

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7670608号
(P7670608)

(45)発行日 令和7年4月30日(2025.4.30)

(24)登録日 令和7年4月21日(2025.4.21)

(51)国際特許分類

F I

H 0 4 N 19/70 (2014.01)

H 0 4 N 19/70

請求項の数 14 (全85頁)

(21)出願番号	特願2021-520863(P2021-520863)	(73)特許権者	514136668
(86)(22)出願日	令和2年5月21日(2020.5.21)		パナソニック インテレクチュアル プロ
(86)国際出願番号	PCT/JP2020/020204		パティ コーポレーション オブ アメリカ
(87)国際公開番号	WO2020/235659		Panasonic Intellec
(87)国際公開日	令和2年11月26日(2020.11.26)		tual Property Corpo
審査請求日	令和5年4月21日(2023.4.21)		ration of America
(31)優先権主張番号	62/850,682		アメリカ合衆国 9 0 5 0 4 カリフォル
(32)優先日	令和1年5月21日(2019.5.21)		ニア州, トーランス, スイート 4 5 0
(33)優先権主張国・地域又は機関	米国(US)	(74)代理人	100109210
			弁理士 新居 広守
		(74)代理人	100137235
			弁理士 寺谷 英作
		(74)代理人	100131417
			弁理士 道坂 伸一

最終頁に続く

(54)【発明の名称】 符号化装置、復号装置、符号化方法および復号方法

(57)【特許請求の範囲】

【請求項1】

回路と、
メモリと、を備え、
前記回路は、前記メモリを用いて、
可変長符号化処理において、1つのスライスを1以上のデータアクセス領域に符号化し、
シーケンスヘッダに記述されたフラグと、前記1以上のデータアクセス領域の個数とに
基づいて、ビットストリーム上における前記1以上のデータアクセス領域のそれぞれの先
頭位置を特定するための1つ以上のオフセットを、スライスヘッダに符号化し、
前記1つ以上のオフセットを符号化する、と前記フラグが示し、且つ、前記個数が2以
上の場合に、前記1つ以上のオフセットは符号化され、
前記1つ以上のオフセットを符号化しない、と前記フラグが示す場合には、前記1つ以
上のオフセットは符号化されず、
前記フラグは、前記個数が2以上か否かに関わらず、前記1つ以上のオフセットの符号
化をするか否かを切り替え可能とする、
符号化装置。

【請求項2】

前記スライスヘッダに符号化する前記1つ以上のオフセットの前に0もしくは1ビット
以上の信号を挿入し、前記1つ以上のオフセットをバイトアラインされた位置から記述す
る、

請求項 1 に記載の符号化装置。

【請求項 3】

前記スライスヘッダに符号化する前記 1 つ以上のオフセットの後に 0 もしくは 1 ビット以上の信号を挿入し、前記 1 つ以上のオフセットの次の信号をバイトアラインされた位置から記述する、

請求項 1 または 2 に記載の符号化装置。

【請求項 4】

前記スライスヘッダに符号化する前記 1 つ以上のオフセットのそれぞれを表す信号に続いて、値が 0 でない 1 ビット以上の信号を挿入する、

請求項 1 ～ 3 のいずれか 1 項に記載の符号化装置。

10

【請求項 5】

前記スライスヘッダに符号化する前記 1 つ以上のオフセットのそれぞれを表す信号の前に、値が 0 でない 1 ビット以上の信号を挿入する、

請求項 1 ～ 3 のいずれか 1 項に記載の符号化装置。

【請求項 6】

前記 1 つ以上のオフセットに含まれる 1 つのオフセットを 1 ビット以上の複数の信号に分割し、前記複数の信号のうち連続する 2 つの信号の間に、値が 0 でない 1 ビット以上の信号を挿入する、

請求項 1 ～ 5 のいずれか 1 項に記載の符号化装置。

【請求項 7】

前記 1 つ以上のオフセットに含まれる 1 つのオフセットのビット長が、所定値以上だった場合、前記 1 つのオフセットを、複数の信号に分割し、前記複数の信号のうち連続する 2 つの信号の間に、値が 0 でない 1 ビット以上の信号を挿入する、

