明　細　書

発明の名称 ： 符号化方法及び復号方法

技術分野

[0001] 本発明の実施形態は、符号化方法及び復号方法に関する。

背景技術

[0002] 近年、符号化効率を大幅に向上させた画像符号化方法が、ＩＴＵ－Ｔ（International Telecommunication Union Telecommunication Standardization Sector）とＩＳＯ（International Organization for Standardization）／ＩＥＣ（International Electrotechnical Commission）との共同で、ＩＴＵ－Ｔ　ＲＥＣ．　Ｈ．２６４及びＩＳＯ／ＩＥＣ　１４４９６－１０（以下、「Ｈ．２６４」という）として勧告されている。

[0003] Ｈ．２６４には、符号化済みの画像を参照画像に用いて分数精度の動き補償予測を行うことにより、時間方向の冗長性を削除し、高い符号化効率を実現するインター予測符号化方式が開示されている。

[0004] また、ＩＳＯ／ＩＥＣ　ＭＰＥＧ（Moving Picture Experts Group）－１，２，４におけるインター予測符号化方式よりも、フェードやディゾルブ効果を含む動画像を高効率に符号化する方式も提案されている。この方式では、時間方向における明度変化を予測する枠組みとして、輝度と２つの色差とを有する入力動画像に対して分数精度の動き補償予測を行う。そして、参照画像と、輝度及び２つの色差毎の重み係数と、輝度及び２つの色差毎のオフセットと、を含む組み合わせ示すインデックスを用いて、予測画像に重み係数を乗じ、オフセットを加算する。

先行技術文献

特許文献

[0005] 特許文献1：特開２００４－７３７７号公報

発明の概要

発明が解決しようとする課題

[0006] しかしながら、上述したような従来技術では、双方向予測を選択可能な双方向予測スライスにおいて、参照画像が同一であるが参照画像番号が異なる２つのインデックスを用いて重み付き動き補償を行う際に、同一の値を持つインデックスを２回符号化する場合が発生し、符号化効率が低下する場合がある。本発明が解決しようとする課題は、符号化効率を向上できる符号化方法及び復号方法を提供することである。

課題を解決するための手段

[0007] 実施形態の符号化方法は、インデックス生成ステップと、符号化ステップとを、含む。インデックス生成ステップでは、第１参照画像が参照する１以上の参照画像を含む組み合わせを示す第１インデックスと第２参照画像が参照する１以上の参照画像を含む組み合わせを示す第２インデックスとを設定し、前記第１インデックスに含まれる１以上の参照画像の番号と前記第２インデックスに含まれる１以上の参照画像の番号とを、予め定められた順序でスキャンして、同一の参照画像を含まない組み合わせに並び替えた共通インデックスを生成する。符号化ステップでは、前記共通インデックスを符号化する。

図面の簡単な説明

[0008] [図1]第１実施形態の符号化装置の例を示すブロック図。

[図2]第１実施形態における画素ブロックの予測符号化順序例を示す説明図。

[図3A]第１実施形態におけるコーディングツリーブロックのブロックサイズ例を示す図。

[図3B]第１実施形態のコーディングツリーブロックの具体例を示す図。

[図3C]第１実施形態のコーディングツリーブロックの具体例を示す図。

[図3D]第１実施形態のコーディングツリーブロックの具体例を示す図。

[図4]第１実施形態の予測画像生成部の例を示すブロック図。

[図5]第１実施形態の双方向予測における動き補償予測の動きベクトルの関係の例を示す図。

[図6]第１実施形態の複数フレーム動き補償部の例を示すブロック図。

[図7]第１実施形態における重み係数の固定小数点精度の例の説明図。

[図8]第１実施形態のインデックス設定部の例を示すブロック図。

[図9A]第１実施形態のＷＰパラメータ情報例を示す図。

[図9B]第１実施形態のＷＰパラメータ情報例を示す図。

[図10]第１実施形態の低遅延符号化構造における符号化順と表示順の例を示す図。

[図11]第１実施形態の低遅延符号化構造における参照画像と参照番号との関係の例を示す図。

[図12]第１実施形態のランダムアクセス符号化構造における符号化順と表示順の例を示す図。

[図13]第１実施形態のランダムアクセス符号化構造における参照画像と参照番号との関係の例を示す図。

[図14]第１実施形態の参照画像のリスト番号及び参照番号のスキャン順序の例を示す図。

[図15]第１実施形態における共通リスト変換後のＷＰパラメータ情報の例を簡略化して示した図。

[図16]第１実施形態における共通リスト変換後のＷＰパラメータ情報の例を簡略化して示した図。

[図17]第１実施形態のインデックス情報の生成処理例を示すフローチャート。

[図18]第１実施形態のシンタクスの例を示す図。

[図19]第１実施形態のピクチャパラメータセットシンタクスの例を示す図。

[図20]第１実施形態のスライスヘッダーシンタクスの例を示す図。

[図21]第１実施形態のプレッドウェイトテーブルシンタクスの例を示す図。

[図22]変形例のシーケンスパラメータセットシンタクスの例を示す図。

[図23]変形例のアダプテーションパラメータセットシンタクスの例を示す図。

[図24]変形例のプレッドウェイトテーブルシンタクスの例を示す図。

[図25]第２実施形態のインデックス設定部の構成例を示すブロック図。

[図26]第２実施形態のインデックス情報の生成処理例を示すフローチャート。

[図27]第２実施形態のプレッドウェイトテーブルシンタクスの例を示す図。

[図28]第３実施形態の符号化装置の構成例を示すブロック図。

[図29]第３実施形態の動き情報メモリの詳細例を示す図。

[図30A]第３実施形態の符号化画素ブロックに対する動き情報候補を導出するブロック位置の例を示す図。

[図30B]第３実施形態の符号化画素ブロックに対する動き情報候補を導出するブロック位置の例を示す図。

[図31]第３実施形態の複数の動き情報候補の画素ブロック位置と画素ブロック位置インデクスとの関係の例を示す図。

[図32]第３実施形態のMergeCandListへの格納処理例を示すフローチャート。

[図33]第３実施形態の動き情報の格納リストの例を示す図。

[図34]第３実施形態の動き情報の格納リストへの格納方法の例を示すフローチャート。

[図35]第３実施形態の双方向予測の組み合わせ例を示す図。

[図36]第３実施形態のプレッドユニットシンタクスの例を示す図。

[図37]第３実施形態のMergeCandListへの格納処理の他の例を示すフローチャート。

[図38]第４実施形態の復号装置の構成例を示すブロック図。

[図39]第４実施形態のインデックス設定部の構成例を示すブロック図。

[図40]第５実施形態のインデックス設定部の構成例を示すブロック図。

[図41]第６実施形態の復号装置の構成例を示すブロック図。

発明を実施するための形態

[0009] 以下、添付図面を参照しながら、実施形態を詳細に説明する。以下の各実施形態の符号化装置及び復号装置は、ＬＳＩ（Large-Scale Integration）チップ、ＤＳＰ（Digital Signal Processor）、又はＦＰＧＡ（Field Pr

ogrammable Gate Array）などのハードウェアにより実現できる。また、以下の各実施形態の符号化装置及び復号装置は、コンピュータにプログラムを実行させること、即ち、ソフトウェアにより実現させることもできる。なお、以降の説明において、「画像」という用語は、「映像」、「画素」、「画像信号」、「絵」、又は「画像データ」などの用語に適宜読み替えることができる。

[0010] （第１実施形態）

第１実施形態では、動画像を符号化する符号化装置について説明する。

[0011] 図１は、第１実施形態の符号化装置１００の構成の一例を示すブロック図である。

[0012] 符号化装置１００は、入力画像を構成する各フレーム又は各フィールドを複数の画素ブロックに分割し、符号化制御部１１１から入力される符号化パラメータを用いて、分割した画素ブロックに対して予測符号化を行い、予測画像を生成する。そして符号化装置１００は、複数の画素ブロックに分割した入力画像と予測画像とを減算して予測誤差を生成し、生成した予測誤差を直交変換及び量子化し、更にエントロピー符号化を行って符号化データを生成し、出力する。

[0013] 符号化装置１００は、画素ブロックのブロックサイズ及び予測画像の生成方法の少なくともいずれかが異なる複数の予測モードを選択的に適用して予測符号化を行う。予測画像の生成方法は、大別すると、符号化対象フレーム内で予測を行うイントラ予測と、時間的に異なる１以上の参照フレームを用いて動き補償予測を行うインター予測との２種類である。なお、イントラ予測は、画面内予測又はフレーム内予測などとも称され、インター予測は、画面間予測、フレーム間予測、又は動き補償予測などとも称される。

[0014] 図２は、第１実施形態における画素ブロックの予測符号化順序の一例を示す説明図である。図２に示す例では、符号化装置１００は、画素ブロックの左上から右下に向かって予測符号化を行っており、符号化処理対象のフレームｆにおいて、符号化対象画素ブロックｃよりも左側及び上側に符号化済み

画素ブロックｐが位置している。以下では、説明の簡単化のため、符号化装置１００は、図２に示す順序で予測符号化を行うものとするが、予測符号化の順序はこれに限定されるものではない。

[0015] 画素ブロックは、画像を処理する単位を示し、例えば、Ｍ×Ｎサイズのブロック（Ｍ及びＮは自然数）、コーディングツリーブロック、マクロブロック、サブブロック、又は１画素などが該当する。以降の説明では、基本的に、画素ブロックをコーディングツリーブロックの意味で使用するが、他の意味で使用する場合もある。例えば、プレディクションユニットの説明では、画素ブロックを、プレディクションユニットの画素ブロックの意味で使用する。また、ブロックはユニットなどの名称で呼ばれることもある。例えばコーディングブロックをコーディングユニットと呼ぶ。

[0016] 図３Ａは、第１実施形態におけるコーディングツリーブロックのブロックサイズの一例を示す図である。コーディングツリーブロックは、典型的には、図３Ａに示すような６４×６４の画素ブロックである。但し、これに限定されるものではなく、３２×３２の画素ブロック、１６×１６の画素ブロック、８×８の画素ブロック、又は４×４の画素ブロックなどであってもよい。また、コーディングツリーブロックは、正方形でなくてもよく、例えば、Ｍ×Ｎサイズ（Ｍ≠Ｎ）の画素ブロックであってもよい。

[0017] 図３Ｂ～図３Ｄは、第１実施形態のコーディングツリーブロックの具体例を示す図である。図３Ｂは、ブロックサイズが６４×６４（Ｎ＝３２）のコーディングツリーブロックを示している。Ｎは、基準となるコーディングツリーブロックのサイズを表しており、分割された場合のサイズがＮ、分割されない場合のサイズが２Ｎと定義されている。図３Ｃは、図３Ｂのコーディングツリーブロックを四分木分割したコーディングツリーブロックを示している。コーディングツリーブロックは、図３Ｃに示すように、四分木構造を持つ。コーディングツリーブロックが分割された場合、分割後の４つの画素ブロックに対して、図３Ｃに示すように、Ｚスキャン順で番号が付される。

[0018] なお、コーディングツリーブロックは、１つの四分木の番号内で更に四分

木分割することができる。これにより、コーディングツリーブロックを階層的に分割することができる。この場合、分割の深さは、Ｄｅｐｔｈで定義される。図３Ｄは、図３Ｂのコーディングツリーブロックを四分木分割したコーディングツリーブロックの１つを示し、ブロックサイズが３２×３２（Ｎ＝１６）となっている。図３Ｂに示すコーディングツリーブロックのＤｅｐｔｈは０であり、図３Ｄに示すコーディングツリーブロックのＤｅｐｔｈは、１である。なお、最もユニットが大きいコーディングツリーブロックは、ラージコーディングツリーブロックと呼ばれ、この単位で入力画像信号がラスタースキャン順に符号化される。

[0019] 以降の説明では、入力画像の符号化対象ブロック又はコーディングツリーブロックを予測対象ブロック又は予測画素ブロックと称することもある。なお、符号化単位は画素ブロックに限らず、フレーム、フィールド、スライス、ライン、及び画素の少なくともいずれかを用いることもできる。

[0020] 符号化装置１００は、図１に示すように、減算部１０１と、直交変換部１０２と、量子化部１０３と、逆量子化部１０４と、逆直交変換部１０５と、加算部１０６と、予測画像生成部１０７と、インデックス設定部１０８と、動き評価部１０９と、符号化部１１０とを、備える。なお、図１に示す符号化制御部１１１は、符号化装置１００を制御するものであり、例えば、ＣＰＵ（Central Processing Unit）などにより実現できる。

[0021] 減算部１０１は、画素ブロックに分割された入力画像から対応する予測画像を減算して予測誤差を得る。減算部１０１は、予測誤差を出力し、直交変換部１０２に入力する。

[0022] 直交変換部１０２は、減算部１０１から入力された予測誤差に対して、例えば、離散コサイン変換（ＤＣＴ）又は離散サイン変換（ＤＳＴ）のような直交変換を行い、変換係数を得る。直交変換部１０２は、変換係数を出力し、量子化部１０３に入力する。

[0023] 量子化部１０３は、直交変換部１０２から入力された変換係数に対して量子化処理を行い、量子化変換係数を得る。具体的には、量子化部１０３は、

符号化制御部１１１によって指定される量子化パラメータや量子化マトリクスなどの量子化情報に従って量子化を行う。より詳細には、量子化部１０３は、変換係数を量子化情報によって導出される量子化ステップサイズで除算し、量子化変換係数を得る。量子化パラメータは、量子化の細かさを示す。量子化マトリクスは、量子化の細かさを変換係数の成分毎に重み付けするために使用される。量子化部１０３は、量子化変換係数を出力し、逆量子化部１０４及び符号化部１１０に入力する。

[0024] 逆量子化部１０４は、量子化部１０３から入力された量子化変換係数に対して逆量子化処理を行い、復元変換係数を得る。具体的には、逆量子化部１０４は、量子化部１０３において使用された量子化情報に従って逆量子化を行う。より詳細には、逆量子化部１０４は、量子化情報によって導出された量子化ステップサイズを量子化変換係数に乗算し、復元変換係数を得る。なお、量子化部１０３において使用された量子化情報は、符号化制御部１１１の図示せぬ内部メモリからロードされて利用される。逆量子化部１０４は、復元変換係数を出力し、逆直交変換部１０５に入力する。

[0025] 逆直交変換部１０５は、逆量子化部１０４から入力された復元変換係数に対して、例えば、逆離散コサイン変換（ＩＤＣＴ）又は逆離散サイン変換（ＩＤＳＴ）などのような逆直交変換を行い、復元予測誤差を得る。なお、逆直交変換部１０５が行う逆直交変換は、直交変換部１０２において行われた直交変換に対応する。逆直交変換部１０５は、復元予測誤差を出力し、加算部１０６に入力する。

[0026] 加算部１０６は、逆直交変換部１０５から入力された復元予測誤差と対応する予測画像とを加算し、局所復号画像を生成する。加算部１０６は、局所復号画像を出力し、予測画像生成部１０７に入力する。

[0027] 予測画像生成部１０７は、加算部１０６から入力された局所復号画像を参照画像としてメモリ（図１では図示省略）に蓄積し、メモリに蓄積した参照画像を出力し、動き評価部１０９に入力する。また予測画像生成部１０７は、動き評価部１０９から入力される動き情報及びＷＰパラメータ情報に基づ

いて重み付き動き補償予測を行い、予測画像を生成する。予測画像生成部１０７は、予測画像を出力し、減算部１０１及び加算部１０６に入力する。

[0028] 図４は、第１実施形態の予測画像生成部１０７の構成の一例を示すブロック図である。予測画像生成部１０７は、図４に示すように、複数フレーム動き補償部２０１と、メモリ２０２と、単方向動き補償部２０３と、予測パラメータ制御部２０４と、参照画像セレクタ２０５と、フレームメモリ２０６と、参照画像制御部２０７と、を備える。

[0029] フレームメモリ２０６は、参照画像制御部２０７の制御の下、加算部１０６から入力された局所復号画像を参照画像として格納する。フレームメモリ２０６は、参照画像を一時保持するための複数のメモリセットＦＭ１～ＦＭＮ（Ｎ≧２）を有する。

[0030] 予測パラメータ制御部２０４は、動き評価部１０９から入力される動き情報に基づいて、参照画像番号と予測パラメータとの複数の組み合わせをテーブルとして用意している。ここで、動き情報とは、動き補償予測で用いられる動きのズレ量を示す動きベクトルや参照画像番号、単方向／双方向予測などの予測モードに関する情報などを指す。予測パラメータは、動きベクトル及び予測モードに関する情報を指す。そして予測パラメータ制御部２０４は、入力画像に基づいて、予測画像の生成に用いる参照画像番号と予測パラメータとの組み合わせを選択して出力し、参照画像番号を参照画像セレクタ２０５に入力し、予測パラメータを単方向動き補償部２０３に入力する。

