａｔｉｏｎ)方法及び対象シンボル/ビン(ｂｉｎ)の確率モデル(ｐｒｏｂａｂｉｌｉｔｙ ｍｏｄｅｌ)を導出した後、導出された２進化方法または確率モデルを使用してエントロピー符号化を実行することもできる。

【００３０】

量子化された係数は、逆量子化部１６０で逆量子化され、逆変換部１７０で逆変換されることができる。逆量子化、逆変換された係数は、加算器１７５を介して予測ブロックと加えられて復元ブロックが生成されることができる。

【００３１】

復元ブロックは、フィルタ部１８０を経て、フィルタ部１８０は、デブロッキングフィルタ(ｄｅｂｌｏｃｋｉｎｇ ｆｉｌｔｅｒ)、ＳＡＯ(Ｓａｍｐｌｅ Ａｄａｐｔｉｖｅ Ｏｆｆｓｅｔ)、ＡＬＦ(Ａｄａｐｔｉｖｅ Ｌｏｏｐ Ｆｉｌｔｅｒ)のうち少なくとも一つ以上を復元ブロックまたは復元ピクチャに適用することができる。フィルタ部１８０を経た復元ブロックは、参照映像バッファ１９０に格納されることができる。

【００３２】

図２は、映像復号化装置の一実施例に係る構成を示すブロック図である。図１で詳述したように、スケーラブルビデオ符号化/復号化方法または装置は、スケーラビリティを提供しない一般的な映像符号化/復号化方法または装置の拡張により具現されることができ、図２のブロック図は、スケーラブルビデオ復号化装置のベースになることができる映像復号化装置の一実施例を示す。

【００３３】

図２を参照すると、前記映像復号化装置２００は、エントロピー復号化部２１０、逆量子化部２２０、逆変換部２３０、イントラ予測部２４０、動き補償部２５０、フィルタ部２６０及び参照映像バッファ２７０を含む。

【００３４】

映像復号化装置２００は、符号化器から出力されたビットストリームの入力を受けてイントラモードまたはインターモードに復号化を実行することで、再構成された映像、即ち

、復元映像を出力することができる。イントラモードである場合、スイッチがイントラに切り替えられ、インターモードである場合、スイッチがインターに切り替えられる。映像復号化装置２００は、入力されたビットストリームから復元された残余ブロック(ｒｅｓｉｄｕａｌ ｂｌｏｃｋ)を得て、予測ブロックを生成した後、復元された残余ブロックと予測ブロックを加えて再構成されたブロック、即ち、復元ブロックを生成することができる。

【００３５】

エントロピー復号化部２１０は、入力されたビットストリームを確率分布によってエントロピー復号化し、量子化された係数(ｑｕａｎｔｉｚｅｄ ｃｏｅｆｆｉｃｉｅｎｔ)形態のシンボルを含むシンボルを生成することができる。エントロピー復号化方法は、２進数の列の入力を受けて各シンボルを生成する方法である。エントロピー復号化方法は、前述したエントロピー符号化方法と同様である。

【００３６】

量子化された係数は、逆量子化部２２０で逆量子化され、逆変換部２３０で逆変換され、量子化された係数が逆量子化/逆変換された結果、復元された残余ブロック(ｒｅｓｉｄｕａｌ ｂｌｏｃｋ)が生成されることができる。

【００３７】

イントラモードである場合、イントラ予測部２４０は、現在ブロック周辺の既に符号化されたブロックの画素値を利用して空間的予測を実行することで、予測ブロックを生成することができる。インターモードである場合、動き補償部２５０は、動きベクトル及び参照映像バッファ２７０に格納されている参照映像を利用して動き補償を実行することによって、予測ブロックを生成することができる。

【００３８】

復元された残余ブロックと予測ブロックは、加算器２５５を介して加えられ、加えられたブロックは、フィルタ部２６０を経る。フィルタ部２６０は、デブロッキングフィルタ、ＳＡＯ、ＡＬＦのうち少なくとも一つ以上を復元ブロックまたは復元ピクチャに適用することができる。フィルタ部２６０は、再構成された映像、即ち、復元映像を出力する。復元映像は、参照映像バッファ２７０に格納されて画面間予測に使われることができる。

【００３９】

前記映像復号化装置２００に含まれているエントロピー復号化部２１０、逆量子化部２２０、逆変換部２３０、イントラ予測部２４０、動き補償部２５０、フィルタ部２６０及び参照映像バッファ２７０のうち、映像の復号化に直接的に関連した構成要素、例えば、エントロピー復号化部２１０、逆量子化部２２０、逆変換部２３０、イントラ予測部２４０、動き補償部２５０、フィルタ部２６０などを他の構成要素と区分して復号化部またはデコーディング部で表現することができる。

【００４０】

また、映像復号化装置２００は、ビットストリームに含まれているエンコーディングされた映像に関連した情報をパーシングするパーシング部(図示せず)をさらに含むことができる。パーシング部は、エントロピー復号化部２１０を含むこともでき、エントロピー復号化部２１０に含まれることもできる。また、このようなパーシング部は、デコーディング部の一つの構成要素で具現されることもできる。

【００４１】

図３は、本発明が適用されることができる、複数階層を利用したスケーラブルビデオコーディング構造の一実施例を概略的に示す概念図である。図３において、ＧＯＰ(Ｇｒｏｕｐ ｏｆ Ｐｉｃｔｕｒｅ)は、ピクチャ群、即ち、ピクチャのグループを示す。

【００４２】

映像データを送信するためには送信媒体が必要であり、その性能は、多様なネットワーク環境によって送信媒体別に異なる。このような多様な送信媒体またはネットワーク環境への適用のためにスケーラブルビデオコーディング方法が提供されることができる。

【００４３】

スケーラブルビデオコーディング方法は、階層(ｌａｙｅｒ)間のテクスチャ情報、動き情報、残余信号などを活用して階層間重複性を除去して符号化/復号化性能を高めるコーディング方法である。スケーラブルビデオコーディング方法は、送信ビット率、送信エラー率、システムリソースなどの周辺条件によって、空間的、時間的、画質的な観点で多様なスケーラビリティを提供することができる。

【００４４】

スケーラブルビデオコーディングは、多様なネットワーク状況に適用可能なビットストリームを提供することができるように、複数階層(ｍｕｌｔｉｐｌｅ ｌａｙｅｒｓ)構造を使用して実行されることができる。例えば、スケーラブルビデオコーディング構造は、一般的な映像符号化方法を利用して映像データを圧縮処理する基本階層を含むことができ、基本階層の符号化情報及び一般的な映像符号化方法を共に使用して映像データを圧縮処理する向上階層を含むことができる。

【００４５】

ここで、階層(ｌａｙｅｒ)は、空間(ｓｐａｔｉａｌ、例えば、映像の大きさ)、時間(ｔｅｍｐｏｒａｌ、例えば、符号化順序、映像出力順序、フレームレート)、画質、複雑度などを基準にした区分される映像及びビットストリーム(ｂｉｔ ｓｔｒｅａｍ)の集合を意味する。また、基本階層は、参照階層またはＢａｓｅ ｌａｙｅｒを意味し、向上階層は、Ｅｎｈａｎｃｅｍｅｎｔ ｌａｙｅｒを意味する。また、複数の階層は、相互間に従属性を有することもできる。

【００４６】

図３を参照すると、例えば、基本階層は、ＳＤ(ｓｔａｎｄａｒｄ ｄｅｆｉｎｉｔｉｏｎ)、１５Ｈｚのフレーム率、１Ｍｂｐｓビット率で定義されることができ、第１の向上階層は、ＨＤ(ｈｉｇｈ ｄｅｆｉｎｉｔｉｏｎ)、３０Ｈｚのフレーム率、３.９Ｍｂｐｓビット率で定義されることができ、第２の向上階層は、４Ｋ－ＵＨＥ(ｕｌｔｒａ ｈｉｇｈ ｄｅｆｉｎｉｔｉｏｎ)、６０Ｈｚのフレーム率、２７.２Ｍｂｐｓビット率で定義されることができる。前記フォーマット(ｆｏｒｍａｔ)、フレーム率、ビット率等は、一つの実施例に過ぎず、必要によって異なるように決まることができる。また、使われる階層の数も本実施例に限定されるものではなく、状況によって異なるように決まることができる。

【００４７】

例えば、送信帯域幅が４Ｍｂｐｓの場合、前記第１の向上階層ＨＤのフレームレートを減らして１５Ｈｚ以下に送信することができる。スケーラブルビデオコーディング方法は、前記図３の実施例で詳述した方法により時間的、空間的、画質的スケーラビリティを提供することができる。

【００４８】

以下、スケーラブルビデオコーディングは、符号化観点において、スケーラブルビデオ符号化と同じ意味を有し、復号化観点において、スケーラブルビデオ復号化と同じ意味を有する。

【００４９】

前述したように、異種の通信網及び多様な端末によって、スケーラビリティは、現在ビデオフォーマットの重要な機能になった。ＡＶＣ(Ａｄｖａｎｃｅｄ Ｖｉｄｅｏ Ｃｏｄｉｎｇ)の拡張標準であるＳＶＣ(Ｓｃａｌａｂｌｅ Ｖｉｄｅｏ Ｃｏｄｉｎｇ)は、圧縮効率を最大限維持すると共に、多様な範囲のビットレートを有するビットストリームを生成することができるように開発された。多様なデバイス及びネットワークの特性と変化を満たすために、ＳＶＣビットストリームは、容易に多様な方式で抽出されることができる。即ち、ＳＶＣ標準は、空間的(ｓｐａｔｉａｌ)、時間的(ｔｅｍｐｏｒａｌ)、画質(ＳＮＲ)スケーラビリティを提供する。

【００５０】

一方、複数の階層を含むビットストリームは、パケットスイッチングネットワーク(ｐａｃｋｅｔ－ｓｗｉｔｃｈｉｎｇ ｎｅｔｗｏｒｋ)を介してビデオの適応的送信を容易に

するＮＡＬ(Ｎｅｔｗｏｒｋ Ａｂｓｔｒａｃｔｉｏｎ Ｌａｙｅｒ)ｕｎｉｔで構成される。複数の階層と同様に、ビットストリーム内の複数の多視点映像を含むマルチビュービデオコーディング(ｍｕｌｔｉ－ｖｉｅｗ ｖｉｄｅｏ ｃｏｄｉｎｇ)における複数の視点間の関係は、複数のレイヤをサポートするビデオでの空間的階層(ｓｐａｔｉａｌ ｌａｙｅｒ)間の関係と同様である。

【００５１】

コンテンツ伝達経路(ｃｏｎｔｅｎｔ ｄｅｌｉｖｅｒｙ ｐａｔｈ)における全てのノードでビットストリームを効果的且つ効率的に変換するためにはビットストリームのスケーラビリティ情報が相当重要である。現在単一レイヤに対するビデオコーディングの標準(ｈｉｇｈ ｅｆｆｉｃｉｅｎｃｙ ｖｉｄｅｏ ｃｏｄｉｎｇ)では、ＮＡＬ Ｕｎｉｔ(以下、ＮＡＬＵ)ヘッダ(ｈｅａｄｅｒ)に階層情報と関連した２個のフィールドであるｔｅｍｐｏｒａｌ_ｉｄとｒｅｓｅｒｖｅｄ_ｏｎｅ_５ｂｉｔｓが存在する。３ｂｉｔｓの長さを有するｔｅｍｐｏｒａｌ_ｉｄは、ビデオビットストリームの時間的レイヤ(ｔｅｍｐｏｒａｌ ｌａｙｅｒ)を示し、ｒｅｓｅｒｖｅｄ_ｏｎｅ_５ｂｉｔｓは、以後に他の階層情報を示すための領域に該当する。

【００５２】

時間的レイヤは、ビデオコーディングレイヤ(ｖｉｄｅｏ ｃｏｄｉｎｇ ｌａｙｅｒ、ＶＣＬ)ＮＡＬ ｕｎｉｔで構成された時間的にスケールされることができるビットストリームのレイヤを意味し、時間的レイヤは、特定のｔｅｍｐｏｒａｌ_ｉｄ値を有する。

【００５３】

本発明は、複数のレイヤをサポートするビットストリーム内で映像の抽出情報(ｅｘｔｒａｃｔｉｏｎ)及び階層的情報(ｓｃａｌａｂｉｌｉｔｙ ｉｎｆｏｒｍａｔｉｏｎ)を効果的に記述し、それをシグナリングするための方法及びそれを具現する装置に関する発明である。

【００５４】

本発明では、ビットストリームを２つのタイプ、時間スケーラビリティ(ｔｅｍｐｏｒａｌ ｓｃａｌａｂｉｌｉｔｙ)のみをサポートする基本タイプ(ｂａｓｅ ｔｙｐｅ)と、時間を含む空間/画質/視点をサポートするスケーラビリティを有することができる拡張タイプ(ｅｘｔｅｎｄｅｄ ｔｙｐｅ)とに分けて説明する。