請求項 6 に記載の符号化装置。

20

【請求項 8】

回路と、

メモリと、を備え、

前記回路は、前記メモリを用いて、

可変長復号処理において、1 つのスライスが 1 以上のデータアクセス領域に分割されたビットストリームを復号し、シーケンスヘッダに記述されたフラグと、前記 1 以上のデータアクセス領域の個数とに基づいて、前記ビットストリーム上における前記 1 以上のデータアクセス領域のそれぞれの先頭位置を特定するための 1 つ以上のオフセットを、スライスヘッダから復号し、

30

前記 1 つ以上のオフセットを復号する、と前記フラグが示し、且つ、前記個数が 2 以上の場合に、前記 1 つ以上のオフセットは復号され、

前記 1 つ以上のオフセットを復号しない、と前記フラグが示す場合には、前記 1 つ以上のオフセットは復号されず、

前記フラグは、前記個数が 2 以上か否かに関わらず、前記 1 つ以上のオフセットの復号をするか否かを切り替え可能とする、

復号装置。

40

【請求項 9】

前記スライスヘッダから復号する前記 1 つ以上のオフセットのそれぞれを示す信号を、バイトアラインされた位置から読み解く、

請求項 8 に記載の復号装置。

【請求項 10】

前記スライスヘッダから復号する 1 つ以上の前記オフセットのそれぞれを表す信号に続いて、1 ビット以上の信号を挿入された信号として読み解き、バイトアラインする、

請求項 8 または 9 に記載の復号装置。

【請求項 11】

前記 1 つ以上のオフセットに含まれる 1 つのオフセットを、複数の信号に分割し、前記

50

複数の信号のうち連続する2つの信号の間に、値が0でない1ビット以上の信号が挿入された信号として読み解く、

請求項8～10のいずれか1項に記載の復号装置。

【請求項12】

前記1つ以上のオフセットに含まれる1つのオフセットのビット長が、所定値以上だった場合、前記1つのオフセットを、1ビット以上の複数の信号に分割し、前記複数の信号のそれぞれの間の、値が0でない1ビット以上の信号が挿入された信号として読み解く、

請求項8～11のいずれか1項に記載の復号装置。

【請求項13】

可変長符号化処理において、1つのスライスを1以上のデータアクセス領域に符号化し、シーケンスヘッダに記述されたフラグと、前記1以上のデータアクセス領域の個数とに基づいて、ビットストリーム上における前記1以上のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセットを、スライスヘッダに符号化し、

前記1つ以上のオフセットを符号化する、と前記フラグが示し、且つ、前記個数が2以上の場合に、前記1つ以上のオフセットは符号化され、

前記1つ以上のオフセットを符号化しない、と前記フラグが示す場合には、前記1つ以上のオフセットは符号化されず、

前記フラグは、前記個数が2以上か否かに関わらず、前記1つ以上のオフセットの符号化をするか否かを切り替え可能とする、

符号化方法。

【請求項14】

可変長復号処理において、1つのスライスが1以上のデータアクセス領域に分割されたビットストリームを復号し、シーケンスヘッダに記述されたフラグと、前記1以上のデータアクセス領域の個数とに基づいて、前記ビットストリーム上における前記1以上のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセットを、スライスヘッダから復号し、

前記1つ以上のオフセットを復号する、と前記フラグが示し、且つ、前記個数が2以上の場合に、前記1つ以上のオフセットは復号され、

前記1つ以上のオフセットを復号しない、と前記フラグが示す場合には、前記1つ以上のオフセットは復号されず、

前記フラグは、前記個数が2以上か否かに関わらず、前記1つ以上のオフセットの復号をするか否かを切り替え可能とする、

復号方法。

【発明の詳細な説明】

【技術分野】

【0001】

本開示は、ビデオコーディングに関し、例えば、動画像の符号化および復号におけるシステム、構成要素、ならびに方法などに関する。

【背景技術】

【0002】

ビデオコーディング技術は、H.261およびMPEG-1から、H.264/AVC (Advanced Video Coding)、MPEG-LA、H.265/HEVC (High Efficiency Video Coding)、およびH.266/VVC (Versatile Video Codec) へ進歩している。この進歩に伴い、様々な用途において増え続けるデジタルビデオデータ量を処理するために、ビデオコーディング技術の改良および最適化を提供することが常に必要とされている。