[0031] 参照画像セレクタ２０５は、フレームメモリ２０６が有するフレームメモリＦＭ１～ＦＭＮのいずれの出力端を接続するかを、予測パラメータ制御部２０４から入力された参照画像番号に従って切り替えるスイッチである。参照画像セレクタ２０５は、例えば、参照画像番号が０であれば、ＦＭ１の出力端を参照画像セレクタ２０５の出力端に接続し、参照画像番号がＮ－１であれば、ＦＭＮの出力端を参照画像セレクタ２０５の出力端に接続する。参照画像セレクタ２０５は、フレームメモリ２０６が有するフレームメモリＦＭ１～ＦＭＮのうち、出力端が接続されているフレームメモリに格納されて

いる参照画像を出力し、単方向動き補償部２０３及び動き評価部１０９へ入力する。

[0032] 単方向予測動き補償部２０３は、予測パラメータ制御部２０４から入力された予測パラメータと参照画像セレクタ２０５から入力された参照画像に従って、動き補償予測処理を行い、単方向予測画像を生成する。

[0033] 図５は、第１実施形態の双方向予測における動き補償予測の動きベクトルの関係の一例を示す図である。動き補償予測では、参照画像を用いて補間処理が行われ、作成された補間画像と入力画像との符号化対象位置の画素ブロックからの動きのズレ量を元に単方向予測画像が生成される。ここで、ズレ量は、動きベクトルである。図５に示すように、双方向予測スライス（Ｂ－ｓｌｉｃｅ）では、２種類の参照画像と動きベクトルのセットを用いて予測画像が生成される。補間処理としては、１／２画素精度の補間処理や、１／４画素精度の補間処理などが用いられ、参照画像に対してフィルタリング処理が行われることによって、補間画像の値が生成される。例えば、輝度信号に対して１／４画素精度までの補間処理な可能なＨ．２６４では、ズレ量は整数画素精度の４倍で表現される。

[0034] 単方向予測動き補償部２０３は、単方向予測画像を出力し、メモリ２０２に一時的に格納する。ここで、動き情報（予測パラメータ）が双方向予測を示す場合には、複数フレーム動き補償部２０１が２種類の単方向予測画像を用いて重み付き予測を行うため、単方向予測動き補償部２０３は、１つ目に対応する単方向予測画像をメモリ２０２に格納し、２つ目に対応する単法予測画像を複数フレーム動き補償部２０１に直接出力する。ここでは、１つ目に対応する単方向予測画像を第一予測画像とし、２つ目に対応する単方向予測画像を第二予測画像とする。

[0035] なお、単方向動き補償部２０３を２つ用意し、それぞれが２つの単方向予測画像を生成するようにしてもよい。この場合、動き情報（予測パラメータ）が単方向予測を示すときには、単方向動き補償部２０３が、１つ目の単方向予測画像を第一予測画像として複数フレーム動き補償部２０１に直接出力

すればよい。

[0036] 複数フレーム動き補償部２０１は、メモリ２０２から入力される第一予測画像、単方向予測動き補償部２０３から入力される第二予測画像、及び動き評価部１０９から入力されるＷＰパラメータ情報を用いて、重み付き予測を行って予測画像を生成する。複数フレーム動き補償部２０１は、予測画像を出力し、減算部１０１及び加算部１０６に入力する。

[0037] 図６は、第１実施形態の複数フレーム動き補償部２０１の構成の一例を示すブロック図である。複数フレーム動き補償部２０１は、図６に示すように、デフォルト動き補償部３０１と、重み付き動き補償部３０２と、ＷＰパラメータ制御部３０３と、ＷＰセレクタ３０４、３０５とを、備える。

[0038] ＷＰパラメータ制御部３０３は、動き評価部１０９から入力されるＷＰパラメータ情報に基づいて、ＷＰ適用フラグ及び重み情報を出力し、ＷＰ適用フラグをＷＰセレクタ３０４、３０５に入力し、重み情報を重み付き動き補償部３０２に入力する。

[0039] ここで、ＷＰパラメータ情報は、重み係数の固定小数点精度、第一予測画像に対応する第一ＷＰ適用フラグ，第一重み係数，及び第一オフセット、並びに第二予測画像に対応する第二ＷＰ適応フラグ，第二重み係数，及び第二オフセットの情報を含む。ＷＰ適用フラグは、該当する参照画像及び信号成分毎に設定可能なパラメータであり、重み付き動き補償予測を行うかどうかを示す。重み情報は、重み係数の固定小数点精度、第一重み係数、第一オフセット、第二重み係数、及び第二オフセットの情報を含む。

[0040] 詳細には、ＷＰパラメータ制御部３０３は、動き評価部１０９からＷＰパラメータ情報が入力されると、ＷＰパラメータ情報を第一ＷＰ適用フラグ、第二ＷＰ適用フラグ、及び重み情報に分離して出力し、第一ＷＰ適用フラグをＷＰセレクタ３０４に入力し、第二ＷＰ適用フラグをＷＰセレクタ３０５に入力し、重み情報を重み付き動き補償部３０２に入力する。

[0041] ＷＰセレクタ３０４、３０５は、ＷＰパラメータ制御部３０３から入力されたＷＰ適用フラグに基づいて、各々の予測画像の接続端を切り替える。Ｗ

Ｐセレクタ３０４、３０５は、各々のＷＰ適用フラグが０の場合、各々の出力端をデフォルト動き補償部３０１へ接続する。そしてＷＰセレクタ３０４、３０５は、第一予測画像及び第二予測画像を出力し、デフォルト動き補償部３０１に入力する。一方、ＷＰセレクタ３０４、３０５は、各々のＷＰ適用フラグが１の場合、各々の出力端を重み付き動き補償部３０２へ接続する。そしてＷＰセレクタ３０４、３０５は、第一予測画像及び第二予測画像を出力し、重み付き動き補償部３０２に入力する。

[0042] デフォルト動き補償部３０１は、ＷＰセレクタ３０４、３０５から入力された２つの単方向予測画像（第一予測画像及び第二予測画像）を元に平均値処理を行い、予測画像を生成する。具体的には、デフォルト動き補償部３０１は、第一ＷＰ適用フラグ及び第二ＷＰ適用フラグが０の場合、数式（１）に基づいて平均値処理を行う。

[0043] Ｐ［ｘ，ｙ］＝Ｃｌｉｐ１（（ＰＬ０［ｘ，ｙ］＋ＰＬ１［ｘ，ｙ］＋ｏｆｆｓｅｔ２）＞＞（ｓｈｉｆｔ２））　…（１）

[0044] ここで、Ｐ［ｘ，ｙ］は予測画像、ＰＬ０［ｘ，ｙ］は第一予測画像、ＰＬ１［ｘ，ｙ］は第二予測画像である。ｏｆｆｓｅｔ２及びｓｈｉｆｔ２は平均値処理における丸め処理のパラメータであり、第一予測画像及び第二予測画像の内部演算精度によって定まる。予測画像のビット精度をＬ、第一予測画像及び第二予測画像のビット精度をＭ（Ｌ≦Ｍ）とすると、ｓｈｉｆｔ２は数式（２）で定式化され、ｏｆｆｓｅｔ２は数式（３）で定式化される。

[0045] ｓｈｉｆｔ２＝（Ｍ－Ｌ＋１）　…（２）

[0046] ｏｆｆｓｅｔ２＝（１＜＜（ｓｈｉｆｔ２－１）　…（３）

[0047] 例えば、予測画像のビット精度が８であり、第一予測画像及び第二予測画像のビット精度が１４である場合、数式（２）よりｓｈｉｆｔ２＝７、数式（３）よりｏｆｆｓｅｔ２＝（１＜＜６）となる。

[0048] なお、動き情報（予測パラメータ）で示される予測モードが単方向予測である場合、デフォルト動き補償部３０１は、第一予測画像のみを用いて、数

式（４）に基づいて最終的な予測画像を算出する。

[0049] Ｐ［ｘ，ｙ］＝Ｃｌｉｐ１（（ＰＬＸ［ｘ，ｙ］＋ｏｆｆｓｅｔ１）＞＞（ｓｈｉｆｔ１））　…（４）

[0050] ここで、ＰＬＸ［ｘ，ｙ］は単方向予測画像（第一予測画像）を示しており、Ｘは参照リストの０又は１のいずれかを示す識別子である。例えば、参照リストが０の場合はＰＬ０［ｘ，ｙ］、参照リストが１の場合はＰＬ１［ｘ，ｙ］となる。ｏｆｆｓｅｔ１及びｓｈｉｆｔ１は丸め処理のパラメータであり、第一予測画像の内部演算精度によって定まる。予測画像のビット精度をＬ、第一予測画像のビット精度をＭとすると、ｓｈｉｆｔ１は数式（５）で定式化され、ｏｆｆｓｅｔ１は数式（６）で定式化される。

[0051] ｓｈｉｆｔ１＝（Ｍ－Ｌ）　…（５）

[0052] ｏｆｆｓｅｔ１＝（１＜＜（ｓｈｉｆｔ１－１）　…（６）

[0053] 例えば、予測画像のビット精度が８であり、第一予測画像のビット精度が１４である場合、数式（５）よりｓｈｉｆｔ１＝６、数式（６）よりｏｆｆｓｅｔ１＝（１＜＜５）となる。

[0054] 重み付き動き補償部３０２は、ＷＰセレクタ３０４、３０５から入力された２つの単方向予測画像（第一予測画像及び第二予測画像）とＷＰパラメータ制御部３０３から入力された重み情報とを元に重み付き動き補償を行う。具体的には、重み付き動き補償部３０２は、第一ＷＰ適用フラグ及び第二ＷＰ適用フラグが１の場合、数式（７）に基づいて重み付き処理を行う。

[0055] Ｐ［ｘ，ｙ］＝Ｃｌｉｐ１（（（ＰＬ０［ｘ，ｙ］＊w_{0C}＋ＰＬ１［ｘ，ｙ］＊w_{1C}＋（１＜＜ｌｏｇＷＤ$_C$））＞＞（ｌｏｇＷＤ$_C$＋１））＋（（o_{0C}＋o_{1C}＋１）＞＞１））　…（７）

[0056] ここで、w_{0C}は第一予測画像に対応する重み係数、w_{1C}は第二予測画像に対応する重み係数、o_{0C}は第一予測画像に対応するオフセット、o_{1C}は第二予測画像に対応するオフセットを表す。以後、それぞれを第一重み係数、第二重み係数、第一オフセット、第二オフセットと呼ぶ。ｌｏｇＷＤ$_C$はそれぞれの重み係数の固定小数点精度を示すパラメータである。変数Ｃは、信号成

分を意味する。例えば、ＹＵＶ空間信号の場合、輝度信号をＣ＝Ｙとし、Ｃｒ色差信号をＣ＝Ｃｒ、Ｃｂ色差成分をＣ＝Ｃｂと表す。

[0057] なお、重み付き動き補償部３０２は、第一予測画像及び第二予測画像と予測画像との演算精度が異なる場合、固定小数点精度である$logWD_C$を数式（８）のように制御することで丸め処理を実現する。

[0058] $logWD'_C = logWD_C + offset1$ …（８）

[0059] 丸め処理は、数式（７）の$logWD_C$を、数式（８）の$logWD'_C$に置き換えることで実現できる。例えば、予測画像のビット精度が８であり、第一予測画像及び第二予測画像のビット精度が１４である場合、$logWD_C$を再設定することにより、数式（１）のｓｈｉｆｔ２と同様の演算精度における一括丸め処理を実現することが可能となる。

[0060] なお、動き情報（予測パラメータ）で示される予測モードが単方向予測である場合、重み付き動き補償部３０２は、第一予測画像のみを用いて、数式（９）に基づいて最終的な予測画像を算出する。

[0061] $P[x, y] = Clip1((PLX[x, y] * w_{XC} + (1 << (logWD_C - 1))) >> (logWD_C))$ …（９）

[0062] ここで、ＰＬＸ［ｘ，ｙ］は単方向予測画像（第一予測画像）を示し、w_{XC}は単方向予測に対応する重み係数を示しており、Ｘは参照リストの０又は１のいずれかを示す識別子である。例えば、参照リストが０の場合はＰＬ０［ｘ，ｙ］、w_{0C}、参照リストが１の場合はＰＬ１［ｘ，ｙ］、w_{1C}となる。

[0063] なお、重み付き動き補償部３０２は、第一予測画像及び第二予測画像と予測画像との演算精度が異なる場合、固定小数点精度である$logWD_C$を双方向予測時と同様に数式（８）のように制御することで丸め処理を実現する。

[0064] 丸め処理は、数式（７）の$logWD_C$を、数式（８）の$logWD'_C$に置き換えることで実現できる。例えば、予測画像のビット精度が８であり、第一予測画像のビット精度が１４である場合、$logWD_C$を再設定することにより、数式（４）のｓｈｉｆｔ１と同様の演算精度における一括丸め処理を実現することが可能となる。

[0065] 図７は、第１実施形態における重み係数の固定小数点精度の一例の説明図であり、時間方向の明度変化がある動画像と階調値との変化の一例を示す図である。図７に示す例では、符号化対象フレームをＦｒａｍｅ（ｔ）とし、時間的に１つ前のフレームをＦｒａｍｅ（ｔ－１）、時間的に１つ後のフレームをＦｒａｍｅ（ｔ＋１）としている。図７に示すように、白から黒に変化するフェード画像では、画像の明度（階調値）が時間とともに減少していく。重み係数は、図７における変化の度合いを意味しており、数式（７）及び数式（９）から明らかなように、明度変化がない場合に１．０の値を取る。固定小数点精度は、重み係数の小数点に対応する刻み幅を制御するパラメータであり、明度変化がない場合の重み係数は、$1 << \mathrm{log}WD_C$となる。

[0066] なお、単方向予測の場合には、第二予測画像に対応する各種パラメータ（第二ＷＰ適応フラグ，第二重み係数，及び第二オフセットの情報）は利用されないため、予め定めた初期値に設定されていてもよい。

[0067] 図１に戻り、動き評価部１０９は、入力画像と予測画像生成部１０７から入力された参照画像とに基づき複数フレーム間の動き評価を行い、動き情報及びＷＰパラメータ情報を出力し、動き情報を予測画像生成部１０７及び符号化部１１０に入力し、ＷＰパラメータ情報を予測画像生成部１０７及びインデックス設定部１０８に入力する。

[0068] 動き評価部１０９は、例えば、予測対象画素ブロックの入力画像と同位置に対応する複数の参照画像を起点として差分値を計算することで誤差を算出し、この位置を分数精度でずらし、誤差最小のブロックを探すブロックマッチングなどの手法により、最適な動き情報を算出する。動き評価部１０９は、双方向予測の場合には、単方向予測で導出された動き情報を用いて、数式（１）及び数式（４）に示すようなデフォルト動き補償予測を含むブロックマッチングを行うことにより、双方向予測の動き情報を算出する。

[0069] この際、動き評価部１０９は、数式（７）及び数式（９）で示されるような重み付き動き補償予測を含むブロックマッチングを行うことにより、ＷＰパラメータ情報を算出できる。なお、ＷＰパラメータ情報の算出には、入力

画像の明度勾配を用いて重み係数やオフセットを算出する方法や、符号化した際の予測誤差の累積による重み係数やオフセットの算出方法などを用いてもよい。またＷＰパラメータ情報は、符号化装置毎に予め定めた固定値を用いてもよい。

[0070] ここで、図７を参照しながら、時間的に明度変化のある動画像から、重み係数、重み係数の固定小数点精度、及びオフセットを算出する方法を説明する。前述したように、図７に示すような白から黒に変化するフェード画像では、画像の明度（階調値）が時間とともに減少していく。動き評価部１０９は、この傾きを計算することにより、重み係数を算出することができる。

[0071] また、重み係数の固定小数点精度は、この傾きの精度を示す情報であり、動き評価部１０９は、参照画像の時間的な距離と画像明度の変化度から、最適な値を計算できる。例えば、図７において、Ｆｒａｍｅ（ｔ－１）～Ｆｒａｍｅ（ｔ＋１）間の重み係数が小数点精度で０．７５である場合、１／４精度であれば、３／４が表現できるため、動き評価部１０９は、固定小数点精度を２（１＜＜２）に設定する。固定小数点精度の値は、重み係数を符号化した場合の符号量に影響を与えるため、符号量と予測精度を考慮して最適な値を選択すればよい。なお、固定小数点精度の値は、予め定めた固定値としてもよい。

[0072] また、動き評価部１０９は、傾きが一致しない場合、一次関数の切片に対応する補正値（ズレ量）を求めることでオフセット値を算出できる。例えば、図７において、Ｆｒａｍｅ（ｔ－１）～Ｆｒａｍｅ（ｔ＋１）間の重み係数が小数点精度で０．６０であり、固定小数点精度が１（１＜＜１）である場合、重み係数は１（つまり、重み係数の小数点精度０．５０に該当）が設定される可能性が高い。この場合、重み係数の小数点精度は、最適な値である０．６０から０．１０ずれているため、動き評価部１０９は、この分の補正値を画素の最大値から計算し、オフセット値として設定する。画素の最大値が２５５である場合、動き評価部１０９は、２５（２５５×０．１）などの値を設定すればよい。