【００５５】

ビットストリームの第１のタイプは、単一階層ビデオをサポートするビットストリームに対するものであり、第２のタイプは、ＨＥＶＣベースの階層的ビデオ符号化で向上階層のためのものである。以下、２つのビットストリームタイプのスケーラビリティ情報を表現するための改善方案を提案する。本発明によると、拡張タイプにおいて、５ｂｉｔのｒｅｓｅｒｖｅｄ_ｏｎｅ_５ｂｉｔｓは、スケーラブル階層の識別子を示すｌａｙｅｒ_ｉｄとして使われることができる。

【００５６】

ＮＡＬＵヘッダからｎａｌ_ｒｅｆ_ｆｌａｇ除去

【００５７】

ｎａｌ_ｒｅｆ_ｆｌａｇは、非参照ピクチャ(ｎｏｎ－ｒｅｆｅｒｅｎｃｅ ｐｉｃｔｕｒｅ)を示すために使われる。この情報は、非参照ピクチャと参照ピクチャ(ｒｅｆｅｒｅｎｃｅ ｐｉｃｔｕｒｅ)との間の概略的な優先順位を示すが、送信のためのｎａｌ_ｒｅｆ_ｆｌａｇの使用は多少制限的である。

【００５８】

参照ピクチャ(Ｒｅｆｅｒｅｎｃｅ ｐｉｃｔｕｒｅ)は、デコーディング順序上、後続のピクチャ(ｓｕｂｓｅｑｕｅｎｔ ｐｉｃｔｕｒｅｓ)のデコーディング時、画面間予測のために使われることができるサンプルを含むピクチャを意味する。

【００５９】

非参照ピクチャ(ｎｏｎ－ｒｅｆｅｒｅｎｃｅ ｐｉｃｔｕｒｅ)は、デコーディング順序上、後続のピクチャのデコーディング時、画面間予測のために使われないサンプルを含

むピクチャを意味する。
【００６０】
ｎａｌ_ｒｅｆ_ｆｌａｇは、エンコーディング時、該当ｎａｌユニットが全体ビットストリーム上で非参照ピクチャであるか、または参照ピクチャであるかを示す情報を示すフラグである。
【００６１】
ｎａｌ_ｒｅｆ_ｆｌａｇが１の場合、ＮＡＬＵは、ＳＰＳ(ｓｅｑｕｅｎｃｅ ｐａｒａｍｅｔｅｒ ｓｅｔ)、ＰＰＳ(ｐｉｃｔｕｒｅ ｐａｒａｍｅｔｅｒ ｓｅｔ)、ＡＰＳ(ａｄａｐｔａｔｉｏｎ ｐａｒａｍｅｔｅｒ ｓｅｔ)または参照ピクチャのスライスを含むことを意味し、ｎａｌ_ｒｅｆ_ｆｌａｇが０の場合、ＮＡＬＵは、非参照ピクチャの一部または全部を含むスライスを含むことを意味する。
【００６２】
このとき、ｎａｌ_ｒｅｆ_ｆｌａｇ値が１であるＮＡＬＵは、参照ピクチャのスライスを含むことができ、ｎａｌ_ｒｅｆ_ｆｌａｇは、ＶＰＳ(ｖｉｄｅｏ ｐａｒａｍｅｔｅｒ ｓｅｔ)、ＳＰＳ(ｓｅｑｕｅｎｃｅ ｐａｒａｍｅｔｅｒ ｓｅｔ)、ＰＰＳ(ｐｉｃｔｕｒｅ ｐａｒａｍｅｔｅｒ ｓｅｔ)のＮＡＬＵに対して１の値を有する。特定ピクチャのＶＣＬ ＮＡＬＵのうち一つが、ｎａｌ_ｒｅｆ_ｆｌａｇ値が０の場合、該当ピクチャの全てのＶＣＬ ＮＡＬＵに対してｎａｌ_ｒｅｆ_ｆｌａｇは、０の値を有する。
【００６３】
一方、全ての非参照ピクチャ(ｎｏｎ－ｒｅｆｅｒｅｎｃｅ ｐｉｃｔｕｒｅ)、特に、大部分最上位時間的レイヤに該当する非参照ピクチャが抽出されると、抽出後に残った全てのピクチャのｎａｌ_ｒｅｆ_ｆｌａｇは１になる。
【００６４】
しかし、適応変換された(抽出された)ビットストリームの一部ピクチャ、即ち、残ったビットストリームで最上位時間的レイヤに該当するピクチャは、ｎａｌ_ｒｅｆ_ｆｌａｇが１であるとしても非参照ピクチャになる。
【００６５】
即ち、ＮＡＬＵヘッダの他のシンタックス要素(例えば、ｔｅｍｐｏｒａｌ_ｉｄ)が適応変換(抽出)をサポートするのに一層効果的である。即ち、ビットストリームが含む全体時間的レイヤの個数とＮＡＬＵヘッダのｔｅｍｐｏｒａｌ_ｉｄ値を利用し、所望の時間的レイヤを含むビットストリームを抽出することができる。
【００６６】
また、ｎａｌ_ｒｅｆ_ｆｌａｇは、ｎａｌ_ｒｅｆ_ｆｌａｇを含むＮＡＬＵで構成されたピクチャをデコーディング(復元)した後、ＤＰＢ(ｄｅｃｏｄｅｄ ｐｉｃｔｕｒｅ ｂｕｆｆｅｒ)のようなメモリに格納する時、該当ピクチャを以後に参照ピクチャとして使用するかどうかを表示する時も使われることができる。ｎａｌ_ｒｅｆ_ｆｌａｇが１の場合、以後に参照ピクチャとして使われると表示し、ｎａｌ_ｒｅｆ_ｆｌａｇが０の場合、以後に参照ピクチャとして使用しないと表示することができる。
【００６７】
ｎａｌ_ｒｅｆ_ｆｌａｇを介して該当ＮＡＬＵが非参照ピクチャであるか、または参照ピクチャであるかを判断せずに、デコーディングされたピクチャをＤＰＢに格納する時、参照ピクチャで表示することができる。この場合、デコーディングされたピクチャが非参照ピクチャであるが、参照ピクチャで表示されても、デコーディング順序上、該当ピクチャの次のピクチャをデコーディングするにあたって、次のピクチャのヘッダ(ｓｌｉｃｅ ｈｅａｄｅｒ)に伝達される参照ピクチャリストに該当ピクチャが含まれていないため、問題が発生しない。
【００６８】
即ち、次のピクチャをデコーディングする時、スライスヘッダに含まれている参照ピクチャリストによって以前にデコーディングされたピクチャが参照ピクチャであるか、または非参照ピクチャであるかが表示される。したがって、ｎａｌ_ｒｅｆ_ｆｌａｇを介して

デコーディングされたピクチャが参照ピクチャであるかどうかが判断されずに、参照ピクチャで表示されても、デコーディングされたピクチャを参照ピクチャまたは非参照ピクチャと判断するのに問題が発生しない。

【００６９】

本発明では、ＮＡＬＵヘッダからｎａｌ_ｒｅｆ_ｆｌａｇを削除し、またはｎａｌ_ｒｅｆ_ｆｌａｇの意味(ｓｅｍａｎｔｉｃｓ)を変更することを提案する。ｎａｌ_ｒｅｆ_ｆｌａｇ削除と関連した実施例は、下記の通りである。

【００７０】

実施例１

【００７１】

ｎａｌ_ｒｅｆ_ｆｌａｇをｓｌｉｃｅ_ｒｅｆ_ｆｌａｇに変更し、フラグの位置をＮＡＬＵヘッダからスライスヘッダ(ｓｌｉｃｅ ｈｅａｄｅｒ)に移す。スライスヘッダのシンタックスは、表１のように修正されることができる。

【００７２】

【表１】

slice_header( ) {	**Descriptor**
slice_ref_flag	u(1)
first_slice_in_pic_flag	u(1)
if(first_slice_in_pic_flag = = 0)	
slice_address	u(v)
...	ue(v)
}	

【００７３】

表１において、ｓｌｉｃｅ_ｒｅｆ_ｆｌａｇの値が１の場合、スライスが参照ピクチャの一部であることを指示し、０の場合、スライスが非参照ピクチャの一部であることを指示する。

【００７４】

実施例２

【００７５】

ｎａｌ_ｒｅｆ_ｆｌａｇをａｕ_ｒｅｆ_ｆｌａｇに変更し、フラグの位置をＮＡＬＵヘッダからアクセスユニットデリミタ(ａｃｃｅｓｓ ｕｎｉｔ ｄｅｌｉｍｉｔｅｒ)に移す。アクセスユニットデリミタのシンタックスは、表２の通りである。

【００７６】

【表２】

access_unit_delimiter_rbsp( ) {	**Descriptor**
pic_type	u(3)
au_ref_flag	u(1)
rbsp_trailing_bits( )	
}	

【００７７】

表２において、ａｕ_ｒｅｆ_ｆｌａｇが１の場合、アクセスユニットが参照ピクチャを含むことを指示し、０の場合、アクセスユニットが非参照ピクチャを含むことを指示する。

【００７８】

実施例３

【００７９】

ｎａｌ_ｒｅｆ_ｆｌａｇを他のシンタックスに移動させずに、ｎａｌ_ｒｅｆ_ｆｌａｇをＮＡＬＵヘッダから削除する。

【００８０】

エンコーディング時、全体ビットストリームで非参照ピクチャであるか、または参照ピクチャであるかを示す１ビットのフラグ情報であるｎａｌ_ｒｅｆ_ｆｌａｇが削除されると、ｎａｌ_ｒｅｆ_ｆｌａｇによって実行されたピクチャが参照ピクチャであるかどうかに対する判断は、他の過程を介して実行されることができる。受信されたピクチャをデコーディングした後、デコーディングされたピクチャをＤＰＢ(ｄｅｃｏｄｅｄ ｐｉｃｔｕｒｅ ｂｕｆｆｅｒ)に無条件参照ピクチャで表示する。即ち、デコーディングされたピクチャが参照ピクチャであるかどうかを判断せずに、参照ピクチャで表示されることができる。

【００８１】

その後、デコーディングされたピクチャの次のピクチャに対するスライスヘッダをパーシングし、スライスヘッダに含まれている参照ピクチャ情報に基づいてデコーディングされたピクチャが参照ピクチャであるか、または非参照ピクチャであるかを表示することができる。

【００８２】

実施例４

【００８３】

ＮＡＬＵヘッダからｎａｌ_ｒｅｆ_ｆｌａｇを削除し、非参照ピクチャのＮＡＬＵという情報を示すためにｔｅｍｐｏｒａｌ_ｉｄを使用することができる。ｔｅｍｐｏｒａｌ_ｉｄは“７”、またはビットストリームに含まれている最大時間的レイヤの個数－１(即ち、ｍａｘ_ｔｅｍｐｏｒａｌ_ｌａｙｅｒｓ_ｍｉｎｕｓ１)、または“０”を除いた既設定された値になることができる。

【００８４】

実施例５

【００８５】

ＮＡＬＵヘッダからｎａｌ_ｒｅｆ_ｆｌａｇを削除し、非参照ピクチャのＮＡＬＵという情報を示すためにｒｅｓｅｒｖｅｄ_ｏｎｅ_５ｂｉｔｓをｐｒｉｏｒｉｔｙ_ｉｄ構成要素として使用することができる。ｐｒｉｏｒｉｔｙ_ｉｄは、該当ＮＡＬＵの優先順位(ｐｒｉｏｒｉｔｙ)を示す識別子であって、異なる空間、時間及び画質に関係なく優先順位によるビットストリーム抽出機能を提供するために使われる。

【００８６】

即ち、もし、ｔｅｍｐｏｒａｌ_ｉｄ＝Ｔａが最上位時間的レイヤの識別子である場合、ｔｅｍｐｏｒａｌ_ｉｄ＝Ｔａであり、ｐｒｉｏｒｉｔｙ_ｉｄ＝３１(または、他の特定値)であるＮＡＬＵを非参照ピクチャのＮＡＬＵであることを示すために使用する。

【００８７】

ｎａｌ_ｒｅｆ_ｆｌａｇをシグナリングするために使われた１ｂｉｔは、下記のうちいずれか一つとして使われることができる。

【００８８】

(１)ｎａｌ_ｕｎｉｔ_ｔｙｐｅの指示に使われることができる。ｎａｌ_ｕｎｉｔ_ｔｙｐｅは、７ｂｉｔｓ信号になることができ、ＮＡＬＵタイプの個数は、２倍に増加することができる。

【００８９】

(２)ｔｅｍｐｏｒａｌ_ｉｄの指示に使われることができる。ｔｅｍｐｏｒａｌ_ｉｄは、４ｂｉｔｓ信号になることができ、最大時間的レイヤの数が２倍に増加することができる。