【先行技術文献】

【非特許文献】

【0003】

【文献】H.265 (ISO/IEC 23008-2 HEVC) / HEVC (High

10

20

30

40

50

Efficiency Video Coding)

【発明の概要】

【発明が解決しようとする課題】

【0004】

このような符号化方式においては、符号化効率の改善や画質の改善や、回路規模の削減などのために、新たな方式の提案が望まれている。

【0005】

本開示における実施の形態またはその一部で開示された構成または方法のそれぞれは、例えば、符号化効率の改善、符号化／復号化処理量の削減、回路規模の削減、符号化／復号化速度の改善、符号化および復号化において、フィルタ、ブロック、サイズ、動きベクトル、参照ピクチャ、参照ブロックなどの構成要素／動作の適切な選択などのうちの、少なくともいずれか1つに貢献しうる。例えば、アフィンモードにおける制御ポイントの動きベクトルの導出の精度向上、および、アフィンモードにおける制御ポイントの動きベクトルの導出の際の符号量削減において、ビデオコーディングにおけるさらなる進歩、改良および最適化に寄与しうる。

【0006】

なお、本開示は、上記以外の利益を提供し得る構成または方法の開示も含む。例えば、処理量の増加を抑えつつ、符号化効率を改善する構成または方法などである。

【課題を解決するための手段】

【0007】

本開示の一態様に係る符号化装置は、回路と、メモリと、を備え、前記回路は、前記メモリを用いて、可変長符号化処理において、1つのスライスを1以上のデータアクセス領域に符号化し、シーケンスヘッダに記述されたフラグと、前記1以上のデータアクセス領域の個数とに基づいて、ビットストリーム上における前記1以上のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセットを、スライスヘッダに符号化し、前記1つ以上のオフセットを符号化する、と前記フラグが示し、且つ、前記個数が2以上の場合に、前記1つ以上のオフセットは符号化され、前記1つ以上のオフセットを符号化しない、と前記フラグが示す場合には、前記1つ以上のオフセットは符号化されず、前記フラグは、前記個数が2以上か否かに関わらず、前記1つ以上のオフセットの符号化をするか否かを切り替え可能とする。

【0008】

なお、これらの包括的又は具体的な態様は、システム、装置、方法、集積回路、コンピュータプログラム、又は、コンピュータ読み取り可能なCD-ROMなどの非一時的な記録媒体で実現されてもよく、システム、装置、方法、集積回路、コンピュータプログラム、及び、記録媒体の任意な組み合わせで実現されてもよい。

【0009】

上記のような符号化方式に関して、符号化効率の改善、画質の改善、処理量の削減、回路規模の削減、又は、フィルタ、ブロック、サイズ、動きベクトル、参照ピクチャ又は参照ブロック等の要素又は動作の適切な選択等のため、新たな方式の提案が望まれている。

【0010】

本開示は、例えば、符号化効率の改善、画質の改善、処理量の削減、回路規模の削減、処理速度の改善、及び、要素又は動作の適切な選択等のうち1つ以上に貢献し得る構成又は方法を提供する。なお、本開示は、上記以外の利益に貢献し得る構成又は方法を含み得る。

【発明の効果】

【0011】

本開示の一態様に係る構成又は方法は、例えば、符号化効率の改善、画質の改善、処理量の削減、回路規模の削減、処理速度の改善、及び、要素又は動作の適切な選択等のうち1つ以上に貢献し得る。なお、本開示の一態様に係る構成又は方法は、上記以外の利益に貢献してもよい。