[0073] なお第１実施形態では、符号化装置１００の一機能として動き評価部１０９を例示しているが、動き評価部１０９は符号化装置１００の必須の構成ではなく、例えば、動き評価部１０９を符号化装置１００外の装置としてもよい。この場合、動き評価部１０９で算出された動き情報及びＷＰパラメータ情報を符号化装置１００にロードするようにすればよい。

[0074] インデックス設定部１０８は、動き評価部１０９から入力されたＷＰパラメータ情報を受け取り、参照リスト（リスト番号）と参照画像（参照番号）とを確認して、インデックス情報を出力し、インデックス設定部１０８に入力する。

[0075] 図８は、第１実施形態のインデックス設定部１０８の構成の一例を示すブロック図である。インデックス設定部１０８は、図８に示すように、参照画像確認部４０１と、インデックス生成部４０２とを、備える。

[0076] 参照画像確認部４０１は、動き評価部１０９から入力されたＷＰパラメータ情報を受け取り、２つの参照リストに含まれる参照番号が互いに同一の参照画像を指しているＷＰパラメータがあるかどうかを確認する。そして参照画像確認部４０１は、ＷＰパラメータ情報のうち同一の参照画像を指しているＷＰパラメータを削除し、削除後のＷＰパラメータ情報を出力し、インデックス生成部４０２に入力する。

[0077] インデックス生成部４０２は、参照画像確認部４０１から冗長なＷＰパラメータが削除されたＷＰパラメータ情報を受け取り、後述するシンタクス要素にマッピングを行ってインデックス情報を生成する。インデックス生成部４０２は、インデックス情報を出力し、符号化部１１０に入力する。

[0078] 図９Ａ及び図９Ｂは、第１実施形態のＷＰパラメータ情報の一例を示す図である。Ｐ－ｓｌｉｃｅ時のＷＰパラメータ情報の一例は、図９Ａに示すとおりであり、Ｂ－ｓｌｉｃｅ時のＷＰパラメータ情報の一例は、図９Ａ及び図９Ｂに示すとおりである。リスト番号は、予測方向を示す識別子であり、単方向予測時は０の値を取り、双方向予測時は２種類の予測を用いることができるため、０と１の２つの値を取る。参照番号は、フレームメモリ２０６

に示される１～Ｎに対応する値である。ＷＰパラメータ情報は、参照リストと参照画像毎に保持されるため、Ｂ－ｓｌｉｃｅ時で必要な情報は、参照画像がＮ個とすると２Ｎ個となる。参照画像確認部４０１は、これらのＷＰパラメータ情報を再変換し、冗長なＷＰパラメータを削除する。

[0079] 図１０は、第１実施形態の低遅延符号化構造における符号化順と表示順（ＰＯＣ：Ｐｃｉｔｕｒｅ　Ｏｒｄｅｒ　Ｃｏｕｎｔ）の一例を示す図であり、参照画像となることが可能なＢ－ｓｌｉｃｅをｒＢ－ｓｌｉｃｅとして用いた符号化構造の例を示している。低遅延符号化構造では、画像の先読みが出来ないため、符号化順と表示順が同一となる。ここでは、表示順でフレーム０～３までが符号化済みとし、フレーム４を符号化することを考える。

[0080] 図１１は、第１実施形態の低遅延符号化構造における参照画像と参照番号との関係の一例を示す図であり、フレーム４を符号化する場合の参照画像と参照番号との関係を簡略化したＷＰパラメータ情報で示している。図１１に示す例では、参照リスト内の参照番号の順序がリスト０とリスト１とで同一となっており、リスト０とリスト１との参照画像が同一であることを意味している。

[0081] 図１２は、第１実施形態のランダムアクセス符号化構造における符号化順と表示順の一例を示す図であり、ここでは、表示順でフレーム０、２、４、８が符号化済みであり、フレーム１を符号化することを考える。

[0082] 図１３は、第１実施形態のランダムアクセス符号化構造における参照画像と参照番号との関係の一例を示す図であり、フレーム１を符号化する場合の参照画像と参照番号との関係を簡略化したＷＰパラメータ情報で示している。図１３に示す例では、参照リスト別の参照番号の順序はそれぞれ異なっているが、リスト０とリスト１とには、共通の４つの参照画像が含まれており、リスト０とリスト１との参照画像が同一であることを意味している。

[0083] 参照画像確認部４０１は、冗長なＷＰパラメータ、即ち、２つの参照リストに含まれる参照番号が互いに同一の参照画像を指しているＷＰパラメータを削除するため、参照番号を並び替え、共通の新しいインデックスに変換す

る。

[0084] 図１４は、第１実施形態の参照画像のリスト番号及び参照番号のスキャン順序の一例を示す図である。参照画像確認部４０１は、図１４に示すスキャン順で、２次元の参照リスト（２つの参照リスト）を１次元の共通リスト（１つの共通リスト）に変換する。具体的には、参照画像確認部４０１は、図１４に示すスキャン順で、図９Ａ及び図９Ｂに示すＷＰパラメータ情報を読み込むことにより、２つの参照リストを共通リストに並び替え、冗長なＷＰパラメータを削除する。ここでは、参照画像確認部４０１は、数式（１０）に示す疑似コードに従って、ＷＰパラメータ情報を読み込む。

[0085]
```
for (scan=0; scan<=num_of_common_active_ref_minus1; scan++){
list = common_scan_list(scan)
ref_idx = common_scan_ref_idx(scan)
refPOC = RefPicOrderCount(list, ref_idx)
identical_ref_flag=false;
for (curridx=0; curridx<=num_of_common_active_ref_minus1; curridx++){
if( refPOC == CommonRefPicOrderCount(curridx) ){
identical_ref_flag=true;
}
}
if(!identical_ref_flag)
InsertCommonRefPicOrderCount(scan, refPOC);
}
```
… (１０)

[0086] ここで、common_scan_list()は、図１４のスキャン番号に対応するリスト番号を返す関数である。common_scan_ref_idx()は、図１４のスキャン番号に対応する参照番号を返す関数である。RefPicOrderCnt()は、リスト番号及び参照番号に対応するＰＯＣ番号を返す関数である。ＰＯＣ番号は、参照画像の表示順を示す番号である。但し、ＰＯＣ番号は、予め定められた最大数を

超える場合、初期値にマッピングされる。例えば、ＰＯＣ番号の最大値が２５５である場合、２５６に対応するＰＯＣ番号は０となる。CommonRefPicOrderCount()は、共通リスト（CommonList）の参照番号に対応するＰＯＣ番号を返す関数である。InsertCommonRefPicOrderCount()は、スキャン番号に対応するＰＯＣ番号を、共通リスト（CommonList）に挿入する関数である。

[0087] なお、図１４に示すスキャン順序は一例であり、予め定めたスキャン順序であればこれ以外のスキャン順序を用いてもよい。また、数式（１０）で示した疑似コードは一例であり、本処理の目的を実現できれば処理の追加や冗長な処理の削減も可能である。

[0088] 図１５及び図１６は、第１実施形態における共通リスト変換後のＷＰパラメータ情報の一例を簡略化して示した図である。図１５は、図１１に示す参照リストのＷＰパラメータ情報を変換した共通リストのＷＰパラメータ情報を簡略化して示しており、図１６は、図１３に示す参照リストのＷＰパラメータ情報を変換した共通リストのＷＰパラメータ情報を簡略化して示している。

[0089] 図１７は、第１実施形態のインデックス設定部１０８で実行されるインデックス情報の生成処理の一例を示すフローチャートである。

[0090] 参照画像確認部４０１は、ＷＰパラメータ情報が入力されると、スライスタイプに従って処理を分岐する（ステップＳ１０１）。

[0091] スライスタイプが１つの参照リストのみを用いる単方向予測スライス（Ｐ－ｓｌｉｃｅ）である場合には（ステップＳ１０１でＮｏ）、参照画像確認部４０１は、図９Ａに示すＷＰパラメータ情報をそのままインデックス生成部４０２へ出力する。インデックス生成部４０２は、参照画像確認部４０１から入力されたＷＰパラメータ情報を後述する予め定められたシンタクス要素にマッピングしてインデックス情報を生成し、出力する。

[0092] 一方、スライスタイプが２つの参照リストを用いる双方向予測スライス（Ｂ－ｓｌｉｃｅ）である場合には（ステップＳ１０１でＹｅｓ）、参照画像確認部４０１は、変数scanを０に初期化する（ステップＳ１０２）。変数sca

nは、図１４のスキャン番号を示す。

[0093] 続いて、参照画像確認部４０１は、変数scanを用いて、スキャン番号に対応するリスト番号を示す変数listを取得するとともに（ステップＳ１０３）、スキャン番号に対応する参照番号を示す変数refIdxを取得する（ステップＳ１０４）。

[0094] 続いて、参照画像確認部４０１は、変数listが示すリスト番号及び変数refIdxが示す参照番号に対応する参照画像のＰＯＣ番号を示す変数refPOCを導出する（ステップＳ１０５）。

[0095] 続いて、参照画像確認部４０１は、フラグidentical_ref_flagをfalseに設定するとともに（ステップＳ１０６）、変数curridxを０に設定する（ステップＳ１０７）。フラグidentical_ref_flagは、共通リストに同一の参照画像があるか否かを示す。変数curridxは、共通リストの参照番号を示す。

[0096] 続いて、参照画像確認部４０１は、導出した変数refPOCが示すＰＯＣ番号と変数curridxが示す参照番号に対応する参照画像のＰＯＣ番号とが同一であるかどうかを判断する（ステップＳ１０８）。

[0097] 両ＰＯＣ番号が同一である場合（ステップＳ１０８でＹｅｓ）、参照画像確認部４０１は、フラグidentical_ref_flagをtrueに設定する（ステップＳ１０９）。一方、両ＰＯＣ番号が同一でない場合（ステップＳ１０８でＮｏ）、ステップＳ１０９の処理は行われない。

[0098] 続いて、参照画像確認部４０１は、変数curridxをインクリメントする（ステップＳ１１０）。

[0099] 続いて、参照画像確認部４０１は、変数curridxの値が共通リストの最大数から１を引いた値であるnum_of_common_active_ref_minus1より大きいかどうかを判断し（ステップＳ１１１）、num_of_common_active_ref_minus1以下の間（ステップＳ１１１でＮｏ）、ステップＳ１０８～ステップＳ１１０の処理を繰り返す。

[0100] num_of_common_active_ref_minus1より大きくなると（ステップＳ１１１でＹｅｓ）、参照画像確認部４０１は、共通リストの確認を完了し、フラグide

ntical_ref_flagがfalseかどうかを更に確認する（ステップＳ１１２）。

[0101] フラグidentical_ref_flagがfalseの場合（ステップＳ１１２でＹｅｓ）、参照画像確認部４０１は、共通リストに変数refPOCが示すＰＯＣ番号の参照画像と同一の参照画像が含まれないと判断し、変数refPOCが示すＰＯＣ番号を共通リストに追加する（ステップＳ１１３）。一方、フラグidentical_ref_flagがtrueの場合（ステップＳ１１２でＮｏ）、参照画像確認部４０１は、共通リストに変数refPOCが示すＰＯＣ番号の参照画像と同一の参照画像が含まれると判断し、ステップＳ１１３の処理は行わない。

[0102] 続いて、参照画像確認部４０１は、変数scanをインクリメントする（ステップＳ１１４）。

[0103] 続いて、参照画像確認部４０１は、変数scanの値が共通リストの最大数から１を引いた値であるnum_of_common_active_ref_minus1より大きいかどうかを判断し（ステップＳ１１５）、num_of_common_active_ref_minus1以下であれば（ステップＳ１１５でＮｏ）、ステップＳ１０３に戻る。一方、num_of_common_active_ref_minus1より大きければ（ステップＳ１１５でＹｅｓ）、参照画像確認部４０１は、共通リストに変換後のＷＰパラメータ情報をインデックス生成部４０２へ出力する。インデックス生成部４０２は、参照画像確認部４０１から入力されたＷＰパラメータ情報を後述する予め定められたシンタクス要素にマッピングしてインデックス情報を生成し、出力する。

[0104] 図１に戻り、符号化部１１０は、量子化部１０３から入力された量子化変換係数、動き評価部１０９から入力された動き情報、インデックス設定部１０８から入力されたインデックス情報、及び符号化制御部１１１によって指定される量子化情報などの様々な符号化パラメータに対してエントロピー符号化を行い、符号化データを生成する。エントロピー符号化は、例えば、ハフマン符号化や算術符号化などが該当する。

[0105] 符号化パラメータとは、予測方法などを示す予測情報、量子化変換係数に関する情報、及び量子化に関する情報などの復号に必要となるパラメータである。例えば、符号化制御部１１１が図示せぬ内部メモリを持ち、この内部

メモリに符号化パラメータが保持され、画素ブロックを符号化する際に隣接する既に符号化済みの画素ブロックの符号化パラメータを用いるようにできる。例えば、Ｈ．２６４のイントラ予測では、符号化済みの隣接ブロックの予測情報から、画素ブロックの予測情報を導出することができる。

[0106] 符号化部１１０は、生成した符号化データを、符号化制御部１１１が管理する適切な出力タイミングに従って出力する。出力された符号化データは、例えば、図示せぬ多重化部などで様々な情報が多重化されて、図示せぬ出力バッファなどに一時的に蓄積された後に、例えば、図示せぬ蓄積系（蓄積メディア）又は伝送系（通信回線）へ出力される。

[0107] 図１８は、第１実施形態の符号化装置１００が利用するシンタクス５００の一例を示す図である。シンタクス５００は、符号化装置１００が入力画像（動画像データ）を符号化して生成した符号化データの構造を示している。符号化データを復号化する場合、後述の復号装置は、シンタクス５００と同一のシンタクス構造を参照して動画像のシンタクス解釈を行う。

[0108] シンタクス５００は、ハイレベルシンタクス５０１、スライスレベルシンタクス５０２及びコーディングツリーレベルシンタクス５０３の３つのパートを含む。ハイレベルシンタクス５０１は、スライスよりも上位のレイヤのシンタクス情報を含む。スライスとは、フレーム若しくはフィールドに含まれる矩形領域又は連続領域を指す。スライスレベルシンタクス５０２は、各スライスを復号化するために必要な情報を含む。コーディングツリーレベルシンタクス５０３は、各コーディングツリー（即ち、各コーディングツリーブロック）を復号するために必要な情報を含む。これら各パートは、更に詳細なシンタクスを含む。

[0109] ハイレベルシンタクス５０１は、シーケンスパラメータセットシンタクス５０４、ピクチャパラメータセットシンタクス５０５、及びアダプテーションパラメータセットシンタクス５０６などのシーケンス及びピクチャレベルのシンタクスを含む。

[0110] スライスレベルシンタクス５０２は、スライスヘッダーシンタクス５０７

、プレッドウェイトテーブルシンタクス５０８、及びスライスデータシンタクス５０９などを含む。プレッドウェイトテーブルシンタクス５０８は、スライスヘッダーシンタクス５０７から呼び出される。

[0111] コーディングツリーレベルシンタクス５０３は、コーディングツリーユニットシンタクス５１０、トランスフォームユニットシンタクス５１１、及びプレディクションユニットシンタクス５１２などを含む。コーディングツリーユニットシンタクス５１０は、四分木構造を持つことができる。具体的には、コーディングツリーユニットシンタクス５１０のシンタクス要素として、更にコーディングツリーユニットシンタクス５１０を再帰呼び出しすることができる。即ち、１つのコーディングツリーブロックを四分木で細分化することができる。また、コーディングツリーユニットシンタクス５１０内にはトランスフォームユニットシンタクス５１１が含まれている。トランスフォームユニットシンタクス５１１は、四分木の最末端の各コーディングツリーユニットシンタクス５１０において呼び出される。トランスフォームユニットシンタクス５１１は、逆直交変換及び量子化などに関わる情報が記述されている。これらのシンタクスには、重み付き動き補償予測に関する情報が記述されてもよい。

[0112] 図１９は、第１実施形態のピクチャパラメータセットシンタクス５０５の一例を示す図である。weighted_pred_flagは、例えば、Ｐ－ｓｌｉｃｅに関する第１実施形態の重み付き補償予測の有効又は無効を示すシンタクス要素である。weighted_pred_flagが０である場合、Ｐ－ｓｌｉｃｅ内での第１実施形態の重み付き動き補償予測は無効となる。従って、ＷＰパラメータ情報に含まれるＷＰ適用フラグは常に０に設定され、ＷＰセレクタ３０４、３０５は、各々の出力端をデフォルト動き補償部３０１へ接続する。一方、weighted_pred_flagが１である場合、Ｐ－ｓｌｉｃｅ内での第１実施形態の重み付き動き補償予測は有効となる。