【００９０】

（３）ｌａｙｅｒ_ｉｄを指示するために使われることができる。ｌａｙｅｒ_ｉｄは、階層的ビットストリームのスケーラブル階層の識別子を意味し、ｒｅｓｅｒｖｅｄ_ｏｎｅ_５ｂｉｔｓシンタックス要素によってシグナリングされることができる。スケーラブル階層を識別するために使われたｒｅｓｅｒｖｅｄ_ｏｎｅ_５ｂｉｔｓの５ｂｉｔにｎａｌ_ｒｅｆ_ｆｌａｇのシグナリングのために使用した１ｂｉｔが追加されることで、ｌａｙｅｒ_ｉｄは６ｂｉｔ信号になることができる。６ｂｉｔを使用するようになると、６４個のスケーラブル階層を識別することができる。

【００９１】

（４）ｒｅｓｅｒｖｅｄ_ｏｎｅ_５ｂｉｔｓがｐｒｉｏｒｉｔｙを示すかどうかを知らせるｆｌａｇとして使われることができる。

【００９２】

（５）ｒｅｓｅｒｖｅｄ_ｂｉｔとして使われることができる。

【００９３】

もし、ｎａｌ_ｒｅｆ_ｆｌａｇをＮＡＬＵヘッダから削除しない場合、ｎａｌ_ｒｅｆ_ｆｌａｇの意味は、下記のように修正されることができる。

【００９４】

ｎａｌ_ｒｅｆ_ｆｌａｇが０の場合、ＮＡＬＵは、非参照ピクチャのスライスのみを含むことを指示し、ｎａｌ_ｒｅｆ_ｆｌａｇが１の場合、ＮＡＬＵは、参照ピクチャまたは非参照ピクチャのスライスを含むことができることを指示する。

【００９５】

ビデオパラメータセットの活性化シグナリング

【００９６】

ビデオパラメータセット(ｖｉｄｅｏ ｐａｒａｍｅｔｅｒ ｓｅｔ)は、映像をデコーディングするための最も基本的な情報を含み、既存のＳＰＳに存在した内容を含むことができる。

【００９７】

ビデオパラメータセットには、時間的スケーラビリティをサポートする時間的レイヤを示すサブレイヤ(ｓｕｂ－ｌａｙｅｒ)に対する情報と、空間的(ｓｐａｔｉａｌ)、クオリティ的(ｑｕａｌｉｔｙ)及び視点的(ｖｉｅｗ)スケーラビリティをサポートする複数の階層に対する情報とを含むことができる。即ち、ビデオパラメータセットは、複数の階層情報、即ち、ＨＥＶＣ ｅｘｔｅｎｓｉｏｎのためのシンタックスを含むこともできる。

【００９８】

Ａ．ビデオパラメータセット(Ｖｉｄｅｏ Ｐａｒａｍｅｔｅｒ Ｓｅｔ)

【００９９】

ビデオパラメータセットに対するシンタックスは、表３の通りである。

【０１００】

【表３】

video_parameter_set_rbsp( ) {	Descriptor
video_parameter_set_id	ue(v)
max_temporal_layers_minus1	u(3)
priority_id_flag	u(1)
extension_info_flag	u(1)
if(extension_info_flag = = 0) {	
pic_width_in_luma_samples	ue(v)
pic_height_in_luma_samples	ue(v)
pic_cropping_flag	u(1)
if(pic_cropping_flag) {	
pic_crop_left_offset	ue(v)
pic_crop_right_offsetv	ue(v)
pic_crop_top_offset	ue(v)
pic_crop_bottom_offset	ue(v)
}	
bit_depth_luma_minus8	ue(v)
bit_depth_chroma_minus8	ue(v)
chroma_format_idc	u(2)
temporal_id_nesting_flag	u(1)
for (i = 0; i <= num_temporal_layers_minus1; i++) {	
profile_idc[i]	u(8)
reserved_zero_8bits[i] /* equal to 0 */	u(8)
level_idc[i]	u(8)
}	
tiles_or_entropy_coding_sync_idc	u(2)
if(tiles_or_entropy_coding_sync_idc = = 1) {	
num_tile_columns_minus1	ue(v)
num_tile_rows_minus1	ue(v)
uniform_spacing_flag	u(1)
if(!uniform_spacing_flag) {	
for(i = 0; i < num_tile_columns_minus1; i++)	
column_width[i]	ue(v)
for(i = 0; i < num_tile_rows_minus1; i++)	
row_height[i]	ue(v)
}	
loop_filter_across_tiles_enabled_flag	u(1)
}	
}	
if(extension_info_flag = = 1) {	
extension_info()	
}	
rbsp_trailing_bits( )	
}	

【０１０１】

表３において、大部分のシンタックスは、単一レイヤを含むビットストリームに適用されるＳＰＳシンタックスと同じ意味を有し、追加的な部分は、下記の通りである。

【０１０２】

－ｖｉｄｅｏ_ｐａｒａｍｅｔｅｒ_ｓｅｔ_ｉｄは、ビデオパラメータセット(ｖｉｄｅｏ ｐａｒａｍｅｔｅｒ ｓｅｔ)の識別子を意味し、ＳＰＳ(ｓｅｑｕｅｎｃｅ ｐａｒａｍｅｔｅｒ ｓｅｔ)、ＳＥＩ(ｓｕｐｐｌｅｍｅｎｔａｌ ｅｎｈａｎｃｅｍｅｎｔ ｉｎｆｏｒｍａｔｉｏｎ)、アクセスユニットデリミタ(ａｃｃｅｓｓ ｕｎｉｔ ｄｅｌｉｍｉｔｅｒ)で参照されることができる。

【０１０３】

－ｐｒｉｏｒｉｔｙ_ｉｄ_ｆｌａｇが１の場合、ｒｅｓｅｒｖｅｄ_ｏｎｅ_５ｂｉｔｓがＳＶＣ標準のｐｒｉｏｒｉｔｙ_ｉｄと同じく使われることを意味し、ｐｒｉｏｒｉｔｙ_ｉｄ_ｆｌａｇが０の場合、ｒｅｓｅｒｖｅｄ_ｏｎｅ_５ｂｉｔｓがｌａｙｅｒ_ｉｄとして使われることを意味する。

【０１０４】

－ｅｘｔｅｎｓｉｏｎ_ｉｎｆｏ_ｆｌａｇが０の場合、ビットストリームがＨＥＶＣの単一階層標準に従うことを指示し、１の場合、スケーラビリティのサポートのための向上階層(ＨＥＶＣ ｅｘｔｅｎｓｉｏｎをサポートする場合)を示し、階層と関連した情報が提供される。

【０１０５】

Ｂ．シーケンスパラメータセット(Ｓｅｑｕｅｎｃｅ Ｐａｒａｍｅｔｅｒ Ｓｅｔ、ＳＰＳ)修正

【０１０６】

表４のように、既存シンタックスのうち一部は、ＶＰＳに反映され、ＳＰＳから削除されることができる。一方、ＳＰＳにはｖｐｓ_ｉｄシンタックス要素が追加されることができる。ｖｐｓ_ｉｄが追加されたＳＰＳシンタックスは、表４の通りである。表４において、削除されたシンタックスは、シンタックス中間に線を引いて表現される。

【０１０７】

ｖｐｓ_ｉｄは、ＳＰＳにより参照されるビデオパラメータセット(ｖｉｄｅｏ ｐａｒａｍｅｔｅｒ ｓｅｔ)を識別するための識別子を指示し、ｖｐｓ_ｉｄは、０～Ｘの範囲を有することができる。

【０１０８】

【表４】

seq_parameter_set_rbsp( ) {	Descriptor
~~profile_idc~~	~~u(8)~~
~~reserved_zero_8bits /* equal to 0 */~~	~~u(8)~~
~~level_idc~~	~~u(8)~~
seq_parameter_set_id	ue(v)
vps_id	ue(v)
…	
~~pic_width_in_luma_samples~~	~~ue(v)~~
~~pic_height_in_luma_samples~~	~~ue(v)~~
~~pic_cropping_flag~~	~~u(1)~~
~~if(pic_cropping_flag) {~~	
~~pic_crop_left_offset~~	~~ue(v)~~
~~pic_crop_right_offset~~	~~ue(v)~~
~~pic_crop_top_offset~~	~~ue(v)~~
~~pic_crop_bottom_offset~~	~~ue(v)~~
~~}~~	
~~max_temporal_layers_minus1~~	~~u(3)~~
~~bit_depth_luma_minus8~~	~~ue(v)~~
~~bit_depth_chroma_minus8~~	~~ue(v)~~
~~chroma_format_idc~~	~~u(2)~~
~~temporal_id_nesting_flag~~	~~u(1)~~
…	
~~tiles_or_entropy_coding_sync_idc~~	~~u(2)~~
~~if(tiles_or_entropy_coding_sync_idc == 1) {~~	
~~num_tile_columns_minus1~~	~~ue(v)~~
~~num_tile_rows_minus1~~	~~ue(v)~~
~~uniform_spacing_flag~~	~~u(1)~~
~~if(!uniform_spacing_flag) {~~	
~~for(i = 0; i < num_tile_columns_minus1; i++)~~	
~~column_width[i]~~	~~ue(v)~~
~~for(i = 0; i < num_tile_rows_minus1; i++)~~	
~~row_height[i]~~	~~ue(v)~~
~~}~~	
~~loop_filter_across_tiles_enabled_flag~~	~~u(1)~~
~~}~~	
sps_extension_flag	u(1)
if(sps_extension_flag)	
while(more_rbsp_data( ))	
sps_extension_data_flag	u(1)
rbsp_trailing_bits( )	
}	

【０１０９】

Ｃ．ビデオパラメータセット(Ｖｉｄｅｏ Ｐａｒａｍｅｔｅｒ Ｓｅｔ)のための活性化(ａｃｔｉｖａｔｉｏｎ)シグナリング

【０１１０】

スライスヘッダには、該当スライスが参照するピクチャパラメータセットに対するインデックス情報が含まれており、ピクチャパラメータセットには、該当ピクチャが参照するシーケンスパラメータセットに対するインデックス情報が含まれている。シーケンスパラメータセットには、該当シーケンスが参照するビデオパラメータセットに対する情報が含まれている。このようにパラメータセットに対する情報をパーシングし、パーシングされた該当パラメータセット情報を参照することを活性化(ａｃｔｉｖａｔｉｏｎ)という。

【０１１１】

特定パラメータセットに対する情報を利用するために、即ち、パラメータセットを活性化するためにはスライスヘッダから順次にパーシングされなければならない。いずれのＳＰＳが活性化されるか(ａｃｔｉｖｅ)を知るために、全てのスライスヘッダ(ｓｌｉｃｅ ｈｅａｄｅｒ)及び関連したＰＰＳが分析されなければならないことを意味する。

【０１１２】

単一階層を含むビットストリームのうち、サブレイヤ(時間的レイヤ)の中から一部を抽出する時、抽出器(ｅｘｔｒａｃｔｏｒ)は、ＮＡＬＵヘッダと複数のパラメータセット(ｐａｒａｍｅｔｅｒ ｓｅｔ)を分析(パーシング)する必要がある。

【０１１３】

もし、ビデオパラメータセットまたはシーケンスパラメータセットにＮＡＬＵの抽出のための情報が含まれている場合、抽出器は、スライスヘッダから順次に上位のパラメータセットをパーシングしなければならない。これは抽出器がパラメータセット(ｐａｒａｍｅｔｅｒ ｓｅｔ)とスライスヘッダ(ｓｌｉｃｅ ｈｅａｄｅｒ)の全てのシンタックス要素を理解しなければならないということを意味する。

【０１１４】

また、映像のデコーディング過程でも複雑なパーシング過程無しでｖｐｓ_ｉｄまたはｓｐｓ_ｉｄを探し、必要なパラメータセットのみを活性化することができる。この場合、ビデオパラメータセットまたはシーケンスパラメータセットが活性化されるパラメータインデックス情報を含む場合、複雑なスライスヘッダ(ｓｌｉｃｅ ｈｅａｄｅｒ)及び関連したＰＰＳに対するパーシング手順を減少させることができる。

【０１１５】

一方、このようなシンタックスの要素のうち、一部分のみがビットストリーム抽出のために必要な情報を含むことができる。それにも拘わらず、抽出器が全てのシンタックス要素を分析することは、大きい負担になることができる。このような問題を解決するために、下記のような方法を提案する。

【０１１６】

本発明において、パラメータセットの活性化(ａｃｔｉｖａｔｉｏｎ)は、抽出器がスライスヘッダ(ｓｌｉｃｅ ｈｅａｄｅｒ)及びそれと関連したＰＰＳ(ｐｉｃｔｕｒｅ ｐａｒａｍｅｔｅｒ ｓｅｔ)の分析無しでいずれのパラメータセットが活性化されるかを知ることができるようにシグナリングすることを意味する。