10

20

30

40

50

【図面の簡単な説明】

【0012】

【図1】図1は、実施の形態に係る符号化装置の機能構成を示すブロック図である。

【図2】図2は、符号化装置による全体的な符号化処理の一例を示すフローチャートである。

【図3】図3は、ブロック分割の一例を示す概念図である。

【図4A】図4Aは、スライスの構成の一例を示す概念図である。

【図4B】図4Bは、タイルの構成の一例を示す概念図である。

【図5A】図5Aは、様々な変換タイプに対応する変換基底関数を示す表である。

【図5B】図5Bは、SVT (Spatially Varying Transform) の一例を示す概念図である。 10

【図6A】図6Aは、ALF (adaptive loop filter) で用いられるフィルタの形状の一例を示す概念図である。

【図6B】図6Bは、ALFで用いられるフィルタの形状の他の一例を示す概念図である。

【図6C】図6Cは、ALFで用いられるフィルタの形状の他の一例を示す概念図である。

【図7】図7は、DBF (deblocking filter) として機能するループフィルタ部の詳細な構成の一例を示すブロック図である。

【図8】図8は、ブロック境界に対して対称なフィルタ特性を有するデブロッキング・フィルタの例を示す概念図である。

【図9】図9は、デブロッキング・フィルタ処理が行われるブロック境界を説明するための概念図である。 20

【図10】図10は、Bs値の一例を示す概念図である。

【図11】図11は、符号化装置の予測処理部で行われる処理の一例を示すフローチャートである。

【図12】図12は、符号化装置の予測処理部で行われる処理の他の例を示すフローチャートである。

【図13】図13は、符号化装置の予測処理部で行われる処理の他の例を示すフローチャートである。

【図14】図14は、実施の形態のイントラ予測における67個のイントラ予測モードの一例を示す概念図である。 30

【図15】図15は、インター予測の基本的な処理の流れの一例を示すフローチャートである。

【図16】図16は、動きベクトル導出の一例を示すフローチャートである。

【図17】図17は、動きベクトル導出の他の例を示すフローチャートである。

【図18】図18は、動きベクトル導出の他の例を示すフローチャートである。

【図19】図19は、ノーマルインターモードによるインター予測の例を示すフローチャートである。

【図20】図20は、マージモードによるインター予測の例を示すフローチャートである。

【図21】図21は、マージモードによる動きベクトル導出処理の一例を説明するための概念図である。 40

【図22】図22は、FRUC (frame rate up conversion) 処理の一例を示すフローチャートである。

【図23】図23は、動き軌道に沿う2つのブロック間でのパターンマッチング（バイラテラルマッチング）の一例を説明するための概念図である。

【図24】図24は、カレントピクチャ内のテンプレートと参照ピクチャ内のブロックとの間でのパターンマッチング（テンプレートマッチング）の一例を説明するための概念図である。

【図25A】図25Aは、複数の隣接ブロックの動きベクトルに基づくサブブロック単位の動きベクトルの導出の一例を説明するための概念図である。

【図25B】図25Bは、3つの制御ポイントを有するアフィンモードにおけるサブプロ 50

ック単位の動きベクトルの導出の一例を説明するための概念図である。

【図 2 6 A】図 2 6 A は、アフィンマージモードを説明するための概念図である。

【図 2 6 B】図 2 6 B は、2 つの制御ポイントを有するアフィンマージモードを説明するための概念図である。

【図 2 6 C】図 2 6 C は、3 つの制御ポイントを有するアフィンマージモードを説明するための概念図である。

【図 2 7】図 2 7 は、アフィンマージモードの処理の一例を示すフローチャートである。

【図 2 8 A】図 2 8 A は、2 つの制御ポイントを有するアフィンインターモードを説明するための概念図である。

【図 2 8 B】図 2 8 B は、3 つの制御ポイントを有するアフィンインターモードを説明するための概念図である。

10

【図 2 9】図 2 9 は、アフィンインターモードの処理の一例を示すフローチャートである。

【図 3 0 A】図 3 0 A は、カレントブロックが 3 つの制御ポイントを有し、隣接ブロックが 2 つの制御ポイントを有するアフィンインターモードを説明するための概念図である。

【図 3 0 B】図 3 0 B は、カレントブロックが 2 つの制御ポイントを有し、隣接ブロックが 3 つの制御ポイントを有するアフィンインターモードを説明するための概念図である。

【図 3 1 A】図 3 1 A は、DMVR (decoder motion vector refinement) を含むマージモードを示すフローチャートである。