[0113] なお、別の例として、weighted_pred_flagが１である場合には、より下位のレイヤ（スライスヘッダー、コーディングツリーブロック、トランスフォ

ームユニット、及びプレディクションユニットなど）のシンタクスにおいて、スライス内部の局所領域毎に第１実施形態の重み付き動き補償予測の有効又は無効を規定するようにしてもよい。

[0114] weighted_bipred_idcは、例えば、Ｂ－ｓｌｉｃｅに関する第１実施形態の重み付き補償予測の有効又は無効を示すシンタクス要素である。weighted_bipred_idcが０である場合、Ｂ－ｓｌｉｃｅ内での第１実施形態の重み付き動き補償予測は無効となる。従って、ＷＰパラメータ情報に含まれるＷＰ適用フラグは常に０に設定され、ＷＰセレクタ３０４、３０５は、各々の出力端をデフォルト動き補償部３０１へ接続する。一方、weighted_bipred_idcが１である場合、Ｂ－ｓｌｉｃｅ内での第１実施形態の重み付き動き補償予測は有効となる。

[0115] なお、別の例として、weighted_bipred_idcが１である場合には、より下位のレイヤ（スライスヘッダー、コーディングツリーブロック、及びトランスフォームユニットなど）のシンタクスにおいて、スライス内部の局所領域毎に第１実施形態の重み付き動き補償予測の有効又は無効を規定するようにしてもよい。

[0116] 図２０は、第１実施形態のスライスヘッダーシンタクス５０７の一例を示す図である。slice_typeはスライスのスライスタイプ（Ｉ－ｓｌｉｃｅ，Ｐ－ｓｌｉｃｅ，Ｂ－ｓｌｉｃｅなど）を示している。pic_parameter_set_idは、いずれのピクチャパラメータセットシンタクス５０５を参照するかを示す識別子である。num_ref_idx_active_override_flagは、有効な参照画像の数を更新するかどうかを示すフラグであり、本フラグが１の場合、参照リストの参照画像数を定義するnum_ref_idx_l0_active_minus1及びnum_ref_idx_l1_active_minus1が利用できる。pred_weight_table()は、重み付き動き補償予測に利用するプレッドウェイトテーブルシンタクスを示す関数であり、前述のweighted_pred_flagが１かつＰ－ｓｌｉｃｅの場合、及びweighted_bipred_idcが１かつＢ－ｓｌｉｃｅの場合に、本関数が呼び出される。

[0117] 図２１は、第１実施形態のプレッドウェイトテーブルシンタクス５０８の

一例を示す図である。luma_log2_weight_denomは、スライスにおける輝度信号の重み係数の固定小数点精度を表しており、数式（７）又は数式（９）のｌｏｇＷＤ$_C$に対応する値である。chroma_log2_weight_denomは、スライスにおける色差信号の重み係数の固定小数点精度を表しており、数式（７）又は数式（９）のｌｏｇＷＤ$_C$に対応する値である。chroma_format_idcは、色空間を表す識別子であり、MONO_IDXはモノクロ映像を示す値である。num_ref_common_active_minus1は、スライスにおける共通リストに含まれる参照画像の数から１を引いた値を示し、数式（１０）で用いられる。この値の最大値は、リスト０とリスト１の２つの参照画像枚数の最大値を足した値となる。

[0118] luma_weight_common_flagは、輝度信号におけるＷＰ適応フラグを示している。本フラグが１の場合、スライス内全域で第１実施形態の輝度信号の重み付き動き補償予測が有効となる。chroma_weight_common_flagは、色差信号におけるＷＰ適応フラグを示している。本フラグが１の場合、スライス内全域で第１実施形態の色差信号の重み付き動き補償予測が有効となる。luma_weight_common[i]は、共通リストで管理されたi番目に対応する輝度信号の重み係数である。luma_offset_common[i]は、共通リストで管理されたi番目に対応する輝度信号のオフセットである。これらは、それぞれ、数式（７）又は数式（９）のw_{0C}、w_{1C}、o_{0C}、o_{1C}に対応する値である。但し、Ｃ＝Ｙとする。

[0119] chroma_weight_common[i][j]は、共通リストで管理されたi番目に対応する色差信号の重み係数である。chroma_offset_common[i][j]は、共通リストで管理されたi番目に対応する色差信号のオフセットである。これらは、それぞれ、数式（７）又は数式（９）のw_{0C}、w_{1C}、o_{0C}、o_{1C}に対応する値である。但し、Ｃ＝Ｃｒ又はＣｂとする。ｊは色差成分のコンポーネントを示しており、例えばＹＵＶ４：２：０信号の場合、j=0がＣｒ成分、j=1がＣｂ成分であることを示す。

[0120] 以上のように第１実施形態では、インデックス設定部１０８は、ＷＰパラメータ情報において、リスト０及びリスト１の２つの参照リスト内に互いに

同一の参照画像を含む組み合わせがある場合、２つの参照リストを共通リストへ変換することによって、ＷＰパラメータ情報から重複する冗長なＷＰパラメータを削除してインデックス情報を生成する。従って、第１実施形態によれば、インデックス情報の符号量を削減することができる。

[0121] 特に、重み係数及びオフセットは、参照画像毎に符号化する必要があるため、参照画像枚数の増加に従ってデコーダにシグナリングする情報量も増大するが、参照画像枚数が増加すると、参照リスト内に同一の参照画像を参照するケースが増えるため、共通リストで管理することにより大幅な符号量の削減効果が期待できる。

[0122] （第１実施形態の変形例）

第１実施形態の変形例について説明する。第１実施形態の変形例では、インデックス生成部４０２が用いるシンタクス要素が第１実施形態のシンタクス要素と異なる。

[0123] 図２２は、第１実施形態の変形例のシーケンスパラメータセットシンタクス５０４の一例を示す図である。profile_idcは、符号化データのプロファイルに関する情報を示す識別子である。level_idcは、符号化データのレベルに関する情報を示す識別子である。seq_parameter_set_idは、いずれのシーケンスパラメータセットシンタクス５０４を参照するかを示す識別子である。num_ref_framesは、フレームにおける参照画像の最大枚数を示す変数である。weighted_prediction_enabled_flagは、例えば、符号化データに関して変形例の重み付き動き補償予測の有効又は無効を示すシンタクス要素である。

[0124] weighted_prediction_enabled_flagが０である場合、符号化データ内での変形例の重み付き動き補償予測は無効となる。従って、ＷＰパラメータ情報に含まれるＷＰ適用フラグは常に０に設定され、ＷＰセレクタ３０４、３０５は、各々の出力端をデフォルト動き補償部３０１へ接続する。一方、weighted_prediction_enabled_flagが１である場合、符号化データ内全域で変形例の重み付き動き補償予測が有効となる。

[0125] なお、別の例として、weighted_prediction_enabled_flagが１である場合

には、より下位のレイヤ（ピクチャパラメータセット、アダプテーションパラメータセット、スライスヘッダー、コーディングツリーブロック、トランスフォームユニット、及びプレディクションユニットなど）のシンタクスにおいて、スライス内部の局所領域毎に変形例の重み付き動き補償予測の有効又は無効を規定するようにしてもよい。

[0126] 図２３は、第１実施形態の変形例のアダプテーションパラメータセットシンタクス５０６の一例を示す図である。アダプテーションパラメータセットシンタクス５０６は、符号化フレーム全体に影響を与えるパラメータを保持しており、上位ユニットとして独立に符号化される。例えば、Ｈ．２６４では、ハイレベルシンタクス５０１に該当するシンタクス要素をＮＡＬユニットとして符号化する。このため、アダプテーションパラメータセットシンタクス５０６は、スライスや画素ブロックに代表される下位情報のパラメータを保持するスライスレベルシンタクス５０２より下位のシンタクスと符号化ユニットが異なる。

[0127] aps_idは、いずれのアダプテーションパラメータセットシンタクス５０６を参照するかを示す識別子である。本識別子を参照することにより、下位のスライスは、符号化済みの任意のaps_idを参照することができる。この場合、例えば、図２０のスライスヘッダーシンタクス５０７にも同様のaps_idを付加することによって、フレーム全体に影響を与えるパラメータを読み込むことが可能となる。

[0128] aps_weighted_prediction_flagは、例えば、フレームにおけるＰ－ｓｌｉｃｅに関する変形例の重み付き動き補償予測の有効又は無効を示すシンタクス要素である。aps_weighted_prediction_flagが０である場合、フレーム内におけるＰ－ｓｌｉｃｅでの変形例の重み付き動き補償予測は無効となる。従って、ＷＰパラメータ情報に含まれるＷＰ適用フラグは常に０に設定され、ＷＰセレクタ３０４、３０５は、各々の出力端をデフォルト動き補償部３０１へ接続する。一方、aps_weighted_prediction_flagが１である場合、当該フレーム内全域で変形例の重み付き動き補償予測が有効となる。

[0129] なお、別の例として、aps_weighted_prediction_flagが１である場合には、より下位のレイヤ（スライスヘッダー、コーディングツリーブロック、トランスフォームユニット、及びプレディクションユニットなど）のシンタクスにおいて、スライス内部の局所領域毎に変形例の予測の重み付き動き補償予測の有効又は無効を規定するようにしてもよい。

[0130] aps_weighted_bipred_idxは、例えば、フレームにおけるＢ－ｓｌｉｃｅに関する変形例の重み付き動き補償予測の有効又は無効を示すシンタクス要素である。aps_weighted_bipred_idxが０である場合、フレーム内におけるＰ－ｓｌｉｃｅでの変形例の重み付き動き補償予測は無効となる。従って、ＷＰパラメータ情報に含まれるＷＰ適用フラグは常に０に設定され、ＷＰセレクタ３０４、３０５は、各々の出力端をデフォルト動き補償部３０１へ接続する。一方、aps_weighted_bipred_idxが１である場合、当該フレーム内全域で変形例の重み付き動き補償予測が有効となる。

[0131] なお、別の例として、aps_weighted_bipred_idxが１である場合には、より下位のレイヤ（スライスヘッダー、コーディングツリーブロック、トランスフォームユニット、及びプレディクションユニットなど）のシンタクスにおいて、スライス内部の局所領域毎に変形例の予測の重み付き動き補償予測の有効又は無効を規定するようにしてもよい。

[0132] pred_weight_table()は、重み付き動き補償予測に利用するプレッドウェイトテーブルシンタクスを示す関数であり、前述のaps_weighted_prediction_flagが１の場合、又はaps_weighted_bipred_idxが１の場合に、本関数が呼び出される。

[0133] 図２４は、第１実施形態の変形例のプレッドウェイトテーブルシンタクス５０８の一例を示す図である。第１実施形態の変形例では、図１８のシンタクス構造において、アダプテーションパラメータセットシンタクス５０６からプレッドウェイトテーブルシンタクス５０８が呼び出される。図２１のプレッドウェイトテーブルシンタクス５０８との違いは、num_ref_common_active_minus1が、MAX_COMMON_REF_MINUS1に変更されている点である。参照画像

の枚数は、スライスヘッダーシンタクス５０７で記述されているため、上位レイヤであるアダプテーションパラメータセットシンタクス５０６では参照することが出来ない。このため、例えばシーケンスパラメータセットシンタクス５０４に記述されているnum_ref_framesの値から１を引いた値をMAX_COMMON_REF_MINUS1にセットする。また、予め定めたプロファイル又はレベルに応じて最大の参照画像枚数を取れる値を定めておいてもよい。なお、他のシンタクス要素は図２１と同一である。

[0134] 以上のように第１実施形態の変形例によれば、プレッドウェイトテーブルシンタクス５０８をアダプテーションパラメータセットシンタクス５０６から呼び出す構造とすることにより、１フレームを複数のスライスに分割した際のＷＰパラメータ情報の符号量を大幅に削減することができる。

[0135] 例えば、利用する３種類の異なるＷＰパラメータ情報を持つアダプテーションパラメータセットシンタクス５０６を先に符号化しておき、スライスヘッダーシンタクス５０７が、状況に応じてaps_idを用いて、必要なＷＰパラメータ情報を呼び出すことにより、スライスヘッダーシンタクス５０７でＷＰパラメータ情報を常に符号化する構成よりも、符号量を削減できる。

[0136] （第２実施形態）

第２実施形態について説明する。第２実施形態の符号化装置６００では、インデックス設定部６０８の構成が第１実施形態の符号化装置１００と相違する。以下では、第１実施形態との相違点の説明を主に行い、第１実施形態と同様の機能を有する構成要素については、第１実施形態と同様の名称・符号を付し、その説明を省略する。

[0137] 図２５は、第２実施形態のインデックス設定部６０８の構成の一例を示すブロック図である。インデックス設定部６０８は、図２５に示すように、再利用判定部６０１と、インデックス生成部６０２とを、備える。

[0138] 再利用判定部６０１は、リスト１の参照番号をリスト０の参照番号と照合し、リスト０のＷＰパラメータ情報をリスト１のＷＰパラメータ情報として再利用するかどうかを判定する。２つの参照リスト間で同一の参照画像が含

まれており、かつ対応するＷＰパラメータ情報の値が同一である場合、リスト０だけでなく、リスト１でも同じ情報を符号化すると符号量の増大を招く。このため、再利用判定部６０１は、リスト０のＷＰパラメータ情報を再利用する。

[0139] 具体的には、再利用判定部６０１は、リスト０のＷＰパラメータ情報を再利用する場合は再利用フラグを１とし、参照先（リスト０）の参照番号を参照情報として設定する。一方、再利用判定部６０１は、リスト０のＷＰパラメータ情報を再利用しない場合は再利用フラグを０とし、リスト１の参照番号に対応するＷＰパラメータ情報をシンタクスにセットする。

[0140] 再利用判定部６０１は、例えば、数式（１１）に示す疑似コードに従って、リスト０のＷＰパラメータ情報をリスト１のＷＰパラメータ情報として再利用するかどうかを判定する。

[0141]
```
for (refIdx=0; refidx<=num_of_active_ref_l1_minus1; refIdx++){
refPOC = RefPicOrderCnt(ListL1, refIdx)
refWP = RefWPTable(ListL1, refIdx)
reuse_wp_flag=false;
reuse_ref_idx=0;
for (curridx=0; curridx<= num_of_active_ref_l0_minus1; curridx++)
{
if( refPOC == RefPicOrderCnt(ListL0, curridx)
&& RefWPTable(ListL0, curridx, refWP) ){
reuse_wp_flag=true;
reuse_ref_idx=curridx;
}
}
}
```
… (１１)

[0142] ここで、ListL0は、リスト０を示し、ListL1はリスト１を示す。RefWPTable()は、リスト番号、参照番号、及び参照用ＷＰパラメータを入力すると、入

力された参照用ＷＰパラメータが入力されたリスト番号及び参照番号に対応するＷＰパラメータと一致しているかどうかをフラグで返す関数である。フラグの値が１の場合は、両ＷＰパラメータが一致していることを示す。数式（１１）に示す疑似コードでは、リスト１の参照番号を順にスキャンし、リスト０の中に同一のＰＯＣ番号の参照画像が存在した場合に、ＷＰパラメータが同一であれば、再利用フラグを示すreuse_wp_flagをtureに設定するとともに、リスト０の対応する参照番号をreuse_ref_idxに設定する。

[0143] 再利用判定部６０１は、再利用判定後の（再利用の設定を行った）ＷＰパラメータ情報を出力し、インデックス生成部６０２に入力する。

[0144] インデックス生成部６０２は、再利用判定部６０１から再利用判定後のＷＰパラメータ情報を受け取り、後述するシンタクス要素にマッピングを行ってインデックス情報を生成する。なお、インデックス生成部６０２がマッピングを行うシンタクス要素は、第１実施形態と異なる。インデックス生成部６０２は、インデックス情報を出力し、符号化部１１０に入力する。

[0145] 図２６は、第２実施形態のインデックス設定部６０８で実行されるインデックス情報の生成処理の一例を示すフローチャートである。

[0146] 再利用判定部６０１は、ＷＰパラメータ情報が入力されると、スライスタイプに従って処理を分岐する（ステップＳ２０１）。

[0147] スライスタイプが１つの参照リストのみを用いる単方向予測スライス（Ｐ－ｓｌｉｃｅ）である場合には（ステップＳ２０１でＮｏ）、再利用判定部６０１は、図９Ａに示すＷＰパラメータ情報をそのままインデックス生成部６０２へ出力する。インデックス生成部６０２は、再利用判定部６０１から入力されたＷＰパラメータ情報を後述する予め定められたシンタクス要素にマッピングしてインデックス情報を生成し、出力する。

[0148] 一方、スライスタイプが２つの参照リストを用いる双方向予測スライス（Ｂ－ｓｌｉｃｅ）である場合には（ステップＳ２０１でＹｅｓ）、再利用判定部６０１は、変数refIdcを０に初期化する（ステップＳ２０２）。変数refIdcは、リスト１の参照番号を示す。