【０１１７】

本発明によると、いずれのビデオパラメータセット、シーケンスパラメータセットまたはピクチャパラメータセットが活性化されるかどうかを別途にシグナリングすることで、抽出器は、全てのスライスヘッダ(ｓｌｉｃｅ ｈｅａｄｅｒ)及び関連したＰＰＳを分析しなければならない負担を減少させることができる。

【０１１８】

ビデオパラメータセットは、アップデートされることもできる。抽出器がスライスヘッダを分析せずに、現在活性化されるＶＰＳ及び関連したＳＰＳまたはＰＰＳを知ることができるように、下記の方法のうち一つが使われることができる。

【０１１９】

(1)ｖｐｓ_ｉｄ、ｓｐｓ_ｉｄ、ｐｐｓ_ｉｄをアクセスユニットデリミタ(ａｃｃｅｓｓ ｕｎｉｔ ｄｅｌｉｍｉｔｅｒ)に含ませることができる。ｖｐｓ_ｉｄ、ｓｐｓ_ｉｄ、ｐｐｓ_ｉｄは、各々、関連したＡＵ内のＮＡＬＵのために使われたビデオパラメータセット、シーケンスパラメータセット、ピクチャパラメータセットの識別子を示す。

【０１２０】

アクセスユニットデリミタ(Ａｃｃｅｓｓ ｕｎｉｔ ｄｅｌｉｍｉｔｅｒ)内に各識別子の存在可否を示すために、ｖｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇ、ｓｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇ、ｐｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇを使用し、提案するアクセスユニットデリミタのシンタックスは、表５の通りである。

【０１２１】

【表５】

access_unit_delimiter_rbsp( ) {	**Descriptor**
pic_type	u(3)
vps_id_present_flag	u(1)
sps_id_present_flag	u(1)
pps_id_present_flag	u(1)
if(vps_id_present_flag)	
vps_id	ue(v)
if(sps_id_present_flag)	
sps_id	ue(v)
if(pps_id_present_flag)	
pps_id	ue(v)
rbsp_trailing_bits( )	
}	

【０１２２】

(１－１)他の方法は、表６のように、ｓｐｓ_ｉｄ及びｐｐｓ_ｉｄを除いてｖｐｓ_ｉｄのみをアクセスユニットデリミタ(ａｃｃｅｓｓ ｕｎｉｔ ｄｅｌｉｍｉｔｅｒ)に含ませることができる。

【０１２３】

【表６】

access_unit_delimiter_rbsp( ) {	**Descriptor**
pic_type	u(3)
vps_id	ue(v)
rbsp_trailing_bits( )	
}	

【０１２４】

(２)ビデオパラメータセットの活性化シグナリングのための他の方法は、新たなＳＥＩメッセージ(ｐａｒａｍｅｔｅｒ_ｓｅｔ_ｒｅｆｅｒｅｎｃｅ)を使用することである。ＳＥＩメッセージは、関連したＡＵ内のＮＡＬＵのために使われたビデオパラメータセット、シーケンスパラメータセット、ピクチャパラメータセットの識別子を示すｖｐｓ_ｉｄ、ｓｐｓ_ｉｄ、ｐｐｓ_ｉｄの存在可否を知らせるためのシンタックスを含む。

【０１２５】

各識別子の存在可否を示すために、ｖｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇ、ｓｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇ、ｐｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇシンタックスが使

われることができ、ＳＥＩシンタックスは、表７の通りである。

【０１２６】

【表７】

parameter_set_reference(payloadSize) {	**Descriptor**
vps_id_present_flag	u(1)
sps_id_present_flag	u(1)
pps_id_present_flag	u(1)
if(vps_id_present_flag)	
vps_id	ue(v)
if(sps_id_present_flag)	
sps_id	ue(v)
if(pps_id_present_flag)	
pps_id	ue(v)
}	

【０１２７】

（２－１）また、表８のようにｐｐｓ_ｉｄを除いてｓｐｓ_ｉｄとｖｐｓ_ｉｄをＳＥＩメッセージに含ませて活性化を知らせることもできる。ＳＥＩメッセージに含まれているｓｐｓ_ｉｄとｖｐｓ_ｉｄは、該当ＳＥＩメッセージと関連したアクセスユニット(ａｃｃｅｓｓ ｕｎｉｔ)のビデオコーディングレイヤＮＡＬＵが参照するｓｐｓ_ｉｄとｖｐｓ_ｉｄを含むことができる。したがって、ｓｐｓ_ｉｄとｖｐｓ_ｉｄは、活性化される可能性があるパラメータセットの情報を示すことができる。

【０１２８】

【表８】

parameter_set_reference(payloadSize) {	**Descriptor**
vps_id	u(4)
sps_id_present_flag	u(1)
if(sps_id_present_flag)	
sps_id	ue(v)
psr_extension_flag	u(1)
if(psr_extension_flag){	
psr_extension_length	ue(v)
for(i=0; i<psr_extension_length; i++)	
psr_extension_data_byte	u(8)
}	
}	

【０１２９】

表８において、ｖｐｓ_ｉｄは、現在活性化されるビデオパラメータセットのｖｉｄｅｏ_ｐａｒａｍｅｔｅｒ_ｓｅｔ_ｉｄを示す。ｖｐｓ_ｉｄ値は、０～１５の値を有することができる。

【０１３０】

ｓｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇが１の値を有する場合、現在活性化されるシーケンスパラメータセットのｓｅｑｕｅｎｃｅ_ｐａｒａｍｅｔｅｒ_ｓｅｔ_ｉｄが該当ＳＥＩメッセージに含まれていることを示し、ｓｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇが０

の値を有する場合、活性化されるシーケンスパラメータセットのｓｅｑｕｅｎｃｅ_ｐａｒａｍｅｔｅｒ_ｓｅｔ_ｉｄが該当ＳＥＩメッセージに含まれていないことを示す。

【０１３１】

ｓｐｓ_ｉｄは、現在活性化されるシーケンスパラメータセットのｓｅｑｕｅｎｃｅ_ｐａｒａｍｅｔｅｒ_ｓｅｔ_ｉｄを示す。ｓｐｓ_ｉｄは、０～３１の値、より限定的には、０～１５の値を有することができる。

【０１３２】

ｐｓｒ_ｅｘｔｅｎｓｉｏｎ_ｆｌａｇが０の場合、ｐａｒａｍｅｔｅｒ ｓｅｔ ｒｅｆｅｒｅｎｃｅ ＳＥＩ ｍｅｓｓａｇｅ ｅｘｔｅｎｓｉｏｎシンタックス要素がｐａｒａｍｅｔｅｒ ｓｅｔ ｒｅｆｅｒｅｎｃｅ ＳＥＩメッセージに含まれていないことを意味し、ｐｓｒ_ｅｘｔｅｎｓｉｏｎ_ｆｌａｇが１の場合、ｐａｒａｍｅｔｅｒ ｓｅｔ ｒｅｆｅｒｅｎｃｅ ＳＥＩ ｍｅｓｓａｇｅ ｅｘｔｅｎｓｉｏｎシンタックス要素をｐａｒａｍｅｔｅｒ ｓｅｔ ｒｅｆｅｒｅｎｃｅ ＳＥＩメッセージが含まれてシンタックスを拡張して使用することを意味する。

【０１３３】

ｐｓｒ_ｅｘｔｅｎｓｉｏｎ_ｌｅｎｇｔｈは、ｐｓｒ_ｅｘｔｅｎｓｉｏｎ_ｄａｔａの長さを示す。ｐｓｒ_ｅｘｔｅｎｓｉｏｎ_ｌｅｎｇｔｈは、０～２５６の範囲を値を有することができ、ｐｓｒ_ｅｘｔｅｎｓｉｏｎ_ｄａｔａ_ｂｙｔｅは、どのような値も有することができる。

【０１３４】

（２－２）また、表９のように、ｐｐｓ_ｉｄを除いて一つ以上のｓｐｓ_ｉｄとｖｐｓ_ｉｄをＳＥＩメッセージに含ませてシグナリングすることもできる。

【０１３５】

【表９】

parameter_set_reference(payloadSize) {	**Descriptor**
vps_id	u(4)
num_referenced_sps	u(5)
for(i=0; i< num_referenced_sps; i++)	
sps_id(i)	ue(v)
}	

【０１３６】

表９において、ｖｐｓ_ｉｄは、現在活性化されるビデオパラメータセットのｖｉｄｅｏ_ｐａｒａｍｅｔｅｒ_ｓｅｔ_ｉｄを示す。ｖｐｓ_ｉｄは、０～１５の値を有することができる。

【０１３７】

ｎｕｍ_ｒｅｆｅｒｅｎｃｅ_ｓｐｓは、現在活性化されるｖｐｓ_ｉｄを参照するシーケンスパラメータセットの個数を示す。

【０１３８】

ｓｐｓ_ｉｄ（ｉ）は、現在活性化されるシーケンスパラメータセットのｓｅｑｕｅｎｃ

ｅ_ｐａｒａｍｅｔｅｒ_ｓｅｔ_ｉｄを示し、ｓｐｓ_ｉｄは、０～３１の値、より限定的には、０～１５の値を有することができる。

【０１３９】

（２－３)また、表１０のように、ｓｐｓ_ｉｄ及びｐｐｓ_ｉｄを除いてｖｐｓ_ｉｄのみをＳＥＩメッセージに含ませてシグナリングすることもできる。

【０１４０】

【表１０】

parameter_set_reference(payloadSize) {	**Descriptor**
vps_id	ue(v)
}	

【０１４１】

（３)ビデオパラメータセットの活性化シグナリングのための他の方法は、Ｂｕｆｆｅｒｉｎｇ ｐｅｒｉｏｄ ＳＥＩメッセージにｖｐｓ_ｉｄ、ｓｐｓ_ｉｄ、ｐｐｓ_ｉｄを知らせる情報を含ませることである。表１１は、ｖｐｓ_ｉｄ、ｓｐｓ_ｉｄ、ｐｐｓ_ｉｄ識別子の存在可否を示すためのｖｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇ、ｓｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇ、ｐｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇを含んでいるシンタックスを示す。

【０１４２】

【表１１】

buffering_period(payloadSize) {	**Descriptor**
vps_id_present_flag	u(1)
sps_id_present_flag	u(1)
pps_id_present_flag	u(1)
if(vps_id_present_flag)	
vps_id	ue(v)
if(sps_id_present_flag)	
sps_id	ue(v)
if(pps_id_present_flag)	
pps_id	ue(v)
.....	
}	

【０１４３】

（３－１)また、表１２のように、ｓｐｓ_ｉｄ及びｐｐｓ_ｉｄを除いてｖｐｓ_ｉｄのみをＢｕｆｆｅｒｉｎｇ ｐｅｒｉｏｄ ＳＥＩメッセージに含ませてパラメータセットの活性化をシグナリングすることもできる。

【０１４４】

【表１２】

buffering_period(payloadSize) {	**Descriptor**
vps_id_present_flag	u(1)
if(vps_id_present_flag)	
vps_id	ue(v)
.....	
}	

【０１４５】
（４）パラメータセットの活性化シグナリングのための他の方法は、Ｒｅｃｏｖｅｒｙ ｐｏｉｎｔ ＳＥＩメッセージにｖｐｓ_ｉｄ、ｓｐｓ_ｉｄ、ｐｐｓ_ｉｄを知らせる情報を含ませることである。表１３は、ｖｐｓ_ｉｄ、ｓｐｓ_ｉｄ、ｐｐｓ_ｉｄ識別子の存在可否を示すためのｖｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇ、ｓｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇ、ｐｐｓ_ｉｄ_ｐｒｅｓｅｎｔ_ｆｌａｇを含んでいるシンタックスを示す。
【０１４６】
【表１３】

recovery_point(payloadSize) {	**Descriptor**
vps_id_present_flag	**u(1)**
sps_id_present_flag	**u(1)**
pps_id_present_flag	**u(1)**
if(vps_id_present_flag)	
vps_id	**ue(v)**
if(sps_id_present_flag)	
sps_id	**ue(v)**
if(pps_id_present_flag)	
pps_id	**ue(v)**
recovery_poc_cnt	**ue(v)**
exact_match_flag	**u(1)**
broken_link_flag	**u(1)**
}	
recovery_point(payloadSize) {	**Descriptor**
vps_id_present_flag	**u(1)**
if(vps_id_present_flag)	
vps_id	**ue(v)**
recovery_poc_cnt	ue(v)
exact_match_flag	u(1)
broken_link_flag	u(1)
}	