【図 3 1 B】図 3 1 B は、DMVR 処理の一例を説明するための概念図である。

【図 3 2】図 3 2 は、予測画像の生成の一例を示すフローチャートである。

20

【図 3 3】図 3 3 は、予測画像の生成の他の例を示すフローチャートである。

【図 3 4】図 3 4 は、予測画像の生成の他の例を示すフローチャートである。

【図 3 5】図 3 5 は、OBMC (overlapped block motion compensation) 処理による予測画像補正処理の一例を説明するためのフローチャートである。

【図 3 6】図 3 6 は、OBMC 処理による予測画像補正処理の一例を説明するための概念図である。

【図 3 7】図 3 7 は、2 つの三角形の予測画像の生成を説明するための概念図である。

【図 3 8】図 3 8 は、等速直線運動を仮定したモデルを説明するための概念図である。

【図 3 9】図 3 9 は、LIC (local illumination compensation) 処理による輝度補正処理を用いた予測画像生成方法の一例を説明するための概念図である。

30

【図 4 0】図 4 0 は、符号化装置の実装例を示すブロック図である。

【図 4 1】図 4 1 は、実施の形態に係る復号装置の機能構成を示すブロック図である。

【図 4 2】図 4 2 は、復号装置による全体的な復号処理の一例を示すフローチャートである。

【図 4 3】図 4 3 は、復号装置の予測処理部で行われる処理の一例を示すフローチャートである。

【図 4 4】図 4 4 は、復号装置の予測処理部で行われる処理の他の例を示すフローチャートである。

40

【図 4 5】図 4 5 は、復号装置におけるノーマルインターモードによるインター予測の例を示すフローチャートである。

【図 4 6】図 4 6 は、復号装置の実装例を示すブロック図である。

【図 4 7】図 4 7 は、entry__point__offset を説明するための図である。

【図 4 8】図 4 8 は、entry__point__offset のシンタックス構成の例を示す図である。

【図 4 9】図 4 9 は、entry__point__offset を符号化する方法の例を説明するための図である。

【図 5 0】図 5 0 は、entry__point__offset に関する課題を説明するための図である。

50

【図 5 1】図 5 1 は、`entry_point_offset`に関する課題を説明するための別の図である。

【図 5 2】図 5 2 は、実施の形態における`entry_point_offset`のシンタックス構成の第 1 の例を示す図である。

【図 5 3】図 5 3 は、実施の形態における`entry_point_offset`のシンタックス構成の第 2 の例を示す図である。

【図 5 4】図 5 4 は、実施の形態における`entry_point_offset`のシンタックス構成の第 3 の例を示す図である。

【図 5 5】図 5 5 は、実施の形態における符号化装置の処理を示すフローチャートである。

【図 5 6】図 5 6 は、実施の形態における復号装置の処理を示すフローチャートである。

【図 5 7】図 5 7 は、コンテンツ配信サービスを実現するコンテンツ供給システムの全体構成を示すブロック図である。

【図 5 8】図 5 8 は、スケーラブル符号化時の符号化構造の一例を示す概念図である。

【図 5 9】図 5 9 は、スケーラブル符号化時の符号化構造の一例を示す概念図である。

【図 6 0】図 6 0 は、web ページの表示画面例を示す概念図である。

【図 6 1】図 6 1 は、web ページの表示画面例を示す概念図である。

【図 6 2】図 6 2 は、スマートフォンの一例を示すブロック図である。

【図 6 3】図 6 3 は、スマートフォンの構成例を示すブロック図である。

【発明を実施するための形態】

【0013】

例えば、本開示の一態様に係る符号化装置は、回路と、メモリと、を備え、前記回路は、前記メモリを用いて、可変長符号化処理において、1つのスライスを複数のデータアクセス領域に分割して符号化する際に、生成するビットストリーム上における前記複数のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセット信号を、ヘッダ領域に符号化するか否かを指定する信号を、シーケンスヘッダ、ピクチャヘッダまたは拡張補助領域に記述し、前記1つ以上のオフセット信号を、前記ヘッダ領域に符号化すると指定された場合、前記1つ以上のオフセット信号を、ヘッダ領域に符号化し、前記1つ以上のオフセット信号を、前記ヘッダ領域に符号化しないと指定された場合、1つのスライスを複数の前記データアクセス領域に分割する場合でも、前記オフセット信号を前記ヘッダ領域に符号化しない。