[0149] 続いて、再利用判定部６０１は、変数refIdxが示す参照番号に対応する参照画像のＰＯＣ番号を示す変数refPOCを導出するとともに（ステップＳ２０３）、変数refIdxが示す参照番号に対応するＷＰパラメータを導出する（ステップＳ２０４）。

[0150] 続いて、再利用判定部６０１は、フラグreuse_wp_flagをfalseに設定するとともに（ステップＳ２０５）、変数curridxを０に設定する（ステップＳ２０６）。フラグreuse_wp_flagは、リスト０とリスト１のＷＰパラメータが同一であるか否かを示す。変数curridxは、リスト０の参照番号を示す。

[0151] 続いて、再利用判定部６０１は、導出した変数refPOCが示すＰＯＣ番号と変数curridxが示す参照番号に対応する参照画像のＰＯＣ番号とが同一であり、かつ変数refIdxが示す参照番号に対応するＷＰパラメータと変数curridxが示す参照番号に対応するＷＰパラメータとが同一であるかを判断する（ステップＳ２０７）。

[0152] 両ＰＯＣ番号及び両ＷＰパラメータが同一である場合（ステップＳ２０７でＹｅｓ）、再利用判定部６０１は、フラグreuse_wp_flagをtrueに設定するとともに（ステップＳ２０８）、変数reuse_ref_idxに変数curridxの値を代入する（ステップＳ２０９）。一方、両ＰＯＣ番号又は両ＷＰパラメータが同一でない場合（ステップＳ２０７でＮｏ）、ステップＳ２０８、Ｓ２０９の処理は行われない。なお、ＰＯＣ番号の一致を確認しない構成としてもよい。

[0153] 続いて、再利用判定部６０１は、変数curridxをインクリメントする（ステップＳ２１０）。

[0154] 続いて、再利用判定部６０１は、変数curridxの値がリスト０の最大数から１を引いた値であるnum_of_active_ref_l0_minus1より大きいかどうかを判断し（ステップＳ２１１）、num_of_active_ref_l0_minus1以下の間（ステップＳ２１１でＮｏ）、ステップＳ２０７～ステップＳ２１０の処理を繰り返す。

[0155] num_of_active_ref_l0_minus1より大きくなると（ステップＳ２１１でＹｅ

ｓ）、再利用判定部６０１は、リスト０の確認を完了し、変数refIdxをインクリメントする（ステップＳ２１２）。

[0156] 続いて、再利用判定部６０１は、変数refIdxの値がリスト１の最大数から１を引いた値であるnum_of_active_ref_l1_minus1より大きいかどうかを判断し（ステップＳ２１３）、num_of_active_ref_l1_minus1以下の間（ステップＳ２１３でＮｏ）、ステップＳ２０３～ステップＳ２１２の処理を繰り返す。

[0157] num_of_active_ref_l1_minus1より大きくとなると（ステップＳ２１３でＹｅｓ）、再利用判定部６０１は、リスト１の確認を完了し、再利用判定後のＷＰパラメータ情報をインデックス生成部６０２へ出力する。インデックス生成部６０２は、再利用判定部６０１から入力されたＷＰパラメータ情報を後述する予め定められたシンタクス要素にマッピングしてインデックス情報を生成し、出力する。

[0158] 図２７は、第２実施形態のプレッドウェイトテーブルシンタクス５０６の一例を示す図である。図２７に示すシンタクス要素のうち、l0、l1の記号があるものは、それぞれ、リスト０、１に対応する。また、接頭語が図２１と同じシンタクス要素は、共通リストではなくリスト０又はリスト１の違いがあるが、参照リストの扱いが異なる以外は同一の変数として利用する。例えば、luma_weight_l0[i]は、参照リストl0における参照番号i番目の重み係数を表すシンタクス要素である。

[0159] luma_log2_weight_denom及びchroma_log2_weight_denomは、第１実施形態と同じ構成となっている。まず、l0の記号のあるリスト０に対して、luma_weight_l0_flag、luma_weight_l0[i]、luma_offset_l0[i]、chroma_weight_l0_flag、chroma_weight_l0[i][j]、及びchroma_offset_l0[i][j]が定義されている。次に、l1の記号のあるリスト１に関するシンタクス要素が定義されている。

[0160] ここで、reuse_wp_flagは、リスト０に対応するl0のＷＰパラメータを再利用するかどうかを示すシンタクス要素である。reuse_wp_flagが１の場合、重

み係数及びオフセットは符号化されずに、reuse_ref_idxが符号化される。reuse_ref_idxは、リスト０に対応するl0のどの参照番号に対応するＷＰパラメータを利用するかを示す。例えば、reuse_ref_idxが１である場合、リスト０の参照番号１に対応するＷＰパラメータがリスト１の参照番号iに対応するＷＰパラメータにコピーされる。reuse_wp_flagが０の場合、リスト０と同様、luma_weight_l1_flag、luma_weight_l1[i]、luma_offset_l1[i]、chroma_weight_l1_flag、chroma_weight_l1[i][j]、及びchroma_offset_l1[i][j]が符号化される。

[0161] 以上のように第２実施形態では、インデックス設定部６０８は、ＷＰパラメータ情報において、リスト０及びリスト１の２つの参照リスト内に互いに同一の参照画像を含む組み合わせがある場合、重複するＷＰパラメータを再利用することにより、ＷＰパラメータ情報から重複する冗長なＷＰパラメータを削除してインデックス情報を生成する。従って、第２実施形態によれば、インデックス情報の符号量を削減することができる。

[0162] 第２実施形態では、リスト０に対応するＷＰパラメータ情報は従来通り符号化し、リスト１に対応するＷＰパラメータ情報を符号化する際には、リスト０内で同じ参照画像を参照するかどうかをチェックし、同じ参照画像を参照し、ＷＰパラメータも同一である場合には、リスト０のＷＰパラメータを再利用するフラグを符号化するとともに、リスト０のどの参照番号に対応するかを示す情報を符号化する。これらの情報は、重み係数及びオフセットに関する情報量よりも格段に少ない情報であるため、リスト０及びリスト１を別々に符号化する場合と比較して、ＷＰパラメータ情報の符号量を大幅に削減できる。

[0163] また、第２実施形態では、リスト０及びリスト１に含まれる参照番号が同一の参照画像を指しており、且つＷＰパラメータが異なる場合、異なるＷＰパラメータを設定することが可能である。つまり、参照リスト内に同一の参照画像を指す組み合わせがあった場合、第１実施形態では、強制的に同一のＷＰパラメータを設定し、第２実施形態では、ＷＰパラメータが同一の場合

のみ、同じＷＰパラメータの冗長な表現を削除する。

[0164] （第３実施形態）

第３実施形態について説明する。図２８は、第３実施形態の符号化装置７００の構成の一例を示すブロック図である。第３実施形態の符号化装置７００は、動き情報メモリ７０１、動き情報取得部７０２、及び切替スイッチ７０３を更に備えている点で、第１実施形態の符号化装置１００と相違する。以下では、第１実施形態との相違点の説明を主に行い、第１実施形態と同様の機能を有する構成要素については、第１実施形態と同様の名称・符号を付し、その説明を省略する。

[0165] 動き情報メモリ７０１は、符号化が終了した画素ブロックに適用された動き情報を、参照動き情報として一時的に格納する。動き情報メモリ７０１では、動き情報に対してサブサンプリングなどの圧縮処理を行い、情報量を削減しても構わない。

[0166] 図２９は、第３実施形態の動き情報メモリ７０１の詳細例を示す図である。動き情報メモリ７０１は、図２９に示すように、フレーム又はスライス単位で保持されており、同一フレーム上の動き情報を参照動き情報７１０として格納する空間方向参照動き情報メモリ７０１Ａ、及び、既に符号化が終了したフレームの動き情報を参照動き情報７１０として格納する時間方向参照動き情報メモリ７０１Ｂを更に有する。時間方向参照動き情報メモリ７０１Ｂは、符号化対象フレームが予測に用いる参照フレームの数に応じて複数有しても構わない。

[0167] 参照動き情報７１０は、所定の領域単位（例えば、４×４画素ブロック単位）で空間方向参照動き情報メモリ７０１Ａ及び時間方向参照動き情報メモリ７０１Ｂ内に保持される。参照動き情報７１０は、その領域が後述するインター予測で符号化されたのか又は後述するイントラ予測で符号化されたのかを示す情報をさらに有する。また、画素ブロック（コーディングユニット又はプレディクションユニット）がＨ．２６４で規定されるスキップモード、ダイレクトモード又は後述するマージモードのように、動き情報内の動き

ベクトルの値が符号化されず、符号化済みの領域から予測された動き情報を用いてインター予測される場合においても、当該画素ブロックの動き情報が参照動き情報７１０として保持される。

[0168] 符号化対象のフレーム又はスライスの符号化処理が終了したら、当該フレームの空間方向参照動き情報メモリ７０１Ａは、次に符号化処理を行うフレームに用いる時間方向参照動き情報メモリ７０１Ｂとしてその扱いが変更される。この際、時間方向参照動き情報メモリ７０１Ｂのメモリ容量を削減するために、動き情報を圧縮し、圧縮された動き情報を時間方向参照動き情報メモリ７０１Ｂに格納しても構わない。

[0169] 動き情報取得部７０２は、動き情報メモリ７０１から参照動き情報が入力され、符号化画素ブロックに用いる動き情報Ｂを出力する。動き情報取得部７０２の詳細は後述する。

[0170] 切替スイッチ７０３は、後述する予測モード情報に従って動き情報取得部７０２から出力される動き情報Ｂと動き評価部１０９から出力される動き情報Ａの内、いずれかを選択して動き情報として予測画像生成部１０７に出力する。動き評価部１０９から出力される動き情報Ａは、図示せぬ予測動きベクトル取得部から取得する予測動きベクトルとの差分情報や、予測動きベクトル取得位置情報を符号化する。このような予測モードを以降インターモードと称する。一方、予測モード情報に従って動き情報取得部７０２から出力される動き情報Ｂは、隣接する画素ブロックから動き情報をマージして、符号化画素ブロックに対してそのまま適用されるため、その他の動き情報に関する情報（例えば、動きベクトル差分情報）の符号化は不要である。以降、このような予測モードをマージモードと称する。

[0171] 予測モード情報は、符号化制御部１１１が制御する予測モードに従っており、切替スイッチ７０３の切替情報を含む。

[0172] 以下、動き情報取得部７０２の詳細について説明する。

[0173] 動き情報取得部７０２は、参照動き情報を入力として、動き情報Ｂを切替スイッチ７０３に出力する。図３０Ａ及び図３０Ｂは、第３実施形態の符号

化画素ブロックに対する動き情報候補を導出するブロック位置の一例を示す図である。図３０Ａにおける Ａ～Ｅは、符号化画素ブロックに動き情報候補を導出する空間的に隣接する画素ブロック、図３０Ｂにおける Ｔは、動き情報候補を導出する時間的に隣接する画素ブロックを示している。

[0174] 図３１は、第３実施形態の複数の動き情報候補の画素ブロック位置と画素ブロック位置インデクス（ｉｄｘ）との関係の一例を示す図である。動き情報候補のブロック位置は、空間的に隣接するブロック位置（Ａ～Ｅ）、時間的に隣接する画素ブロック位置（Ｔ）の順序で並べられ、当該順序でＡｖａｉｌａｂｌｅ（インター予測が適用されている）な画素ブロックがあれば、マージモードの格納リストMergeCandListへ格納する動き情報の最大数（Ｎ－１）に達するまで順次ｉｄｘが割り当てられる。

[0175] 図３２は、第３実施形態のMergeCandListへの格納処理の一例を示すフローチャートである。

[0176] まず、動き情報取得部７０２は、MergeCandListへの格納数numMergeCandを０で初期化する（ステップＳ３０１）。

[0177] 続いて、動き情報取得部７０２は、全ての空間隣接ブロック(例えば、ブロックＡ～Ｅ)に対して、Ａｖａｉｌａｂｌｅであるか否かを判定する（ステップＳ３０２、Ｓ３０３）。Ａｖａｉｌａｂｌｅである場合（ステップＳ３０３でＹｅｓ）、動き情報取得部７０２は、MergeCandListに当該空間隣接ブロックの動き情報を格納し、numMergeCandをインクリメントする（ステップＳ３０４）。Ａｖａｉｌａｂｌｅでない場合（ステップＳ３０３でＮｏ）、動き情報取得部７０２は、ステップＳ３０４の処理を行わない。

[0178] 動き情報取得部７０２は、全ての空間隣接ブロックに対してステップＳ３０３、Ｓ３０４の処理を実行すると（ステップＳ３０５）、時間隣接ブロック（例えばブロックＴ）に対して、Ａｖａｉｌａｂｌｅであるか否かを判定する（ステップＳ３０６）。

[0179] Ａｖａｉｌａｂｌｅである場合（ステップＳ３０６でＹｅｓ）、動き情報取得部７０２は、MergeCandListに当該時間隣接ブロックＴの動き情報を格納

し、numMergeCandをインクリメントする（ステップＳ３０７）。Ａｖａｉｌａｂｌｅでない場合（ステップＳ３０６でＮｏ）、動き情報取得部７０２は、ステップＳ３０７の処理を行わない。

[0180] 続いて、動き情報取得部７０２は、MergeCandList内で重複する動き情報を削除し、削除した数だけnumMergeCandをデクリメントする（ステップＳ３０８）。MergeCandList内で重複する動き情報か否かを判定するために、MergeCandList内の２種類の動き情報（動きベクトルmv，参照番号refIdx）及びＷＰパラメータ情報が一致しているかを判定し、全ての情報が一致している場合には、一方をMergeCandList[]から削除する。

[0181] 図３３は、第３実施形態の動き情報の格納リストの一例を示す図であり、詳細には、Ｎ＝５として図３１に示す表に従った、各ｉｄｘに対応する動き情報（動きベクトルmv，参照番号refIdx）の格納リスト（MergeCandList[idx]）の一例を示している。図３３において、ｉｄｘ＝０は、リスト０の参照動き情報（動きベクトルmv0L0，参照番号refIdx0L0）、ｉｄｘ＝１は、リスト１の参照動き情報（動きベクトルmv1L1，参照番号refIdx1L1）、ｉｄｘ＝２～４は、リスト０及びリスト１の両方の参照動き情報（動きベクトルmv2L0，mv2L1，参照番号refIdx2L0，refIdx2L1）が格納されている。

[0182] 符号化画素ブロックは、最大Ｎ種類のｉｄｘのうちいずれかを選択し、選択したｉｄｘに対応する画素ブロックの参照動き情報をMergeCandList[idx]から導出し、動き情報Ｂとして出力する。なお符号化画素ブロックは、参照動き情報が存在しない場合、ゼロベクトルを有する動き情報を、予測動き情報候補Ｂとして出力する。選択したｉｄｘの情報（後述するシンタクスmerge_idx）は予測モード情報に含まれ、符号化部１１０において符号化される。

[0183] なお、空間及び時間的に隣接するブロックは図３０Ａ及び図３０Ｂに限らず、既に符号化済みの領域であればいずれの位置でも構わない。また、上記リストの別の例として、MergeCandListへの格納最大数（N）や順序は、上述の限りではなく任意の順序・最大数を用いても構わない。

[0184] 符号化スライスがＢスライスであって、空間・時間的に隣接する画素ブロ

ックの参照動き情報をMergeCandList[]へ格納しても格納最大数（N）に達しなかった場合には、その時点でMergeCandList[]に格納されているリスト０及びリスト１の動き情報を相互に組み合わせて、新たに双方向予測を生成しMergeCandList[]に格納する。

[0185] 図３４は、第３実施形態の動き情報の格納リストへの格納方法の一例を示すフローチャートである。

[0186] まず、動き情報取得部７０２は、空間・時間的に隣接する画素ブロックからのMergeCandListへの格納数をnumInputMergeCandとし、現在の格納数を示す変数numMergeCandをnumInputMergeCandで初期化するとともに、変数combIdx及びcombCntを０で初期化する（ステップＳ４０１）。

[0187] 続いて、動き情報取得部７０２は、新たに生成する双方向予測の組み合わせ（図３５参照）を用いて、combIdxから変数l0CandIdx，変数l1CandIdxを導出する（ステップＳ４０２）。なお、combIdxから変数l0CandIdx，変数l1CandIdxを導出する組み合わせは、図３６に示す例に限らず、重複しなければ任意の順序に並び替えても構わない。