【０１４７】
（４－１）また、表１４のように、ｓｐｓ_ｉｄ及びｐｐｓ_ｉｄを除いてｖｐｓ_ｉｄのみをＲｅｃｏｖｅｒｙ ｐｏｉｎｔ ＳＥＩメッセージに含ませて知らせる方法もある。
【０１４８】

【表１４】

recovery_point(payloadSize) {	**Descriptor**
vps_id_present_flag	**u(1)**
if(vps_id_present_flag)	
vps_id	**ue(v)**
recovery_poc_cnt	ue(v)
exact_match_flag	u(1)
broken_link_flag	u(1)
}	

【０１４９】
前述したｖｐｓ_ｉｄまたはｓｐｓ_ｉｄを伝達するメッセージは、ＩＲＡＰ(ｉｎｔｒａ ｒａｎｄｏｍ ａｃｃｅｓｓ ｐｏｉｎｔ)アクセスユニットに含まれることができる。
【０１５０】
前述した情報シグナリング方法のうち少なくとも一つがアクセスユニットに含まれて使われる場合、抽出器は、ビットストリームを抽出するために、前記シグナリング方法を介してｖｐｓ_ｉｄ、ｓｐｓ_ｉｄ、ｐｐｓ_ｉｄ値を探し、一つ以上のｖｐｓ/ｓｐｓ/ｐｐｓを管理することができる。
【０１５１】
また、デコーディング装置またはデコーディングを実行するデコーディング部は、前記シグナリング方法を介してｖｐｓ_ｉｄ、ｓｐｓ_ｉｄ、ｐｐｓ_ｉｄ値を探し、該当パラメータセットを活性化してパラメータセットと関連したＡＵをデコーディングすることができる。
【０１５２】
拡張タイプ(Ｅｘｔｅｎｄｅｄ ｔｙｐｅ)におけるビットストリームの表現
【０１５３】
以下、階層拡張をサポートするビットストリームを含む場合、スケーラブル階層に対する情報を表示し、これをシグナリングするためのＶＰＳのｅｘｔｅｎｓｉｏｎ_ｉｎｆｏ()と新たなＳＥＩメッセージを提案する。拡張タイプ(Ｅｘｔｅｎｄｅｄ ｔｙｐｅ)において、ビットストリームを表現するためには下記のような情報がシグナリングされることができる。
【０１５４】
ｌａｙｅｒ_ｉｄは、レイヤの優先順位(ｐｒｉｏｒｉｔｙ)値を伝達するかどうかを示すものをシグナリングする。
【０１５５】
このとき、各ｌａｙｅｒ_ｉｄ値に対応して空間階層(ｄｅｐｅｎｄｅｎｃｙ_ｉｄ値により識別)、画質階層(ｑｕａｌｉｔｙ_ｉｄ値により識別)、視点(ｖｉｅｗ_ｉｄ値により識別)などをシグナリングすることができ、時間的レイヤは、ＮＡＬＵヘッダのｔｅｍｐｏｒａｌ_ｉｄにより識別されることができる。
【０１５６】
また、ｌａｙｅｒ_ｉｄと関連したビデオの領域はｒｅｇｉｏｎ_ｉｄによりシグナリングされることができる。
【０１５７】
また、スケーラブル階層のうち、ｄｅｐｅｎｄｅｎｃｙ情報、各スケーラブル階層のｂｉｔｒａｔｅ情報、各スケーラブル階層の品質情報がシグナリングされることができる。
【０１５８】
ｅｘｔｅｎｓｉｏｎ_ｉｎｆｏ()シンタックスは、表１５の通りである。
【０１５９】

【表１５】

extension_info( ){	
num_frame_sizes_minus1	ue(v)
num_rep_formats_minus1	ue(v)
for(i = 0; i <= num_frame_sizes_minus1; i++) {	
pic_width_in_luma_samples[i]	ue(v)
pic_height_in_luma_samples[i]	ue(v)
pic_cropping_flag[i]	u(1)
if(pic_cropping_flag[i]) {	
pic_crop_left_offset[i]	ue(v)
pic_crop_right_offsetv[i]	ue(v)
pic_crop_top_offset[i]	ue(v)
pic_crop_bottom_offset[i]	ue(v)
}	
}	
for(i = 0; i <= num_rep_formats_minus1; i++) {	
bit_depth_luma_minus8[i]	ue(v)
bit_depth_chroma_minus8[i]	ue(v)
chroma_format_idc[i]	u(2)
}	
num_layers_minus1	ue(v)
dependency_id_flag	u(1)
quality_id_flag	u(1)
view_id _flag	u(1)
region_id_flag	u(1)
layer_dependency_info_flag	u(1)
for(i = 1; i <= num_layers_minus1; i++) {	
frame_size_idx[i]	ue(v)
rep_format_idx[i]	ue(v)
if(dependency_id_flag) {	
one_dependency_id_flag[i]	u(1)
if (one_dependency_id_flag[i] = = 1)	
dependency_id[i]	ue(v)
else if(one_dependency_id_flag[i] = = 0) {	
dependency_id_min[i]	ue(v)
dependency_id_max[i]	ue(v)
}	
}	

if(quality_id_flag) {	
one_quality_id_flag[i]	u(1)
if (one_quality_id_flag[i] = = 1)	
quality_id[i]	ue(v)
else if(one_quality_id_flag[i] = = 0) {	
quality_id_min[i]	ue(v)
quality_id_max[i]	ue(v)
}	
}	
if(view_id_flag) {	
one_view_id_flag[i]	u(1)
depth_flag[i]	u(1)
if (one_view_id_flag[i] = = 1)	
view_id[i]	ue(v)
else if(one_view_id_flag[i] = = 0) {	
view_id_min[i]	ue(v)
view_id_max[i]	ue(v)
}	
}	
if(region_id_flag) {	
num_regions_minus1[i]	u(8)
for(j = 1; j <= num_regions_minus1[i]; j++)	
region_id[i][j]	u(8)
}	
if(layer_dependency_info_flag) {	
num_directly_dependent_layers[i]	u(8)
for(j = 0; j < num_directly_dependent_layers[i]; j++)	
directly_dependent_layer_id_delta_minus1[i][j]	u(8)
}	
}	
}	

【０１６０】
表１５のシンタックスに対する意味は、下記の通りである。
【０１６１】
－ｎｕｍ_ｆｒａｍｅ_ｓｉｚｅｓ_ｍｉｎｕｓ１ ｐｌｕｓ１は、符号化されたビデオシーケンス内に含まれている他の種類の映像の大きさ情報(例えば、ｐｉｃ_ｗｉｄｔｈ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ[ｉ]、ｐｉｃ_ｈｅｉｇｈｔ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ[ｉ]、ｐｉｃ_ｃｒｏｐｐｉｎｇ_ｆｌａｇ[ｉ]、ｐｉｃ_ｃｒｏｐｐｉｎｇ_ｆｌａｇ[ｉ]、ｐｉｃ_ｃｒｏｐ_ｌｅｆｔ_ｏｆｆｓｅｔ[ｉ]、ｐｉｃ_ｃｒｏｐ_ｒｉｇｈｔ_ｏｆｆｓｅｔｖ[ｉ]、ｐｉｃ_ｃｒｏｐ_ｔｏｐ_ｏｆｆｓｅｔ[ｉ]、ｐｉｃ_ｃｒｏｐ_ｂｏｔｔｏｍ_ｏｆｆｓｅｔ[ｉ])の最大個数を示す。ｎｕｍ_ｆｒａｍｅ_ｓｉｚｅｓ_ｍｉｎｕｓ１値は、０～Ｘの範囲を有することができる。他の種類の映像とは、異なる解像度を有する映像を含むことができる。
【０１６２】

－ｎｕｍ_ｒｅｐ_ｆｏｒｍａｔｓ_ｍｉｎｕｓ１ ｐｌｕｓ１は、符号化されたビデオシーケンス内に含まれている他の種類のビット深さ(ｂｉｔ ｄｅｐｔｈ)と色差フォーマット(ｃｈｒｏｍａ ｆｏｒｍａｔ)(例えば、ｂｉｔ_ｄｅｐｔｈ_ｌｕｍａ_ｍｉｎｕｓ８[ｉ]、ｂｉｔ_ｄｅｐｔｈ_ｃｈｒｏｍａ_ｍｉｎｕｓ８[ｉ]、及びｃｈｒｏｍａ_ｆｏｒｍａｔ_ｉｄｃ ｖａｌｕｅｓ[ｉ])の最大個数を示す。ｎｕｍ_ｒｅｐ_ｆｏｒｍａｔｓ_ｍｉｎｕｓ１値は、０～Ｘの範囲を有する。

【０１６３】

－ｐｉｃ_ｗｉｄｔｈ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ[ｉ]、ｐｉｃ_ｈｅｉｇｈｔ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ[ｉ]、ｐｉｃ_ｃｒｏｐｐｉｎｇ_ｆｌａｇ[ｉ]、ｐｉｃ_ｃｒｏｐｐｉｎｇ_ｆｌａｇ[ｉ]、ｐｉｃ_ｃｒｏｐ_ｌｅｆｔ_ｏｆｆｓｅｔ[ｉ]、ｐｉｃ_ｃｒｏｐ_ｒｉｇｈｔ_ｏｆｆｓｅｔｖ[ｉ]、ｐｉｃ_ｃｒｏｐ_ｔｏｐ_ｏｆｆｓｅｔ[ｉ]、ｐｉｃ_ｃｒｏｐ_ｂｏｔｔｏｍ_ｏｆｆｓｅｔ[ｉ]は、符号化されたビデオシーケンスのｉ番目のｐｉｃ_ｗｉｄｔｈ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ、ｐｉｃ_ｈｅｉｇｈｔ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ、ｐｉｃ_ｃｒｏｐｐｉｎｇ_ｆｌａｇ、ｐｉｃ_ｃｒｏｐｐｉｎｇ_ｆｌａｇ、ｐｉｃ_ｃｒｏｐ_ｌｅｆｔ_ｏｆｆｓｅｔ、ｐｉｃ_ｃｒｏｐ_ｒｉｇｈｔ_ｏｆｆｓｅｔｖ、ｐｉｃ_ｃｒｏｐ_ｔｏｐ_ｏｆｆｓｅｔ、ｐｉｃ_ｃｒｏｐ_ｂｏｔｔｏｍ_ｏｆｆｓｅｔ値を示す。

【０１６４】

－ｂｉｔ_ｄｅｐｔｈ_ｌｕｍａ_ｍｉｎｕｓ８[ｉ]、ｂｉｔ_ｄｅｐｔｈ_ｃｈｒｏｍａ_ｍｉｎｕｓ８[ｉ]、及びｃｈｒｏｍａ_ｆｏｒｍａｔ_ｉｄｃ[ｉ]は、符号化されたビデオシーケンスのｉ番目のｂｉｔ_ｄｅｐｔｈ_ｌｕｍａ_ｍｉｎｕｓ８、ｂｉｔ_ｄｅｐｔｈ_ｃｈｒｏｍａ_ｍｉｎｕｓ８、及びｃｈｒｏｍａ_ｆｏｒｍａｔ_ｉｄｃ値を示す。

【０１６５】

－ｎｕｍ_ｌａｙｅｒｓ_ｍｉｎｕｓ１は、ビットストリームで可能なスケーラブル階層の数を示す。

【０１６６】

－ｄｅｐｅｎｄｅｎｃｙ_ｉｄ_ｆｌａｇが１の場合、ｌａｙｅｒ_ｉｄ値と関連した一つ以上のｄｅｐｅｎｄｅｎｃｙ_ｉｄ値があることを示す。

【０１６７】

－ｑｕａｌｉｔｙ_ｉｄ_ｆｌａｇが１の場合、ｌａｙｅｒ_ｉｄ値と関連した一つ以上のｑｕａｌｉｔｙ_ｉｄ値があることを示す。

【０１６８】

－ｖｉｅｗ_ｉｄ_ｆｌａｇが１の場合、ｌａｙｅｒ_ｉｄ値と関連した一つ以上のｖｉｅｗ_ｉｄ値があることを示す。

【０１６９】

－ｒｅｇｉｏｎ_ｉｄ_ｆｌａｇが１の場合、ｌａｙｅｒ_ｉｄ値と関連した一つ以上のｒｅｇｉｏｎ_ｉｄ値があることを示す。

【０１７０】

－ｌａｙｅｒ_ｄｅｐｅｎｄｅｎｃｙ_ｉｎｆｏ_ｆｌａｇが１の場合、スケーラブル階層のｄｅｐｅｎｄｅｎｃｙ情報を提供することを示す。

【０１７１】

－ｆｒａｍｅ_ｓｉｚｅ_ｉｄｘ[ｉ]は、ｌａｙｅｒ_ｉｄ値がｉである階層に適用されるｆｒａｍｅ ｓｉｚｅのセットに対するインデックスを示す。ｆｒａｍｅ_ｓｉｚｅ_ｉｄｘ[ｉ]は、０～Ｘ範囲の値を有する。