[0188] 続いて、動き情報取得部７０２は、MergeCandList[l0CandIdx]に格納されている参照動き情報（双方向予測の場合はリスト０及びリスト１の２種類)をl0Candに設定する。l0Candにおける参照番号をrefIdxL0l0Cand、動きベクトルをmvL0l0Candと称する。同様に、動き情報取得部７０２は、MergeCandList[l1CandIdx]に格納されている参照動き情報をl1Candに設定する。l1Candにおける参照番号をrefIdxL1l1Cand、動きベクトルをmvL1l1Candと称する（ステップＳ４０３）。リスト０の動き情報としてl0Cand、リスト１の動き情報としてL1Candを用いて双方向予測を行う動き情報の組み合わせをcombCandと称する。

[0189] 続いて、動き情報取得部７０２は、条件式（１２）～（１５）を用いて、l0Candとl1Candが参照するブロックを同一であるか否かを判定する（ステップＳ４０４）。

[0190] l0Cand，l1CandがAvailableか？ …（１２）

[0191] RefPicOrderCnt(refIdxL0l0Cand,L0)!=RefPicOrderCnt(refIdxL1l1Cand,L1) …（１３）

[0192] mvL0l0Cand!=mvL1l1Cand …（１４）

[0193] WpParam(refIdxL0l0Cand,L0)!=WpParam(refIdxL1l1Cand,L1) …（１５）

[0194] RefPicOrderCnt(refIdx,LX)は、参照リストＸ(Ｘ＝０，１)において、参照番号refIdxが対応する参照フレームのPOC(Picture Order Count)を導出する関数である。WpParam(refIdx,LX)は、参照リストＸ(Ｘ＝０，１)において、参照番号refIdxが対応する参照フレームのＷＰパラメータ情報を導出する関数である。この関数は、リスト０の参照フレームとリスト１の参照フレームが、同一のＷＰパラメータ（ＷＰの有無WP_flag、重み係数Weight、オフセットOffset）ではない場合はＹｅｓ、完全に同一のパラメータである場合にはＮｏを返す。WpParam()の別の例として、全てのパラメータの一致を確認する必要はなく、ＷＰの有無WP_flag及びオフセットOffsetのみの一致を確認するなど、ＷＰパラメータの一部のデータのみを用いて判定を行ってもよい。

[0195] 条件式（１２）を満たし、かつ条件式（１３）～（１５）の何れかを満たす場合には（ステップＳ４０４でＹｅｓ）、動き情報取得部７０２は、mergeCandList[]内でl0Candとl1Candと同一の組み合わせを用いる双方向予測があるか否かを判定する（ステップＳ４０５）。

[0196] mergeCandList[]内でl0Candとl1Candと同一の組み合わせがない場合（ステップＳ４０５でＹｅｓ）、動き情報取得部７０２は、当該combCandをmergeCandListの最後尾に追加する（ステップＳ４０６）。具体的には、動き情報取得部７０２は、combCandをmergeCandList[numMergeCand]に代入し、numMergeCand及びcombCntをインクリメントする。

[0197] 続いて、動き情報取得部７０２は、combIdxをインクリメントする（ステップＳ４０７）。

[0198] 続いて、動き情報取得部７０２は、条件式（１６）～（１８）を用いて、mergeCandList [] への格納を終了するか否かを判定する（ステップＳ４０８

）。

[0199] combIdx == numInputMergeCand*(numInputMergeCand?1)) …（１６）

[0200] numMergeCand == maxNumMergeCand …（１７）

[0201] combCnt == N …（１８）

[0202] 条件式（１６）～（１８）の何れかを満たす場合には（ステップＳ４０８でＹｅｓ）、処理は終了となり、条件式（１６）～（１８）の何れも満たす場合には（ステップＳ４０８でＮｏ）、ステップＳ４０２へ戻る。

[0203] 以上が、符号化スライスがＢスライスであって、空間・時間的に隣接する画素ブロックの参照動き情報をMergeCandList[]へ格納しても格納最大数(N)に達しなかった場合の処理の説明である。以上により、動き情報取得部７０２から動き情報Ｂが切替スイッチ７０３に出力される。

[0204] 図３６は、第３実施形態のプレッドユニットシンタクス５１２の一例を示す図である。skip_flagは、プレディクションユニットシンタクスが属するコーディングユニットの予測モードがスキップモードであるか否かを示すフラグである。skip_flagが１である場合、予測モード情報以外のシンタクス（コーディングユニットシンタクス、プレディクションユニットシンタクス、トランスフォームユニットシンタクス）を符号化しないことを示す。skip_flagが０である場合、プレディクションユニットシンタクスが属するコーディングユニットの予測モードがスキップモードではないことを示す。

[0205] 次に符号化画素ブロック（プレディクションユニット）がマージモードであるか否かを示すフラグであるmerge_flagが符号化される。Merge_flagは、値が１である場合、プレディクションユニットがマージモードであることを示し、値が０である場合、プレディクションユニットがインターモードを用いることを示す。Merge_flagの場合、前述の画素ブロック位置インデクス(ｉｄｘ)を特定する情報であるmerge_idxが符号化される。

[0206] Merge_flagが１である場合、merge_flag、merge_idx以外のプレディクションユニットシンタクスは符号化する必要はなく、Merge_flagが０である場合

、プレディクションユニットがインターモードであることを示す。

[0207] 以上のように第３実施形態では、符号化装置７００は、重み付き動き補償とマージモードを同時に適用するにあたって、空間・時間的に隣接する画素ブロックの参照動き情報をマージモードの格納リスト（MergeCandList）内の重複する動き情報を削除する際に、動きベクトル及び参照番号が一致しても、ＷＰパラメータ情報が異なる場合に、MergeCandListから削除されてしまう問題を解消する。また、MergeCandListへ格納しても格納最大数（N）に達しなかった場合に、その時点でMergeCandListに格納されているリスト０及びリスト１の動き情報を相互に組み合わせて、新たに双方向予測を生成し、MergeCandList[]に格納する際に、双方向予測に用いる２種類の動き情報が同一のブロックを参照する問題を解消する。

[0208] 双方向予測に用いる２種類の動き情報が同一のブロックを参照する場合には、双方向予測による予測値と単方向予測による予測値の値が一致する。一般に双方向予測の予測効率は単方向予測の予測効率より高いために、双方向予測に用いる２種類の動き情報が同一のブロックを参照しないことが望ましい。符号化装置７００は、双方向予測に用いる２種類の動き情報が同一のブロックを参照するか否かを判定する際に、動きベクトル、参照番号から導出される参照フレーム番号位置（POC）の他に、重み付き動き補償のパラメータを判定項目に導入する。これにより、符号化装置７００は、動きベクトルと参照フレーム番号位置（POC）が同一であっても、重み付き動き補償のパラメータの異なる２種類の動き情報は同一のブロックを参照しないと判定し、マージモードの格納リストに格納される双方向予測に用いる動き情報の予測効率を向上させることができる。

[0209] なお、MergeCandListへの格納処理の別の例として、時間隣接ブロックＴの動き情報をMergeCandListに格納する際に、時間隣接ブロックＴの参照フレームにおけるＷＰパラメータと符号化画素ブロックの参照フレームにおけるＷＰパラメータが一致しているかを判定し、ＷＰパラメータが一致している場合のみ、MergeCandListに格納するようにしてもよい。これは、時間隣接ブロ

ックＴの参照フレームにおけるＷＰパラメータと符号化画素ブロックの参照フレームにおけるＷＰパラメータが異なる場合には、時間隣接ブロックＴと符号化画素ブロックで動き情報の相関が低下すると推測できるためである。

[0210] 図３７は、第３実施形態のMergeCandListへの格納処理の他の例を示すフローチャートである。図３８に示すフローチャートは、ステップＳ５０７を除き、図３２に示すフローチャートと同一である。

[0211] また、更に別の例として、時間隣接ブロックＴの参照フレームにおけるＷＰパラメータと符号化画素ブロックの参照フレームにおけるＷＰパラメータが異なる場合には、ブロックＴに空間的に隣接するブロックのうち、符号化画素ブロックの参照フレームにおけるＷＰパラメータが同一となるブロックをブロックＴと置き換えてもよい。この時、時間隣接ブロックＴと符号化画素ブロックで動き情報の相関は低下しない。

[0212] （第４実施形態）

第４実施形態では、第１実施形態の符号化装置で符号化された符号化データを復号する復号装置について説明する。

[0213] 図３８は、第４実施形態の復号装置８００の構成の一例を示すブロック図である。

[0214] 復号装置８００は、図示せぬ入力バッファなどに蓄積された符号化データを復号画像に復号し、出力画像として図示せぬ出力バッファに出力する。符号化データは、例えば、図１の符号化装置１００などから出力され、図示せぬ蓄積系、伝送系、又はバッファなどを経て、復号装置８００に入力される。

[0215] 復号装置８００は、図３８に示すように、復号部８０１と、逆量子化部８０２と、逆直交変換部８０３と、加算部８０４と、予測画像生成部８０５と、インデックス設定部８０６とを、備える。逆量子化部８０２、逆直交変換部８０３、加算部８０４、予測画像生成部８０５は、それぞれ、図１の逆量子化部１０４、逆直交変換部１０５、加算部１０６、予測画像生成部１０７、と実質的に同一又は類似の要素である。なお、図３８に示す復号制御部８

０７は、復号装置８００を制御するものであり、例えば、ＣＰＵなどにより実現できる。

[0216] 復号部８０１は、符号化データの復号のために、１フレーム又は１フィールド毎にシンタクスに基づいて解読を行う。復号部８０１は、各シンタクスの符号列を順次エントロピー復号し、予測モード、動きベクトル、及び参照番号などを含む動き情報、重み付き動き補償予測のためのインデックス情報、並びに量子化変換係数などの符号化対象ブロックの符号化パラメータを再生する。符号化パラメータとは、上記以外の変換係数に関する情報や量子化に関する情報などの復号に必要となる全てのパラメータである。復号部８０１は、動き情報、インデックス情報、及び量子化変換係数を出力し、量子化変換係数を逆量子化部８０２に入力し、インデックス情報をインデックス設定部８０６に入力し、動き情報を予測画像生成部８０５に入力する。

[0217] 逆量子化部８０２は、復号部８０１から入力された量子化変換係数に対して逆量子化処理を行い、復元変換係数を得る。具体的には、逆量子化部８０２は、復号部８０１において使用された量子化情報に従って逆量子化を行う。より詳細には、逆量子化部８０２は、量子化情報によって導出された量子化ステップサイズを量子化変換係数に乗算し、復元変換係数を得る。逆量子化部８０２は、復元変換係数を出力し、逆直交変換部８０３に入力する。

[0218] 逆直交変換部８０３は、逆量子化部８０２から入力された復元変換係数に対して、符号化側において行われた直交変換に対応する逆直交変換を行い、復元予測誤差を得る。逆直交変換部８０３は、復元予測誤差を出力し、加算部８０４に入力する。

[0219] 加算部８０４は、逆直交変換部８０３から入力された復元予測誤差と、対応する予測画像とを加算し、復号画像を生成する。加算部８０４は、復号画像を出力し、予測画像生成部８０５に入力する。また加算部８０４は、復号画像を出力画像として外部に出力する。出力画像は、その後図示せぬ外部の出力バッファなどに一時的に蓄積され、例えば、復号制御部８０７によって管理される出力タイミングに従って、図示せぬディスプレイやモニタなどの

表示装置系又は映像デバイス系へ出力される。

[0220] インデックス設定部８０６は、復号部８０１から入力されたインデックス情報を受け取り、参照リスト（リスト番号）と参照画像（参照番号）とを確認して、ＷＰパラメータ情報を出力し、予測画像生成部８０５に入力する。

[0221] 図３９は、第４実施形態のインデックス設定部８０６の構成の一例を示すブロック図である。インデックス設定部８０６は、図３９に示すように、参照画像確認部９０１と、ＷＰパラメータ生成部９０２とを、備える。

[0222] 参照画像確認部９０１は、復号部８０１からインデックス情報を受け取り、２つの参照リストに含まれる参照番号が互いに同一の参照画像を指しているかどうかを確認する。

[0223] ここで、参照リスト内の参照番号は、例えば、Ｈ．２６４などで規定される方法により既に解読されている。このため、参照画像確認部９０１は、Ｈ．２６４などで規定されているＤｅｃｏｄｅｄ Ｐｉｃｔｕｒｅ Ｂｕｆｆｅｒ（ＤＰＢ）の管理に従って、導出された参照リストと参照番号を用いて、同一の参照画像を指す組み合わせがあるかどうかを確認できる。なお、ＤＰＢの制御については、Ｈ．２６４などに記述されている方式でもよいし、別の方式であってもよい。ここでは、ＤＰＢ制御により参照リストと参照番号が予め確定していればよい。

[0224] 参照画像確認部９０１は、共通参照リストを作成し、図１４に示すスキャン順で参照リストと参照番号を確認する。なお、共通参照リストは、数式（１０）に示す疑似コードに従って作成されている。

[0225] ＷＰパラメータ生成部９０２は、参照画像確認部９０１で確認された参照リストと参照番号との関係を元に、作成された共通参照リストから、参照リストと参照番号に対応するＷＰパラメータ情報を生成して出力し、予測画像生成部８０５に入力する。ＷＰパラメータ情報については、図９Ａ及び図９Ｂを参照して既に説明しているため、説明を省略する。

[0226] ＷＰパラメータ生成部９０２は、共通リストに従って、common_scan_list()関数及びcommon_scan_ref_idx()関数を用いて、スキャンで参照リストと参

照番号を引き、抜けている場所にＷＰパラメータ情報を補填する。

[0227] つまり、図１５、図１６の共通リストを、それぞれ、図１１、図１３へと復元し、これらの対応に基づいて、共通リストに対応するインデックス情報を、図９Ａ及び図９Ｂに示されるＷＰパラメータ情報に割り当てる。

[0228] なお、図１４に示すスキャン順序は一例であり、予め定めたスキャン順序であればこれ以外のスキャン順序を用いてもよい。また、数式（１０）で示した疑似コードは一例であり、本処理の目的を実現できれば処理の追加や冗長な処理の削減も可能である。

[0229] 図３８に戻り、予測画像生成部８０５は、復号部８０１から入力された動き情報、インデックス設定部８０６から入力されたＷＰパラメータ情報、及び加算部８０４から入力された復号画像を用いて、予測画像を生成する。

[0230] ここで、図４を参照しながら、予測画像生成部８０５の詳細について説明する。予測画像生成部８０５は、予測画像生成部１０７同様、複数フレーム動き補償部２０１と、メモリ２０２と、単方向動き補償部２０３と、予測パラメータ制御部２０４と、参照画像セレクタ２０５と、フレームメモリ２０６と、参照画像制御部２０７と、を備える。

[0231] フレームメモリ２０６は、参照画像制御部２０７の制御の下、加算部１０６から入力された復号画像を参照画像として格納する。フレームメモリ２０６は、参照画像を一時保持するための複数のメモリセットＦＭ１～ＦＭＮ（Ｎ≧２）を有する。

[0232] 予測パラメータ制御部２０４は、復号部８０１から入力される動き情報に基づいて、参照画像番号と予測パラメータとの複数の組み合わせをテーブルとして用意している。ここで、動き情報とは、動き補償予測で用いられる動きのズレ量を示す動きベクトルや参照画像番号、単方向／双方向予測などの予測モードに関する情報などを指す。予測パラメータは、動きベクトル及び予測モードに関する情報を指す。そして予測パラメータ制御部２０４は、動き情報に基づいて、予測画像の生成に用いる参照画像番号と予測パラメータとの組み合わせを選択して出力し、参照画像番号を参照画像セレクタ２０５

に入力し、予測パラメータを単方向動き補償部２０３に入力する。

[0233] 参照画像セレクタ２０５は、フレームメモリ２０６が有するフレームメモリＦＭ１～ＦＭＮのいずれの出力端を接続するかを、予測パラメータ制御部２０４から入力された参照画像番号に従って切り替えるスイッチである。参照画像セレクタ２０５は、例えば、参照画像番号が０であれば、ＦＭ１の出力端を参照画像セレクタ２０５の出力端に接続し、参照画像番号がＮ－１であれば、ＦＭＮの出力端を参照画像セレクタ２０５の出力端に接続する。参照画像セレクタ２０５は、フレームメモリ２０６が有するフレームメモリＦＭ１～ＦＭＮのうち、出力端が接続されているフレームメモリに格納されている参照画像を出力し、単方向動き補償部２０３へ入力する。なお、復号装置８００では、予測画像生成部８０５以外で参照画像は利用されないため、予測画像生成部８０５の外部へ参照画像を出力しなくてもよい。

[0234] 単方向予測動き補償部２０３は、予測パラメータ制御部２０４から入力された予測パラメータと参照画像セレクタ２０５から入力された参照画像に従って、動き補償予測処理を行い、単方向予測画像を生成する。動き補償予測については、図５を参照して既に説明しているため、説明を省略する。