【０１７２】

－ｒｅｐ_ｆｏｒｍａｔ_ｉｄｘ[ｉ]は、ｌａｙｅｒ_ｉｄ値がｉである階層に適用されるビット深さ(ｂｉｔ ｄｅｐｔｈ)と色差フォーマット(ｃｈｒｏｍａ ｆｏｒｍａｔ)のセットに対するインデックスを示す。ｒｅｐ_ｆｏｒｍａｔ_ｉｄｘ[ｉ]は、０～Ｘ範囲の値を有する。

【０１７３】

－ｏｎｅ_ｄｅｐｅｎｄｅｎｃｙ_ｉｄ_ｆｌａｇ[ｉ]が１の場合、ｌａｙｅｒ_ｉｄ値がｉと関連した一つのｄｅｐｅｎｄｅｎｃｙ_ｉｄのみが存在することを示し、ｏｎｅ_ｄｅｐｅｎｄｅｎｃｙ_ｉｄ_ｆｌａｇ[ｉ]が０の場合、ｌａｙｅｒ_ｉｄ値がｉと関連した二つまたはそれ以上のｄｅｐｅｎｄｅｎｃｙ_ｉｄ値が存在することを示す。

【０１７４】

－ｄｅｐｅｎｄｅｎｃｙ_ｉｄ[ｉ]は、ｌａｙｅｒ_ｉｄ値がｉと関連したｄｅｐｅｎｄｅｎｃｙ_ｉｄ値を示す。

【０１７５】

－ｄｅｐｅｎｄｅｎｃｙ_ｉｄ_ｍｉｎ[ｉ]及びｄｅｐｅｎｄｅｎｃｙ_ｉｄ_ｍａｘ[ｉ]は、ｌａｙｅｒ_ｉｄ値がｉと関連した最小ｄｅｐｅｎｄｅｎｃｙ_ｉｄ値と最大ｄｅｐｅｎｄｅｎｃｙ_ｉｄ値を各々示す。

【０１７６】

－ｏｎｅ_ｑｕａｌｉｔｙ_ｉｄ_ｆｌａｇ[ｉ]が１の場合、ｌａｙｅｒ_ｉｄ値がｉと関連した一つのｑｕａｌｉｔｙ_ｉｄのみ存在することを示し、ｏｎｅ_ｑｕａｌｉｔｙ_ｉｄ_ｆｌａｇ[ｉ]が０の場合、ｌａｙｅｒ_ｉｄ値がｉと関連した二つまたはそれ以上のｑｕａｌｉｔｙ_ｉｄ値が存在することを示す。

【０１７７】

－ｑｕａｌｉｔｙ_ｉｄ[ｉ]は、ｌａｙｅｒ_ｉｄ値がｉと関連したｑｕａｌｉｔｙ_ｉｄ値を示す。

【０１７８】

－ｑｕａｌｉｔｙ_ｉｄ_ｍｉｎ[ｉ]及びｑｕａｌｉｔｙ_ｉｄ_ｍａｘ[ｉ]は、ｌａｙｅｒ_ｉｄ値がｉと関連した最小ｑｕａｌｔｉｙ_ｉｄ値と最大ｑｕａｌｉｔｙ_ｉｄ値を各々示す。

【０１７９】

－ｏｎｅ_ｖｉｅｗ_ｉｄ_ｆｌａｇ[ｉ]が１の場合、ｌａｙｅｒ_ｉｄ値がｉと関連した一つのｖｉｅｗ_ｉｄが存在することを示し、０の場合、ｌａｙｅｒ_ｉｄ値がｉと関連した二つまたはそれ以上のｖｉｅｗ_ｉｄ値が存在することを示す。

【０１８０】

－ｖｉｅｗ_ｉｄ[ｉ]はｌａｙｅｒ_ｉｄ値がｉと関連したｖｉｅｗ_ｉｄ値を示す。

【０１８１】

－ｄｅｐｔｈ_ｆｌａｇ[ｉ]が１の場合、ｌａｙｅｒ_ｉｄ値がｉである現在スケーラブル階層が３Ｄビデオビットストリームの深さ情報を含んでいることを示す。

【０１８２】

－ｖｉｅｗ_ｉｄ_ｍｉｎ[ｉ]及びｖｉｅｗ_ｉｄ_ｍａｘ[ｉ]は、ｌａｙｅｒ_ｉｄ値がｉと関連した最小ｖｉｅｗ_ｉｄ値と最大ｖｉｅｗ_ｉｄ値を各々示す。

【０１８３】

－ｎｕｍ_ｒｅｇｉｏｎｓ_ｍｉｎｕｓ１ ｐｌｕｓ１は、ｌａｙｅｒ_ｉｄ値がｉと関連した領域の個数を示す。

【０１８４】

－ｒｅｇｉｏｎ_ｉｄ[ｊ]は、ｌａｙｅｒ_ｉｄ値がｉと関連した領域ｊの識別子を示す。

【０１８５】

－ｎｕｍ_ｄｉｒｅｃｔｌｙ_ｄｅｐｅｎｄｅｎｔ_ｌａｙｅｒｓ[ｉ]は、現在スケーラブル階層ｉが直接的に関連したスケーラブル階層(デコーディング時、予測信号の形成に必要な階層)の数を示す。

【０１８６】

－ｄｉｒｅｃｔｌｙ_ｄｅｐｅｎｄｅｎｔ_ｌａｙｅｒ_ｉｄ_ｄｅｌｔａ_ｍｉｎｕｓ１[ｉ][ｊ] ｐｌｕｓ１は、現在スケーラブル階層であるｌａｙｅｒ_ｉｄ[ｉ]と、現在スケーラブル階層が直接的に関連したｊ番目のスケーラブル階層の階層識別子との間の差を示す。ｊ番目の直接的に関連したスケーラブル階層の階層識別子は、(ｌａｙｅｒ_ｉｄ[ｉ]

－ｄｉｒｅｃｔｌｙ_ｄｅｐｅｎｄｅｎｔ_ｌａｙｅｒ_ｉｄ_ｄｅｌｔａ_ｍｉｎｕｓ１[ｉ][ｊ]－１)である。

【０１８７】

他の実施例に係るｅｘｔｅｎｓｉｏｎ_ｉｎｆｏ()シンタックスは、表１６の通りである。

【０１８８】

【表１６】

extension_info( ){	
num_frame_sizes_minus1	ue(v)
num_rep_formats_minus1	ue(v)
for(i = 0; i <= num_rep_formats_minus1; i++) {	
pic_width_in_luma_samples[i]	ue(v)
pic_height_in_luma_samples[i]	ue(v)
bit_depth_luma_minus8[i]	ue(v)
bit_depth_chroma_minus8[i]	ue(v)
chroma_format_idc[i]	u(2)
pic_cropping_flag[i]	u(1)
if(pic_cropping_flag[i]) {	
pic_crop_left_offset[i]	ue(v)
pic_crop_right_offsetv[i]	ue(v)
pic_crop_top_offset[i]	ue(v)
pic_crop_bottom_offset[i]	ue(v)
}	
}	
num_layers_minus1	ue(v)
dependency_id_flag	u(1)
quality_id_flag	u(1)
view_id _flag	u(1)
region_id_flag	u(1)
layer_dependency_info_flag	u(1)
for(i = 1; i <= num_layers_minus1; i++) {	
frame_size_idx[i]	ue(v)
rep_format_idx[i]	ue(v)
if(dependency_id_flag) {	
one_dependency_id_flag[i]	u(1)
if (one_dependency_id_flag[i] = = 1)	
dependency_id[i]	ue(v)
else if(one_dependency_id_flag[i] = = 0) {	
dependency_id_min[i]	ue(v)
dependency_id_max[i]	ue(v)
}	

【０１８９】

表１６に示すように、ｐｉｃ_ｗｉｄｔｈ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ[ｉ]及びｐｉｃ_ｈｅｉｇｈｔ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ[ｉ]、ｂｉｔ_ｄｅｐｔｈ_ｌｕｍａ_ｍｉｎｕｓ８[ｉ]、ｂｉｔ_ｄｅｐｔｈ_ｃｈｒｏｍａ_ｍｉｎｕｓ８[ｉ]、及びｃｈｒｏｍａ_ｆｏｒｍａｔ_ｉｄｃ[ｉ]は、異なるリプリゼンテーションフォーマットに対する情報

にシグナリングされることができる。

【０１９０】

他の実施例によると、ｐｉｃ_ｗｉｄｔｈ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ[ｉ]、ｐｉｃ_ｈｅｉｇｈｔ_ｉｎ_ｌｕｍａ_ｓａｍｐｌｅｓ[ｉ]、ｂｉｔ_ｄｅｐｔｈ_ｌｕｍａ_ｍｉｎｕｓ８[ｉ]、ｂｉｔ_ｄｅｐｔｈ_ｃｈｒｏｍａ_ｍｉｎｕｓ８[ｉ]、及びｃｈｒｏｍａ_ｆｏｒｍａｔ_ｉｄｃ[ｉ]は、異なる映像、即ち、異なる解像度を有するピクチャに対する情報にシグナリングされることができる。

【０１９１】

ビットレートとクオリティ情報のシグナリングのための活性化ＳＥＩメッセージに対するシンタックスは、表１７の通りである。

【０１９２】

【表１７】

bitrate_quality_info(payloadSize) {	**Descriptor**
num_layers_minus1	ue(v)
bitrate_info_flag	u(1)
quality_info_flag	u(1)
quality_type_flag	u(1)
for(i = 1; i <= num_layers_minus1; i++) {	
if(bitrate_info_flag) {	
max_bitrate[i]	u(8)
average_bitrate[i]	u(8)
}	
if(quality_info_flag) {	
quality_value[i]	u(8)
}	
}	
if(quality_type_flag) {	
QualityTypeUriIdx = 0	
do	
quality_type_uri[**QualityTypeUriIdx]**	b(8)
while(quality_type_uri[QualityTypeUriIdx++] != 0)	
}	
}	

【０１９３】

表１７のシンタックスに対する意味は、下記の通りである。

【０１９４】

－ｎｕｍ_ｌａｙｅｒｓ_ｍｉｎｕｓ１は、ビットストリームで提供可能なスケーラブル階層の数を示す。

【０１９５】

－ｂｉｔｒａｔｅ_ｉｎｆｏ_ｆｌａｇが１の場合、各々のスケーラブル階層に対するビットレート情報が提供されることを指示する。

【０１９６】

－ｑｕａｌｉｔｙ_ｉｎｆｏ_ｆｌａｇが１の場合、各々のスケーラブル階層に対するクオリティ値に対する情報が提供されることを示す。

【０１９７】

－ｑｕａｌｉｔｙ_ｔｙｐｅ_ｆｌａｇが１の場合、各々のスケーラブル階層に対するクオリティタイプに対する情報が提供されることを示す。

【０１９８】

－ｍａｘ_ｂｉｔｒａｔｅ[ｉ]は、ｌａｙｅｒ_ｉｄ値がｉであるスケーラブル階層の最大ビットレートを示し、ａｖｅｒａｇｅ_ｂｉｔｒａｔｅ[ｉ]は、ｌａｙｅｒ_ｉｄ値がｉであるスケーラブル階層の平均ビットレートを示す。

【０１９９】

－ｑｕａｌｉｔｙ_ｖａｌｕｅ[ｉ]は、スケーラブル階層ｉのクオリティ値を示す。

【０２００】

－ｑｕａｌｉｔｙ_ｔｙｐｅ_ｕｒｉ[ＱｕａｌｉｔｙＴｙｐｅＵｒｉＩｄｘ]は、ＵＴＦ－８ ｃｈａｒａｃｔｅｒで符号化されたｎｕｌｌ０ｔｅｒｍｉｎａｔｅｄ ｓｔｒｉｎｇのＱｕａｌｉｔｙＴｙｐｅＵｒｉＩｄｘ－ｔｈバイト(ｂｙｔｅ)であり、クオリティ値のタイプに対する表現を含むＵＲＩ(ｕｎｉｖｅｒｓａｌ ｒｅｓｏｕｒｃｅ ｉｄｅｎｔｉｆｉｅｒ)を示す。

【０２０１】

以下、効率的なビットストリーム抽出のための記述方式の改善として、ＶＰＳ(ｖｉｄｅｏ ｐａｒａｍｅｔｅｒ ｓｅｔ)を改善させる方案を提案する。

【０２０２】

階層参照(Ｌａｙｅｒ ｒｅｆｅｒｅｎｃｉｎｇ)

【０２０３】

複数の階層をサポートするビットストリームでｌａｙｅｒ_ｉｄとスケーラビリティ次元ＩＤ(ｓｃａｌａｂｉｌｉｔｙ ｄｉｍｅｎｓｉｏｎ ＩＤ)との間の関係を指示する方法として、ｌａｙｅｒ_ｉｄとスケーラビリティ次元ＩＤ(ｓｃａｌａｂｉｌｉｔｙ ｄｉｍｅｎｓｉｏｎ ＩＤ)との間のマッピング方法を知らせる第１の方法と、ｌａｙｅｒ_ｉｄのビットを分割(ｐａｒｔｉｔｉｏｎｉｎｇまたはｓｐｌｉｃｉｎｇ)して割り当てられたビットにどのような次元タイプが存在するかを知らせる第２の方法とが存在できる。

【０２０４】

複数の階層をサポートするビットストリームで次元タイプ(ｄｉｍｅｎｓｉｏｎ ｔｙｐｅ)とは、空間的スケーラビリティ、クオリティ的スケーラビリティのようなスケーラビリティのタイプを意味し、次元ＩＤ(ｄｉｍｅｎｓｉｏｎ ＩＤ)は、特定の次元タイプが有することができるレイヤに対するインデックスを意味する。

【０２０５】

複数の階層をサポートするビットストリームで、特定次元(ｄｉｍｅｎｓｉｏｎ)では特定階層(理解を助けるために、例えば、単一階層のビットストリームで時間的スケーラビリティをサポートする場合、時間的レイヤ(ｓｕｂ－ｌａｙｅｒ)３)が次の低い階層(例えば、時間的レイヤ(ｓｕｂ－ｌａｙｅｒ))を直接的に参照することは一般的である。

【０２０６】

また、例えば、空間スケーラビリティをサポートする場合は、空間レイヤ２が次の低い空間階層１を直接的に参照することを意味する。

【０２０７】

したがって、前記のような場合を示すために、基本参照(ｄｅｆａｕｌｔ ｄｉｒｅｃｔ ｄｅｐｅｎｄｅｎｃｙ)を有する次元を先に記述することを提案する。

【０２０８】

その後、特定連関性(ｄｅｐｅｎｄｅｎｃｙ)を階層(ｓｃａｌａｂｌｅ ｌａｙｅｒ)に対する説明パート(ｄｅｓｃｒｉｐｔｉｏｎ ｌｏｏｐ)で具体的に記述することができる。

【０２０９】

以下、前記二つの方法を利用して階層参照に対するシグナリングをするための方案を提示する。ｖｐｓ_ｅｘｔｅｎｓｉｏｎのための改善されたシンタックスは、表１８乃至表２１の通りである。

【０２１０】
【表１８】

vps_extension( ) {	Descriptor
while(!byte_aligned( ))	
vps_extension_byte_alignment_reserved_one_bit	u(1)
all_default_dependency_flag	u(1)
if (all_default_dependency_flag = = 0) {	
num_default_dim_minus1	u(4)
for(j = 0; j <= num_default_dim_minus1; j++) {	
dimension[j]	u(4)
}	
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
// mapping of layer ID to scalability dimension IDs	
num_dimensions_minus1[i]	u(4)
for(j = 0; j <= num_dimensions_minus1; j++) {	
dimension_type[i][j]	u(4)
dimension_id[i][j]	u(8)
}	
// layer dependency	
specific_dependency_flag[i]	u(1)
if(**specific_dependency_flag[i]** = =1) {	
num_direct_ref_layers[i]	u(6)
for(j = 0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	
}	

【０２１１】
表１８は、第１の方法を利用してｌａｙｅｒ_ｉｄとスケーラビリティ次元ＩＤ(ｓｃａｌａｂｉｌｉｔｙ ｄｉｍｅｎｓｉｏｎ ＩＤ)をマッピングさせているシンタックスを示している。表１８のシンタックスに対する意味は、下記の通りである。
【０２１２】
－ａｌｌ_ｄｅｆａｕｌｔ_ｄｅｐｅｎｄｅｎｃｙ_ｆｌａｇが１の場合、全ての階層次元が基本参照(ｄｅｆａｕｌｔ ｄｅｐｅｎｄｅｎｃｙ)を有することを指示する。即ち、特定次元ｉにおいて、ｄｉｍｅｎｓｉｏｎ_ｉｄ[ｉ]＝ｎである階層は、デフォルトとしてｄｉｍｅｎｓｉｏｎ_ｉｄ[ｉ]＝ｎ－１を有する他の階層を直接的に参照することを意味する。
【０２１３】
ａｌｌ_ｄｅｆａｕｌｔ_ｄｅｐｅｎｄｅｎｃｙ_ｆｌａｇが０の場合、全ての階層次元が基本参照を有するものではないことを示す。ａｌｌ_ｄｅｆａｕｌｔ_ｄｅｐｅｎｄｅｎｃｙ_ｆｌａｇが０の場合、下のｎｕｍ_ｄｅｆａｕｌｔ_ｄｉｍ_ｍｉｎｕｓ１がシグナリングされる。
【０２１４】
－ｎｕｍ_ｄｅｆａｕｌｔ_ｄｉｍ_ｍｉｎｕｓ１は、基本参照(ｄｅｆａｕｌｔ ｄｅｐ

ｅｎｄｅｎｃｙ)を有する次元の数を示す。
【０２１５】
－ｄｉｍｅｎｓｉｏｎ[ｊ]は、基本参照(ｄｅｆａｕｌｔ ｄｅｐｅｎｄｅｎｃｙ)を有する階層次元のタイプを明示する。即ち、基本参照(ｄｅｆａｕｌｔ ｄｅｐｅｎｄｅｎｃｙ)を有する次元の数を一つずつ増加させながら、基本参照を有する階層次元のタイプに対する情報がシグナリングされる。該当次元において、上位階層(例えば、ｄｉｍｅｎｓｉｏｎ_ｉｄ＝ｎ)は、次の下位階層(例えば、ｄｉｍｅｎｓｉｏｎ_ｉｄ＝ｎ－１)を直接的に参照する。
【０２１６】
－ｓｐｅｃｉｆｉｃ_ｄｅｐｅｎｄｅｎｃｙ_ｆｌａｇ[ｉ]が１の場合、該当階層のために具体的に記述された直接参照(ｄｉｒｅｃｔ ｄｅｐｅｎｄｅｎｃｅｓ/ｒｅｆｅｒｅｎｃｅｓ)があることを意味する。したがって、ｓｐｅｃｉｆｉｃ_ｄｅｐｅｎｄｅｎｃｙ_ｆｌａｇ[ｉ]が１の場合、該当階層が直接参照するレイヤの個数と該当レイヤのＩＤがシグナリングされる。
【０２１７】
階層Ｃが階層Ｂを直接的に参照するということは、階層Ｃを復号化するために、デコーダは、階層Ｂの情報(デコーディングされ、またはデコーディングされない)を使用しなければならないという意味である。しかし、もし、階層Ｂが直接的に階層Ａの情報を使用する場合、階層Ｃは階層Ａを直接的に参照すると判断されない。
【０２１８】

【表１９】

vps_extension( ) {	Descriptor
while(!byte_aligned( ))	
vps_extension_byte_alignment_reserved_one_bit	u(1)
all_default_dependency_flag	u(1)
if (all_default_dependency_flag = = 0) {	
num_default_dim_minus1	u(4)
for(j = 0; j <= num_default_dim_minus1; j++) {	
dimension[j]	u(4)
}	
}	
// scalability type and layer_id partitioning method	
~~**scalability_type**~~	~~u(4)~~
~~for(i = 0; i < MaxDim(scalability_type); i++)~~	
num_dimensions_minus1	
for(i = 0; i <= num_dimensions_minus1; i++) {	
dimension_type[i]	u(4)
layer_id_dim_len[i]	u(3)
}	
for(i = 0; i <= max_num_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
// layer dependency	
specific_dependency_flag[i]	u(1)
if(**specific_dependency_flag[i]** = =1) {	
num_direct_ref_layers[i]	u(6)
for(j = 0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	
}	

【０２１９】

表１９は、第２の方法を利用してｌａｙｅｒ_ｉｄのビットをスケーラビリティ次元タイプに割り当て、割り当てられた次元タイプの長さをシグナリングするシンタックスを示している。

【０２２０】

表１９に含まれているｎｕｍ_ｄｉｍｅｎｓｉｏｎｓ_ｍｉｎｕｓ１は、ＮＡＬＵヘッダ内に存在する階層次元の数を示す。即ち、ＮＡＬＵヘッダに存在する階層次元の数を把握し、該当階層次元毎に存在する階層タイプと次元タイプに割り当てられたビット数を把握する。

【０２２１】

表１９のシンタックス階層参照のためのシンタックスａｌｌ_ｄｅｆａｕｌｔ_ｄｅｐｅｎｄｅｎｃｙ_ｆｌａｇ、ｎｕｍ_ｄｅｆａｕｌｔ_ｄｉｍ_ｍｉｎｕｓ１、ｄｉｍｅｎｓｉｏｎ[ｊ]及びｓｐｅｃｉｆｉｃ_ｄｅｐｅｎｄｅｎｃｙ_ｆｌａｇ[ｉ]に対する説明は、表１８に含まれているシンタックスと同じ意味を有する。

【０２２２】

表２０及び表２１は、表１８及び表１９と異なる方式のシンタックスを示す。表２０は、第１の方法を利用する場合、基本参照(ｄｅｆａｕｌｔ　ｄｅｐｅｎｄｅｎｃｙ)を示す他のシンタックスを示し、表２１は、第２の方法を利用する場合、基本参照(ｄｅｆａｕｌｔ　ｄｅｐｅｎｄｅｎｃｙ)を示す他のシンタックスを示す。

【０２２３】

【表２０】

vps_extension( ) {	Descriptor
while(!byte_aligned( ))	
vps_extension_byte_alignment_reserved_one_bit	u(1)
num_dimensions_minus1]	u(4)
for(i = 0; i <= num_dimensions_minus1; i++) {	
dimension_type[i]	u(4)
default_dependency_flag [i]	u(1)
}	
for(i = 1; i <= vps_max_layers_minus1; i++) {	
// mapping of layer ID to scalability dimension IDs	
for(j = 0; j <= num_dimensions_minus1; j++) {	
dimension_id[i][j]	u(8)
}	
// layer dependency	
specific_dependency_flag [i]	u(1)
if(**specific_dependency_flag [i]** = =1) {	
num_direct_ref_layers[i]	u(6)
for(j = 0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	
}	

【０２２４】

【表２１】

vps_extension( ) {	Descriptor
while(!byte_aligned( ))	
vps_extension_byte_alignment_reserved_one_bit	u(1)
// scalability type and layer_id partitioning method	
num_dimensions_minus1	
for(i = 0; i <= num_dimensions_minus1; i++) {	
dimension_type[i]	u(4)
layer_id_dim_len[i]	u(3)
default_dependency_flag [i]	u(1)
}	
for(i = 0; i <= max_num_layers_minus1; i++) {	
vps_layer_id[i]	u(6)
// layer dependency	
specific_dependency_flag [i]	u(1)
if(**specific_dependency_flag [i]** = =1) {	
num_direct_ref_layers[i]	u(6)
for(j = 0; j < num_direct_ref_layers[i]; j++)	
ref_layer_id[i][j]	u(6)
}	
}	
}	

【０２２５】

表２０及び表２１のシンタックスのうち、表１８及び表１９と重複するシンタックスに対する説明は省略する。

【０２２６】

表２０及び表２１に含まれている新たなシンタックスｄｅｆａｕｌｔ_ｄｅｐｅｎｄｅｎｃｙ_ｆｌａｇ[ｉ]は、次元タイプｉが基本参照を使用するかどうかを示す。該当次元において、高い階層(例えば、ｄｉｍｅｎｓｉｏｎ_ｉｄ[ｉ]＝ｎ)は、下の階層(例えば、ｄｉｍｅｎｓｉｏｎ_ｉｄ[ｉ]＝ｎ－１)を直接的に参照する。

【０２２７】

即ち、ｎｕｍ_ｄｉｍｅｎｓｉｏｎｓ_ｍｉｎｕｓ１とｄｉｍｅｎｓｉｏｎ_ｔｙｐｅ[ｉ]によって特定次元タイプが指定される場合、該当次元タイプが基本参照を使用するかどうかをシグナリングし、そうでない場合、該当階層が直接的に参照するレイヤに対する情報をシグナリングする。