[0235] 単方向予測動き補償部２０３は、単方向予測画像を出力し、メモリ２０２に一時的に格納する。ここで、動き情報（予測パラメータ）が双方向予測を示す場合には、複数フレーム動き補償部２０１が２種類の単方向予測画像を用いて重み付き予測を行うため、単方向予測動き補償部２０３は、１つ目に対応する単方向予測画像をメモリ２０２に格納し、２つ目に対応する単法予測画像を複数フレーム動き補償部２０１に直接出力する。ここでは、１つ目に対応する単方向予測画像を第一予測画像とし、２つ目に対応する単方向予測画像を第二予測画像とする。

[0236] なお、単方向動き補償部２０３を２つ用意し、それぞれが２つの単方向予測画像を生成するようにしてもよい。この場合、動き情報（予測パラメータ）が単方向予測を示すときには、単方向動き補償部２０３が、１つ目の単方向予測画像を第一予測画像として複数フレーム動き補償部２０１に直接出力

すればよい。

[0237] 複数フレーム動き補償部２０１は、メモリ２０２から入力される第一予測画像、単方向予測動き補償部２０３から入力される第二予測画像、及び動き評価部１０９から入力されるＷＰパラメータ情報を用いて、重み付き予測を行って予測画像を生成する。複数フレーム動き補償部２０１は、予測画像を出力し、加算部８０４に入力する。

[0238] ここで、図６を参照しながら、複数フレーム動き補償部２０１の詳細について説明する。複数フレーム動き補償部２０１は、予測画像生成部１０７同様、デフォルト動き補償部３０１と、重み付き動き補償部３０２と、ＷＰパラメータ制御部３０３と、ＷＰセレクタ３０４、３０５とを、備える。

[0239] ＷＰパラメータ制御部３０３は、インデックス設定部８０６から入力されるＷＰパラメータ情報に基づいて、ＷＰ適用フラグ及び重み情報を出力し、ＷＰ適用フラグをＷＰセレクタ３０４、３０５に入力し、重み情報を重み付き動き補償部３０２に入力する。

[0240] ここで、ＷＰパラメータ情報は、重み係数の固定小数点精度、第一予測画像に対応する第一ＷＰ適用フラグ，第一重み係数，及び第一オフセット、並びに第二予測画像に対応する第二ＷＰ適応フラグ，第二重み係数，及び第二オフセットの情報を含む。ＷＰ適用フラグは、該当する参照画像及び信号成分毎に設定可能なパラメータであり、重み付き動き補償予測を行うかどうかを示す。重み情報は、重み係数の固定小数点精度、第一重み係数、第一オフセット、第二重み係数、及び第二オフセットの情報を含む。なおＷＰパラメータ情報は、第１実施形態と同様の情報を表す。

[0241] 詳細には、ＷＰパラメータ制御部３０３は、インデックス設定部８０６からＷＰパラメータ情報が入力されると、ＷＰパラメータ情報を第一ＷＰ適用フラグ、第二ＷＰ適用フラグ、及び重み情報に分離して出力し、第一ＷＰ適用フラグをＷＰセレクタ３０４に入力し、第二ＷＰ適用フラグをＷＰセレクタ３０５に入力し、重み情報を重み付き動き補償部３０２に入力する。

[0242] ＷＰセレクタ３０４、３０５は、ＷＰパラメータ制御部３０３から入力さ

れたＷＰ適用フラグに基づいて、各々の予測画像の接続端を切り替える。ＷＰセレクタ３０４、３０５は、各々のＷＰ適用フラグが０の場合、各々の出力端をデフォルト動き補償部３０１へ接続する。そしてＷＰセレクタ３０４、３０５は、第一予測画像及び第二予測画像を出力し、デフォルト動き補償部３０１に入力する。一方、ＷＰセレクタ３０４、３０５は、各々のＷＰ適用フラグが１の場合、各々の出力端を重み付き動き補償部３０２へ接続する。そしてＷＰセレクタ３０４、３０５は、第一予測画像及び第二予測画像を出力し、重み付き動き補償部３０２に入力する。

[0243] デフォルト動き補償部３０１は、ＷＰセレクタ３０４、３０５から入力された２つの単方向予測画像（第一予測画像及び第二予測画像）を元に平均値処理を行い、予測画像を生成する。具体的には、デフォルト動き補償部３０１は、第一ＷＰ適用フラグ及び第二ＷＰ適用フラグが０の場合、数式（１）に基づいて平均値処理を行う。

[0244] なお、動き情報（予測パラメータ）で示される予測モードが単方向予測である場合、デフォルト動き補償部３０１は、第一予測画像のみを用いて、数式（４）に基づいて最終的な予測画像を算出する。

[0245] 重み付き動き補償部３０２は、ＷＰセレクタ３０４、３０５から入力された２つの単方向予測画像（第一予測画像及び第二予測画像）とＷＰパラメータ制御部３０３から入力された重み情報とを元に重み付き動き補償を行う。具体的には、重み付き動き補償部３０２は、第一ＷＰ適用フラグ及び第二ＷＰ適用フラグが１の場合、数式（７）に基づいて重み付き処理を行う。

[0246] なお、重み付き動き補償部３０２は、第一予測画像及び第二予測画像と予測画像との演算精度が異なる場合、固定小数点精度である$logWD_C$を数式（８）のように制御することで丸め処理を実現する。

[0247] また、動き情報（予測パラメータ）で示される予測モードが単方向予測である場合、重み付き動き補償部３０２は、第一予測画像のみを用いて、数式（９）に基づいて最終的な予測画像を算出する。

[0248] なお、重み付き動き補償部３０２は、第一予測画像及び第二予測画像と予

測画像との演算精度が異なる場合、固定小数点精度である$logWD_C$を双方向予測時と同様に数式（８）のように制御することで丸め処理を実現する。

[0249] 重み係数の固定小数点精度については、図７を参照して既に説明しているため、説明を省略する。なお、単方向予測の場合には、第二予測画像に対応する各種パラメータ（第二ＷＰ適応フラグ，第二重み係数，及び第二オフセットの情報）は利用されないため、予め定めた初期値に設定されていてもよい。

[0250] 復号部８０１は、図１８に示すシンタクス５００を利用する。シンタクス５００は、復号部８０１の復号対象の符号化データの構造を示している。シンタクス５００については、図１８を参照して既に説明しているため、説明を省略する。また、ピクチャパラメータセットシンタクス５０５については、符号化が復号である点を除き、図１９を参照して既に説明しているため、説明を省略する。また、スライスヘッダーシンタクス５０７についても、符号化が復号である点を除き、図２０を参照して既に説明しているため、説明を省略する。また、プレッドウェイトテーブルシンタクス５０８についても、符号化が復号である点を除き、図２１を参照して既に説明しているため、説明を省略する。

[0251] 以上のように第４実施形態では、復号化装置８００は、双方向予測を選択可能な双方向予測スライスで、参照画像が同一であるが参照画像番号が異なる２つのインデックスを用いて重み付き動き補償を行う際に、同一の値を持つインデックスを２回復号することにより、符号化効率が低下する問題を解消する。

[0252] 双方向予測スライスで所持する２つの参照リストとリスト毎に定められた参照番号を、共通リストと共通の参照番号に並び変えることにより、参照リスト内に同一の参照画像を指す参照番号の組み合わせが含まれないようになるので、復号化装置８００は、冗長なインデックスの符号量を削減することができる。

[0253] （第４実施形態の変形例）

第４実施形態の変形例について説明する。第４実施形態の変形例では、復号部８０１が用いるシンタクス要素が第４実施形態のシンタクス要素と異なる。シーケンスパラメータセットシンタクス５０４については、符号化が復号である点を除き、図２２を参照して既に説明しているため、説明を省略する。アダプテーションパラメータセットシンタクス５０６についても、符号化が復号である点を除き、図２３を参照して既に説明しているため、説明を省略する。プレッドウェイトテーブルシンタクス５０８についても、符号化が復号である点を除き、図２４を参照して既に説明しているため、説明を省略する。

[0254] 以上のように第４実施形態の変形例によれば、プレッドウェイトテーブルシンタクス５０８をアダプテーションパラメータセットシンタクス５０６から呼び出す構造とすることにより、１フレームを複数のスライスに分割した際のＷＰパラメータ情報の符号量を大幅に削減することができる。

[0255] 例えば、利用する３種類の異なるＷＰパラメータ情報を持つアダプテーションパラメータセットシンタクス５０６を先に復号しておき、スライスヘッダーシンタクス５０７が、状況に応じてaps_idを用いて、必要なＷＰパラメータ情報を呼び出すことにより、スライスヘッダーシンタクス５０７でＷＰパラメータ情報を常に復号する構成よりも、符号量を削減できる。

[0256] （第５実施形態）

第５実施形態では、第２実施形態の符号化装置で符号化された符号化データを復号する復号装置について説明する。第５実施形態の復号装置１０００では、インデックス設定部１００６の構成が第４実施形態の復号装置８００と相違する。以下では、第４実施形態との相違点の説明を主に行い、第４実施形態と同様の機能を有する構成要素については、第４実施形態と同様の名称・符号を付し、その説明を省略する。

[0257] 図４０は、第５実施形態のインデックス設定部１００６の構成の一例を示すブロック図である。インデックス設定部１００６は、図４０に示すように、再利用判定部１００１と、ＷＰパラメータ生成部１００２とを、備える。

[0258] 再利用判定部１００１は、再利用フラグをチェックし、リスト０のＷＰパラメータを再利用するかどうかを確認する。ＷＰパラメータ生成部１００２は、リスト０のＷＰパラメータを再利用する場合、リスト０のどのＷＰパラメータをコピーするかの情報に従って、参照先のリスト０のＷＰパラメータをリスト１のＷＰパラメータにコピーする。再利用フラグと参照先を示す参照番号に関する情報は、シンタクス要素に記述されており、ＷＰパラメータ生成部１００２がインデックス情報を解釈することにより、ＷＰパラメータ情報を復元する。

[0259] 再利用判定部１００１は、例えば、数式（１１）に示す疑似コードに従って、リスト０のＷＰパラメータを再利用するかどうかを確認する。

[0260] プレッドウェイトテーブルシンタクス５０８については、符号化が復号である点を除き、図２７を参照して既に説明しているため、説明を省略する。

[0261] 以上のように第５実施形態では、復号化装置１０００は、双方向予測を選択可能な双方向予測スライスで、参照画像が同一であるが参照画像番号が異なる２つのインデックスを用いて重み付き動き補償を行う際に、同一の値を持つインデックスを２回復号することにより、符号化効率が低下する問題を解消する。

[0262] 復号化装置１０００は、リスト０のインデックスを従来通り復号し、リスト１のインデックスを復号する際に、参照リスト内に同一の参照画像を指す参照番号の組み合わせが存在するか確認し、インデックスが同一の場合には、リスト１で用いるインデックスをリスト０で用いたインデックスで再利用することにより、同一のインデックスが２度復号されることを避け、冗長なインデックスの符号量を削減することができる。

[0263] （第６実施形態）

第６実施形態では、第３実施形態の符号化装置で符号化された符号化データを復号する復号装置について説明する。図４１は、第６実施形態の復号装置１１００の構成の一例を示すブロック図である。第６実施形態の復号装置１１００は、動き情報メモリ１１０１、動き情報取得部１１０２、及び切替

スイッチ１１０３を更に備えている点で、第４実施形態の復号装置８００と相違する。以下では、第４実施形態との相違点の説明を主に行い、第４実施形態と同様の機能を有する構成要素については、第４実施形態と同様の名称・符号を付し、その説明を省略する。

[0264] 動き情報メモリ１１０１は、復号が終了した画素ブロックに適用された動き情報を参照動き情報として一時的に格納する。動き情報メモリ１１０１では、動き情報に対してサブサンプリングなどの圧縮処理を行って情報量を削減しても構わない。動き情報メモリ１１０１は、図２９に示すように、フレーム又はスライス単位で保持されており、同一フレーム上の動き情報を参照動き情報７１０として格納する空間方向参照動き情報メモリ７０１Ａ、及び既に復号が終了したフレームの動き情報を参照動き情報７１０として格納する時間方向参照動き情報メモリ７０１Ｂを更に有する。時間方向参照動き情報メモリ７０１Ｂは、復号対象フレームが予測に用いる参照フレームの数に応じて、複数有しても構わない。

[0265] 参照動き情報７１０は、所定の領域単位（例えば、４×４画素ブロック単位）で空間方向参照動き情報メモリ７０１Ａ及び時間方向参照動き情報メモリ７０１Ｂ内に保持される。参照動き情報７１０は、その領域が後述するインター予測で符号化されたのか又は後述するイントラ予測で符号化されたのかを示す情報をさらに有する。また、画素ブロック（コーディングユニット又はプレディクションユニット）がＨ．２６４で規定されるスキップモード、ダイレクトモード又は後述するマージモードのように、動き情報内の動きベクトルの値が復号されず、復号済みの領域から予測された動き情報を用いてインター予測される場合においても、当該画素ブロックの動き情報が参照動き情報７１０として保持される。

[0266] 復号対象のフレーム又はスライスの復号処理が終了したら、当該フレームの空間方向参照動き情報メモリ７０１Ａは、次に復号処理を行うフレームに用いる時間方向参照動き情報メモリ７０１Ｂとしてその扱いが変更される。この際、時間方向参照動き情報メモリ７０１Ｂのメモリ容量を削減するため

に、動き情報を圧縮し、圧縮された動き情報を時間方向参照動き情報メモリ７０１Ｂに格納しても構わない。

[0267] 動き情報取得部１１０２は、動き情報メモリ１１０１から参照動き情報が入力され、復号画素ブロックに用いる動き情報Ｂを出力する。動き情報取得部１１０２の動作は、第３実施形態の動き情報取得部７０２と同一であるため、その説明を省略する。

[0268] 切替スイッチ１１０３は、後述する予測モード情報に従って動き情報取得部１１０２から出力される動き情報Ｂと復号部８０１から出力される動き情報Ａの内、いずれかを選択して動き情報として予測画像生成部８０５に出力する。復号部８０１から出力される動き情報Ａは、図示せぬ予測動きベクトル取得部から取得する予測動きベクトルとの差分情報や、予測動きベクトル取得位置情報を復号する。このような予測モードを以降インターモードと称する。一方、予測モード情報に従って動き情報取得部１１０２から出力される動き情報Ｂは、隣接する画素ブロックから動き情報をマージして、復号画素ブロックに対してそのまま適用されるため、その他の動き情報に関する情報（例えば、動きベクトル差分情報）の復号は不要である。以降、このような予測モードをマージモードと称する。

[0269] 予測モード情報は、復号制御部８０７が制御する予測モードに従っており、切替スイッチ１１０３の切替情報を含む。

[0270] プレッドユニットシンタクス５１２については、符号化が復号である点を除き、図３６を参照して既に説明しているため、説明を省略する。

[0271] 第６実施形態の効果は、第３実施形態と同様であるため、その説明を省略する。

[0272] なお、MergeCandListへの格納処理の別の例として、時間隣接ブロックＴの動き情報をMergeCandListに格納する際に、時間隣接ブロックＴの参照フレームにおけるＷＰパラメータと復号画素ブロックの参照フレームにおけるＷＰパラメータが一致しているかを判定し、ＷＰパラメータが一致している場合のみ、MergeCandListに格納するようにしてもよい。これは、時間隣接ブロッ

クＴの参照フレームにおけるＷＰパラメータと復号画素ブロックの参照フレームにおけるＷＰパラメータが異なる場合には、時間隣接ブロックＴと復号画素ブロックで動き情報の相関が低下すると推測できるためである。

[0273] また、更に別の例として、時間隣接ブロックＴの参照フレームにおけるＷＰパラメータと復号画素ブロックの参照フレームにおけるＷＰパラメータが異なる場合には、ブロックＴに空間的に隣接するブロックのうち、復号画素ブロックの参照フレームにおけるＷＰパラメータが同一となるブロックをブロックＴと置き換えてもよい。この時、時間隣接ブロックＴと復号画素ブロックで動き情報の相関は低下しない。

[0274] （変形例）

なお、第１～第３実施形態を選択的に用いてもよい。つまり、予めいずれの方法を選択するかの情報を定めておき、双方向スライスが選択された場合に、第１実施形態の手法を用いるか第２実施形態の手法を用いるかを示す情報により、切り替えてもよい。例えば、図２０に示すスライスヘッダーシンタクスに、これらの手法を切り替えるフラグを設定し、符号化状況に合わせて選択することも容易に可能である。また、特定のハードウェアの構成に合わせて予めいずれかの手法を用いるかを定めておいてもよい。第１～第３実施形態の予測画像生成部及びインデックス設定部は、ハードウェア実装及びソフトウェア実装のいずれにも好適である。

[0275] また、第４～第６実施形態を選択的に用いてもよい。つまり、予めいずれの方法を選択するかの情報を定めておき、双方向スライスが選択された場合に、第４実施形態の手法を用いるか第５実施形態の手法を用いるかを示す情報により、切り替えてもよい。例えば、図２０に示すスライスヘッダーシンタクスに、これらの手法を切り替えるフラグを設定し、符号化状況に合わせて選択することも容易に可能である。また、特定のハードウェアの構成に合わせて予めいずれかの手法を用いるかを定めておいてもよい。第４～第６実施形態の予測画像生成部及びインデックス設定部は、ハードウェア実装及びソフトウェア実装のいずれにも好適である。