【０２２８】

本発明による次元タイプ(ｄｉｍｅｎｓｉｏｎｓ　ｔｙｐｅ)を示す場合、表２２の通りである。

【０２２９】

【表２２】

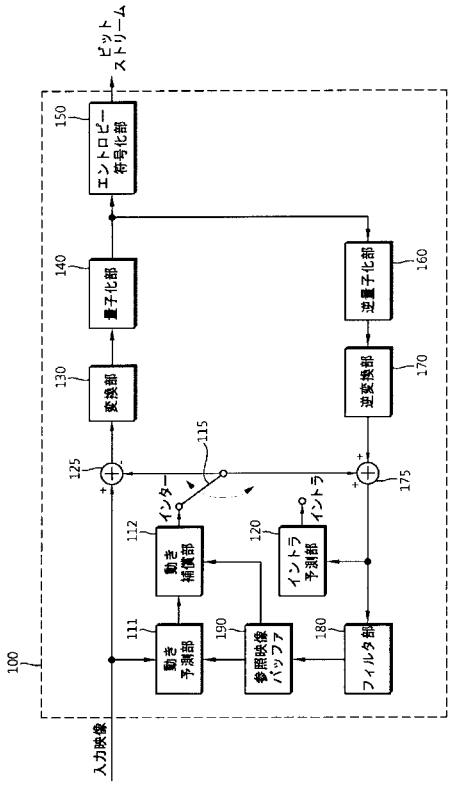
dimension_type[i][j]	dimension_id[i][j]
0	view order idx
1	depth flag
2	dependency ID
3	quality ID
4	priority ID
5	region ID
6..15	Reserved

【０２３０】
本発明によると、既存の次元タイプで次元タイプ４及び５、即ち、ｐｒｉｏｒｉｔｙ ＩＤ及びｒｅｇｉｏｎ ＩＤを示すタイプが追加された。
【０２３１】
ｄｉｍｅｎｓｉｏｎ_ｔｙｐｅ[ｉ][ｊ]は、基本的に０～５の値を有することができる。他の値は、以後に定義されることができ、デコーダは、０～５の値でない場合、ｄｉｍｅｎｓｉｏｎ_ｔｙｐｅ[ｉ][ｊ]の値を無視することができる。
【０２３２】
ｄｉｍｅｎｓｉｏｎ_ｔｙｐｅが４の値を有する場合、該当ｄｉｍｅｎｓｉｏｎ_ｉｄは、ＳＶＣ標準でビットストリームのｐｒｉｏｒｉｔｙ階層のｉｄを示す。
【０２３３】
ｄｉｍｅｎｓｉｏｎ_ｔｙｐｅが５の値を有する場合、該当ｄｉｍｅｎｓｉｏｎ_ｉｄは、ビットストリームの特定領域のｉｄを示す。特定領域は、ビットストリーム内で一つ以上の時空間セグメント(ｓｐａｔｉａｌ－ｔｅｍｐｏｒａｌ ｓｅｇｍｅｎｔ)になることができる。
【０２３４】
図４は、本発明による映像情報のエンコーディング方法を説明するための制御流れ図である。
【０２３５】
図示されているように、符号化装置は、映像に関連した情報を含むＮＡＬ(Ｎｅｔｗｏｒｋ Ａｂｓｔｒａｃｔｉｏｎ Ｌａｙｅｒ)ユニットをエンコーディングする(Ｓ４０１)。
【０２３６】
ＮＡＬユニットのＮＡＬユニットヘッダは、ＮＡＬユニットが非参照ピクチャの少なくとも一部または全部を含むスライスを含むかどうかを示す情報を含まない。
【０２３７】
一方、ＮＡＬユニットヘッダは、スケーラブル階層をサポートするビットストリームでスケーラブル階層を識別するための階層識別情報を含んでいる。
【０２３８】
このとき、ＮＡＬユニットヘッダに含まれないＮＡＬユニットが非参照ピクチャの少なくとも一部または全部を含むスライスを含むかどうかを示す情報をシグナリングするために使われたビットは、階層識別情報をシグナリングするために使われることができる。
【０２３９】
また、ＮＡＬユニットは、映像のデコーディングのために必要な多様な前記パラメータセットに対する情報を含むことができる。
【０２４０】
符号化装置は、活性化されるパラメータセットに対する情報を含むＳＥＩ(Ｓｕｐｐｌｅｍｅｎｔａｌ ｅｎｈａｎｃｅｍｅｎｔ ｉｎｆｏｒｍａｔｉｏｎ)メッセージを独立さ

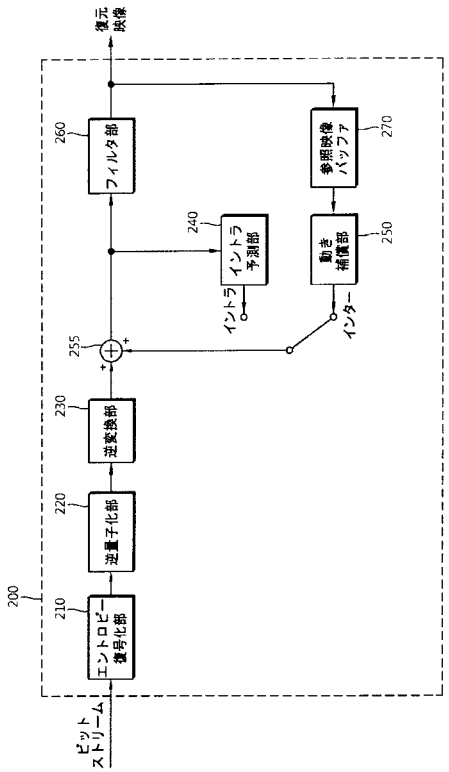
れたＮＡＬユニットにエンコーディングすることができる。
【０２４１】
活性化されるパラメータセットに対する情報は、活性化されるビデオパラメータセットをインデクシングする情報及び活性化されるシーケンスパラメータセットをインデクシングする情報のうち少なくとも一つを含むことができる。
【０２４２】
また、活性化されるパラメータセットに対する情報は、活性化されるビデオパラメータセットをインデクシングする情報、活性化されるビデオパラメータセットを参照するシーケンスパラメータセットの個数を示す情報、及びシーケンスパラメータセットをインデクシングする情報を含むことができる。
【０２４３】
このようなパラメータセットに対する情報は、復号化装置が時間的スケーラビリティを提供するサブレイヤを抽出する時に利用されることができる。
【０２４４】
また、デコーディング装置またはデコーディングを実行するデコーディング部は、ビデオコーディングレイヤＮＡＬＵのデコーディング時に必要なパラメータセットを活性化する時、前記パラメータセットに対する情報を利用することができる。
【０２４５】
符号化装置は、エンコーディングされた映像に関連した情報を含むＮＡＬユニットをビットストリームに送信する(Ｓ４０２)。
【０２４６】
図５は、本発明による映像情報のデコーディング方法を説明するための制御流れ図である。
【０２４７】
図５を参照すると、復号化装置は、ビットストリームを介してエンコーディングされた映像に関連した情報を含むＮＡＬユニットを受信する(Ｓ５０１)。
【０２４８】
復号化装置は、ＮＡＬユニットのヘッダ及びＮＡＬペイロード(ｐａｙｌｏａｄ)をパーシングする(Ｓ５０２)。映像情報に対するパーシングは、エントロピー復号化部または別途のパーシング部で実行されることができる。
【０２４９】
復号化装置は、パーシングを介してＮＡＬユニットヘッダ及びＮＡＬペイロードに含まれている多様な情報を取得することができる。
【０２５０】
ＮＡＬユニットヘッダは、スケーラブル階層をサポートするビットストリームでスケーラブル階層を識別するための階層識別情報を含み、ＮＡＬユニットのエンコーディング時、全体ビットストリームで非参照ピクチャであるか、または参照ピクチャであるかを示す１ビットのフラグ情報を含まない。
【０２５１】
このとき、ＮＡＬユニットヘッダに含まれないＮＡＬユニットが非参照ピクチャの少なくとも一部または全部を含むスライスを含むかどうかを示す情報をシグナリングするために使われたビットは、階層識別情報をシグナリングするために使われることができる。
【０２５２】
また、復号化装置は、パーシングを介してＳＥＩメッセージに含まれている該当ＳＥＩメッセージと関連したＮＡＬＵをデコーディングするために必要なパラメータセットに対する情報を取得することができる。
【０２５３】
活性化されるパラメータセットに対する情報は、活性化されるビデオパラメータセットをインデクシングする情報及び活性化されるシーケンスパラメータセットをインデクシングする情報のうち少なくとも一つを含むことができる。

【０２５４】
また、活性化されるパラメータセットに対する情報は、活性化されるビデオパラメータセットをインデクシングする情報、活性化されるビデオパラメータセットを参照するシーケンスパラメータセットの個数を示す情報、及びシーケンスパラメータセットをインデクシングする情報を含むことができる。
【０２５５】
このようなパラメータセットに対する情報は、復号化装置が時間的スケーラビリティを提供するサブレイヤを抽出する時に利用されることができる。
【０２５６】
追加的にパラメータセットに対する情報は、ビットストリームを復号化する時またはセッションネゴシエーション(例えば、ＩＰ網おけるストリーミング時、ｓｅｓｓｉｏｎ ｎｅｇｏｔｉａｔｉｏｎ)する時に利用されることができる。
【０２５７】
前述した実施例において、方法は一連のステップまたはブロックで流れ図に基づいて説明されているが、本発明は、ステップの順序に限定されるものではなく、あるステップは、前述と異なるステップと、異なる順序にまたは同時に発生できる。また、当該技術分野において、通常の知識を有する当業者であれば、流れ図に示すステップが排他的でなく、他のステップが含まれ、または流れ図の一つまたはそれ以上のステップが本発明の範囲に影響を及ぼさすに削除可能であることを理解することができる。
【０２５８】
前述した実施例は、多様な態様の例示を含む。多様な態様を示す全ての可能な組合せを記述することはできないが、当該技術分野の通常の知識を有する者であれば、他の組合せが可能であることを認識することができる。したがって、本発明は、特許請求の範囲内に属する全ての交替、修正、及び変更を含む。

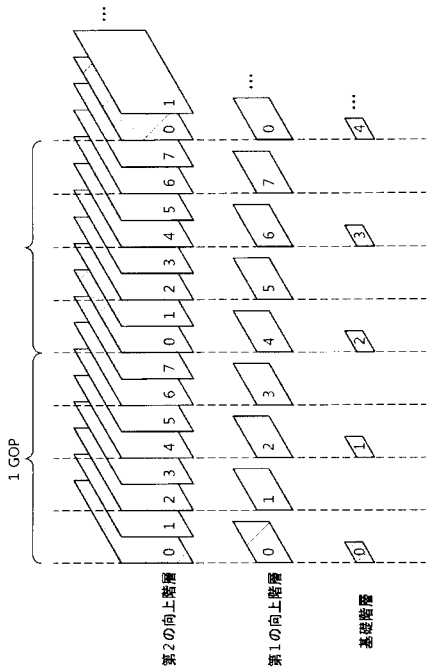
【図１】



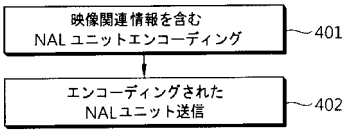
【図２】



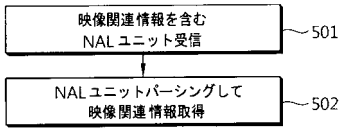
【図３】


1 GOP
第２の向上階層
第１の向上階層
基礎階層

【図４】


映像関連情報を含む
NAL ユニットエンコーディング
401
エンコーディングされた
NALユニット送信
402

【図５】


映像関連情報を含む
NAL ユニット受信
501
NALユニットパーシングして
映像関連情報取得
502

フロントページの続き

(31)優先権主張番号 10-2012-0071933
(32)優先日 平成24年7月2日(2012.7.2)
(33)優先権主張国 韓国(KR)
(31)優先権主張番号 10-2012-0077012
(32)優先日 平成24年7月16日(2012.7.16)
(33)優先権主張国 韓国(KR)
(31)優先権主張番号 10-2012-0108925
(32)優先日 平成24年9月28日(2012.9.28)
(33)優先権主張国 韓国(KR)
(31)優先権主張番号 10-2012-0112598
(32)優先日 平成24年10月10日(2012.10.10)
(33)優先権主張国 韓国(KR)
(31)優先権主張番号 10-2013-0041862
(32)優先日 平成25年4月16日(2013.4.16)
(33)優先権主張国 韓国(KR)

(74)代理人 100105153
弁理士 朝倉 悟
(74)代理人 100152205
弁理士 吉田 昌司
(72)発明者 カン、ジョン、ウォン
大韓民国テジョン、ユソン－グ、クワンピョン－ドン、ウナム、ネオミア、６０９－５０１
(72)発明者 リー、ハ、ヒョン
大韓民国ソウル特別市、チュンナン－グ、ミョンモク、２－ドン、１３６－５
(72)発明者 チェ、ジン、ス
大韓民国テジョン、ユソン－グ、バンソク－ドン、６１３、バンソクマウル、６、ダンジ、アパート、６０９－１６０５
(72)発明者 キム、ジン、ウン
大韓民国テジョン、ユソン－グ、ジョンミン－ドン、エキスポ、アパート、３０５－１６０３

Ｆターム(参考) 5C159 MA04 MA05 MA21 MA31 MC11 ME01 NN01 RB09 UA02 UA05
UA16 UA33