[0276] また、第１～第６実施形態で例示したシンタクステーブルの行間には、上記実施形態において規定していないシンタクス要素が挿入されてもよいし、その他の条件分岐に関する記述が含まれていてもよい。また、シンタクステーブルを複数のテーブルに分割したり、複数のシンタクステーブルを統合したりしてもよい。また、例示した各シンタクス要素の用語は、任意に変更可能である。

[0277] また、第１実施形態の変形例は、第２実施形態に容易に適用できる。この場合、リスト０とリスト１に対応する参照画像の最大数を設定してもよいし、予め定めた最大数としてもよい。

[0278] また、第４実施形態の変形例は、第５実施形態に容易に適用できる。この場合、リスト０とリスト１に対応する参照画像の最大数を設定してもよいし、予め定めた最大数としてもよい。

[0279] 上記第１～第６実施形態では、フレームを１６×１６画素サイズなどの矩形ブロックに分割し、画面左上のブロックから右下に向かって順に符号化／復号を行う例について説明している（図３Ａを参照）。しかしながら、符号化順序及び復号順序はこの例に限定されない。例えば、右下から左上に向かって順に符号化及び復号が行われてもよいし、画面中央から画面端に向かって渦巻を描くように符号化及び復号が行われてもよい。更に、右上から左下に向かって順に符号化及び復号が行われてもよいし、画面端から画面中央に向かって渦巻きを描くように符号化及び復号が行われてもよい。この場合、符号化順序によって参照できる隣接画素ブロックの位置が変わるので、適宜利用可能な位置に変更すればよい。

[0280] 上記第１～第６実施形態では、４×４画素ブロック、８×８画素ブロック、１６×１６画素ブロックなどの予測対象ブロックサイズを例示して説明を行ったが、予測対象ブロックは均一なブロック形状でなくてもよい。例えば、予測対象ブロックサイズは、１６×８画素ブロック、８×１６画素ブロック、８×４画素ブロック、４×８画素ブロックなどであってもよい。また、１つのコーディングツリーブロック内で全てのブロックサイズを統一させる

必要はなく、複数の異なるブロックサイズを混在させてもよい。１つのコーディングツリーブロック内で複数の異なるブロックサイズを混在させる場合、分割数の増加に伴って分割情報を符号化又は復号するための符号量も増加する。そこで、分割情報の符号量と局部復号画像または復号画像の品質との間のバランスを考慮して、ブロックサイズを選択することが望ましい。

[0281] 上記第１～第６実施形態では、簡単化のために、輝度信号と色差信号における予測処理とを区別せず、色信号成分に関して包括的な説明を記述した。しかしながら、予測処理が輝度信号と色差信号との間で異なる場合には、同一または異なる予測方法が用いられてよい。輝度信号と色差信号との間で異なる予測方法が用いられるならば、色差信号に対して選択した予測方法を輝度信号と同様の方法で符号化又は復号できる。

[0282] 上記第１～第６実施形態では、簡単化のために、輝度信号と色差信号における重み付き動き補償予測処理とを区別せず、色信号成分に関して包括的な説明を記述した。しかしながら、重み付き動き補償予測処理が輝度信号と色差信号との間で異なる場合には、同一または異なる重み付き動き補償予測処理が用いられてよい。輝度信号と色差信号との間で異なる重み付き動き補償予測処理が用いられるならば、色差信号に対して選択した重み付き動き補償予測処理を輝度信号と同様の方法で符号化又は復号できる。

[0283] 上記第１～第６実施形態では、シンタクス構成に示す表の行間には、本実施形態で規定していないシンタクス要素が挿入されることも可能であるし、それ以外の条件分岐に関する記述が含まれていても構わない。或いは、シンタクステーブルを複数のテーブルに分割、統合することも可能である。また、必ずしも同一の用語を用いる必要は無く、利用する形態によって任意に変更しても構わない。

[0284] 以上説明したように、各実施形態は、重み付き動き補償予測を行う際に冗長な情報を符号化する問題を解消しつつ、高効率な重み付き動き補償予測処理を実現する。故に、各実施形態によれば、符号化効率が向上し、ひいては主観画質も向上する。

[0285] 本発明のいくつかの実施形態を説明したが、これらの実施形態は、例として提示したものであり、発明の範囲を限定することは意図していない。これら新規な実施形態は、その他の様々な形態で実施されることが可能であり、発明の要旨を逸脱しない範囲で、種々の省略、置き換え、変更を行うことができる。これら実施形態やその変形は、発明の範囲や要旨に含まれるとともに、特許請求の範囲に記載された発明とその均等の範囲に含まれる。

[0286] 例えば、上記各実施形態の処理を実現するプログラムを、コンピュータで読み取り可能な記憶媒体に格納して提供することも可能である。記憶媒体としては、磁気ディスク、光ディスク（ＣＤ－ＲＯＭ、ＣＤ－Ｒ、ＤＶＤ等）、光磁気ディスク（ＭＯ等）、半導体メモリなど、プログラムを記憶でき、かつ、コンピュータが読み取り可能な記憶媒体であれば、その記憶形式は何れの形態であってもよい。

[0287] また、上記各実施形態の処理を実現するプログラムを、インターネットなどのネットワークに接続されたコンピュータ（サーバ）上に格納し、ネットワーク経由でコンピュータ（クライアント）にダウンロードさせてもよい。

符号の説明

[0288] １００、６００、７００　符号化装置

１０１　減算部

１０２　直交変換部

１０３　量子化部

１０４　逆量子化部

１０５　逆直交変換部

１０６　加算部

１０７　予測画像生成部

１０８、６０８　インデックス設定部

１０９　動き評価部

１１０　符号化部

１１１　符号化制御部

２０１　複数フレーム動き補償部

２０２　メモリ

２０３　単方向動き補償部

２０４　予測パラメータ制御部

２０５　参照画像セレクタ

２０６　フレームメモリ

２０７　参照画像制御部

３０１　デフォルト動き補償部

３０２　重み付き動き補償部

３０３　ＷＰパラメータ制御部

３０４、３０５　ＷＰセレクタ

４０１、９０１　参照画像確認部

４０２、６０２　インデックス生成部

６０１、１００１　再利用判定部

７０１、１１０１　動き情報メモリ

７０２、１１０２　動き情報取得部

７０３、１１０３　切替スイッチ

８００、１０００、１１００　復号装置

８０１　復号部

８０２　逆量子化部

８０３　逆直交変換部

８０４　加算部

８０５　予測画像生成部

８０６、１００６　インデックス設定部

８０７　復号制御部

９０２、１００２　ＷＰパラメータ生成部

請求の範囲

[請求項1] 第１参照画像が参照する１以上の参照画像を含む組み合わせを示す第１インデックスと第２参照画像が参照する１以上の参照画像を含む組み合わせを示す第２インデックスとを設定し、前記第１インデックスに含まれる１以上の参照画像の番号と前記第２インデックスに含まれる１以上の参照画像の番号とを、予め定められた順序でスキャンして、同一の参照画像を含まない組み合わせに並び替えた共通インデックスを生成するインデックス生成ステップと、

前記共通インデックスを符号化する符号化ステップと、

を含む符号化方法。

[請求項2] 前記第１インデックス及び前記第２インデックスは、それぞれ、１以上の重み係数を含み、

入力画像を分割した複数のブロックのうちの対象ブロックで参照される前記第１参照画像及び前記第２参照画像と前記第１参照画像及び前記第２参照画像それぞれに対応する動きベクトルとで動き補償予測した値に、前記第１参照画像及び前記第２参照画像それぞれの重み係数を乗じることにより、前記対象ブロックの予測画像を生成する予測画像生成ステップと、

前記入力画像と前記予測画像に基づいて、前記対象ブロックの量子化変換係数を生成する量子化変換係数生成ステップと、を更に含み、

前記符号化ステップでは、更に、前記量子化変換係数及び前記動きベクトルを符号化することを特徴とする請求項１に記載の符号化方法。

[請求項3] 前記第１インデックス及び前記第２インデックスは、それぞれ、１以上のオフセットを含み、

前記予測画像生成ステップでは、入力画像を分割した複数のブロックのうちの対象ブロックで参照される前記第１参照画像及び前記第２参照画像と前記第１参照画像及び前記第２参照画像それぞれに対応す

る前記動きベクトルとで動き補償予測して動き補償予測画像を生成し、生成した前記動き補償予測画像に前記第１参照画像及び前記第２参照画像それぞれの重み係数を乗じ、前記第１参照画像及び前記第２参照画像それぞれのオフセットを加算することにより、前記対象ブロックの予測画像を生成することを特徴とする請求項２に記載の符号化方法。

［請求項4］ 前記符号化ステップでは、前記共通インデックスを含む情報で構成される１以上のピクチャ単位の上位情報と、前記量子化変換係数及び前記動きベクトルとを含む情報で構成され、前記上位情報の識別番号を参照する下位情報とを、異なるユニットとして符号化することを特徴とする請求項３に記載の符号化方法。

［請求項5］ 前記共通インデックスは、許容する最大枚数の参照画像及びピクチャより上位のユニット構造が保持する参照画像の枚数に関する情報を含むことを特徴とする請求項４に記載の符号化方法。

［請求項6］ 前記インデックス生成ステップでは、前記第２インデックスに含まれる１以上の参照画像のうち前記第１インデックスに含まれる参照画像と同一の参照画像を、前記第１インデックスに含まれる前記参照画像を参照する参照情報に置き換え、

前記符号化ステップでは、前記第１インデックス及び前記参照情報に置き換え後の前記第２インデックスを符号化することを特徴とする請求項１に記載の符号化方法。

［請求項7］ 第１参照画像が参照する１以上の参照画像を含む組み合わせを示す第１インデックスと第２参照画像が参照する１以上の参照画像を含む組み合わせを示す第２インデックスとを設定し、前記第２インデックスに含まれる１以上の参照画像のうち前記第１インデックスに含まれる参照画像と同一の参照画像を、前記第１インデックスに含まれる前記参照画像を参照する参照情報に置き換えるインデックス生成ステップと、

前記第１インデックス及び前記参照情報に置き換え後の前記第２インデックスを符号化する符号化ステップと、

を含む符号化方法。

[請求項8] 前記第１インデックス及び前記第２インデックスは、それぞれ、１以上の重み係数を含み、

入力画像を分割した複数のブロックのうちの対象ブロックで参照される前記第１参照画像及び前記第２参照画像と前記第１参照画像及び前記第２参照画像それぞれに対応する動きベクトルとで動き補償予測した値に、前記第１参照画像及び前記第２参照画像それぞれの重み係数を乗じることにより、前記対象ブロックの予測画像を生成する予測画像生成ステップと、

前記入力画像と前記予測画像に基づいて、前記対象ブロックの量子化変換係数を生成する量子化変換係数生成ステップと、を更に含み、

前記符号化ステップでは、更に、前記量子化変換係数及び前記動きベクトルを符号化することを特徴とする請求項７に記載の符号化方法。

[請求項9] 前記第１インデックス及び前記第２インデックスは、それぞれ、１以上のオフセットを含み、

前記予測画像生成ステップでは、入力画像を分割した複数のブロックのうちの対象ブロックで参照される前記第１参照画像及び前記第２参照画像と前記第１参照画像及び前記第２参照画像それぞれに対応する前記動きベクトルとで動き補償予測して動き補償予測画像を生成し、生成した前記動き補償予測画像に前記第１参照画像及び前記第２参照画像それぞれの重み係数を乗じ、前記第１参照画像及び前記第２参照画像それぞれのオフセットを加算することにより、前記対象ブロックの予測画像を生成することを特徴とする請求項８に記載の符号化方法。

[請求項10] 符号化データを復号して、第１インデックスに含まれる１以上の参

照画像の番号と第２インデックスに含まれる１以上の参照画像の番号とが、予め定められた順序でスキャンされて、同一の参照画像を含まない組み合わせに並び替えられている共通インデックスを生成する復号ステップと、

生成された前記共通インデックスから前記第１インデックスと前記第２インデックスとを生成するインデックス生成ステップと、

を含む復号方法。

［請求項11］ 前記第１インデックス及び前記第２インデックスは、それぞれ、１以上の重み係数を含み、

前記復号ステップでは、前記符号化データを復号して、量子化変換係数及び動きベクトルを生成し、

前記量子化変換係数に基づく復号画像を分割した複数のブロックのうちの対象ブロックで参照される前記第１参照画像及び前記第２参照画像と前記第１参照画像及び前記第２参照画像それぞれに対応する前記動きベクトルとで動き補償予測した値に、前記第１参照画像及び前記第２参照画像それぞれの重み係数を乗じることにより、前記対象ブロックの予測画像を生成する予測画像生成ステップと、

前記復号画像と前記予測画像とを用いて、出力画像を生成する出力画像生成ステップと、

を更に含むことを特徴とする請求項１０に記載の復号方法。

［請求項12］ 前記第１インデックス及び前記第２インデックスは、それぞれ、１以上のオフセットを含み、

前記予測画像生成ステップでは、前記復号画像を分割した複数のブロックのうちの対象ブロックで参照される前記第１参照画像及び前記第２参照画像と前記第１参照画像及び前記第２参照画像それぞれに対応する前記動きベクトルとで動き補償予測して動き補償予測画像を生成し、生成した前記動き補償予測画像に前記第１参照画像及び前記第２参照画像それぞれの重み係数を乗じ、前記第１参照画像及び前記第

２参照画像それぞれのオフセットを加算することにより、前記対象ブロックの予測画像を生成することを特徴とする請求項１１に記載の復号方法。

[請求項13] 前記復号ステップでは、前記共通インデックスを含む情報で構成される１以上のピクチャ単位の上位情報と、前記量子化変換係数及び前記動きベクトルとを含む情報で構成され、前記上位情報の識別番号を参照する下位情報とを、異なるユニットとして復号することを特徴とする請求項１２に記載の復号方法。

[請求項14] 前記共通インデックスは、許容する最大枚数の参照画像及びピクチャより上位のユニット構造が保持する参照画像の枚数に関する情報を含むことを特徴とする請求項１３に記載の復号方法。

[請求項15] 前記復号ステップでは、前記符号化データを復号して、第１参照画像が参照する１以上の参照画像を含む組み合わせを示す第１インデックスと、１以上の参照画像のうち前記第１インデックスに含まれる参照画像と同一の参照画像が、前記第１インデックスに含まれる前記参照画像を参照する参照情報に置き換えられた第２インデックスと、を生成し、

前記インデックス生成ステップでは、前記第２インデックスの前記参照情報を前記第１インデックスに含まれる前記参照画像に置き換えることを特徴とする請求項１０に記載の復号方法。

[請求項16] 符号化データを復号して、第１参照画像が参照する１以上の参照画像を含む組み合わせを示す第１インデックスと、１以上の参照画像のうち前記第１インデックスに含まれる参照画像と同一の参照画像が、前記第１インデックスに含まれる前記参照画像を参照する参照情報に置き換えられた第２インデックスと、を生成する復号ステップと、

前記第２インデックスの前記参照情報を前記第１インデックスに含まれる前記参照画像に置き換えるインデックス生成ステップと、

を含む復号方法。

［請求項17］ 前記第１インデックス及び前記第２インデックスは、それぞれ、１以上の重み係数を含み、

前記復号ステップでは、前記符号化データを復号して、量子化変換係数及び動きベクトルを生成し、

前記量子化変換係数に基づく復号画像を分割した複数のブロックのうちの対象ブロックで参照される前記第１参照画像及び前記第２参照画像と前記第１参照画像及び前記第２参照画像それぞれに対応する前記動きベクトルとで動き補償予測した値に、前記第１参照画像及び前記第２参照画像それぞれの重み係数を乗じることにより、前記対象ブロックの予測画像を生成する予測画像生成ステップと、

前記復号画像と前記予測画像とを用いて、出力画像を生成する出力画像生成ステップと、

を更に含むことを特徴とする請求項１６に記載の復号方法。

［請求項18］ 前記第１インデックス及び前記第２インデックスは、それぞれ、１以上のオフセットを含み、

前記予測画像生成ステップでは、前記復号画像を分割した複数のブロックのうちの対象ブロックで参照される前記第１参照画像及び前記第２参照画像と前記第１参照画像及び前記第２参照画像それぞれに対応する前記動きベクトルとで動き補償予測して動き補償予測画像を生成し、生成した前記動き補償予測画像に前記第１参照画像及び前記第２参照画像それぞれの重み係数を乗じ、前記第１参照画像及び前記第２参照画像それぞれのオフセットを加算することにより、前記対象ブロックの予測画像を生成することを特徴とする請求項１７に記載の復号方法。