【0014】

これにより、符号化装置は、処理スライスを構成する複数のデータアクセス領域の先頭位置を、複雑な管理処理を行わずに、把握することができる。よって、符号化装置は、処理量および回路規模を削減することができる。

【0015】

また、例えば、本開示の一態様に係る符号化装置は、前記ヘッダ領域に符号化する前記1つ以上のオフセット信号の前に0もしくは1ビット以上の信号を挿入し、前記1つ以上のオフセット信号をバイトアラインされた位置から記述する。

【0016】

これにより、符号化装置は、処理スライスを構成する複数のデータアクセス領域の先頭位置をバイト境界から記述することができる。

【0017】

また、例えば、本開示の一態様に係る符号化装置は、前記ヘッダ領域に符号化する前記1つ以上のオフセット信号の後に0もしくは1ビット以上の信号を挿入し、前記1つ以上のオフセット信号の次の信号をバイトアラインされた位置から記述する。

【0018】

これにより、符号化装置は、処理スライスを構成する複数のデータアクセス領域の先頭位置をバイト境界から記述することができる。

【0019】

また、例えば、本開示の一態様に係る符号化装置は、前記ヘッダ領域に符号化する前記

10

20

30

40

50

1つ以上のオフセット信号のそれぞれを表す信号に続いて、値が0でない1ビット以上の信号を挿入する。

【0020】

これにより、符号化装置は、処理スライスを構成する複数のデータアクセス領域の先頭位置を表す信号を複数の信号に分割して、バイト境界から記述することができる。

【0021】

また、例えば、本開示の一態様に係る符号化装置は、前記ヘッダ領域に符号化する前記1つ以上のオフセット信号のそれぞれを表す信号の前に、値が0でない1ビット以上の信号を挿入する。

【0022】

これにより、符号化装置は、0でない1ビット以上の信号を記述することで、処理スライスを構成する複数のデータアクセス領域の先頭位置を表す信号を複数の信号に分割して、バイト境界から記述することができる。

【0023】

また、例えば、本開示の一態様に係る復号装置は、前記1つ以上のオフセット信号に含まれる1つのオフセット信号を1ビット以上の複数の信号に分割し、前記複数の信号のうち連続する2つの信号の間に、値が0でない1ビット以上の信号を挿入する。

【0024】

これにより、復号装置は、処理スライスを構成する複数のデータアクセス領域の先頭位置を、複雑な管理処理を行わずに、把握することができる。よって、符号化装置は、処理量および回路規模を削減することができる。

【0025】

また、例えば、本開示の一態様に係る復号装置は、前記1つ以上のオフセット信号に含まれる1つのオフセット信号のビット長が、所定値以上だった場合、前記1つのオフセット信号を、複数の信号に分割し、前記複数の信号のうち連続する2つの信号の間に、値が0でない1ビット以上の信号を挿入する。

【0026】

これにより、復号装置は、処理スライスを構成する複数のデータアクセス領域の先頭位置をバイト境界から記述することができる。

【0027】

また、例えば、本開示の一態様に係る復号装置は、前記ヘッダ領域に復号する1つ以上の前記オフセット信号のそれぞれを表す信号に続いて、1ビット以上の信号を挿入された信号として読み解き、バイトアラインする。

【0028】

これにより、復号装置は、処理スライスを構成する複数のデータアクセス領域の先頭位置をバイト境界から記述することができる。

【0029】

また、例えば、本開示の一態様に係る復号装置は、前記1つ以上のオフセット信号に含まれる1つのオフセット信号を、複数の信号に分割し、前記複数の信号のうち連続する2つの信号の間に、値が0でない1ビット以上の信号が挿入された信号として読み解く。

【0030】

これにより、復号装置は、処理スライスを構成する複数のデータアクセス領域の先頭位置を表す信号を複数の信号に分割して、バイト境界から記述することができる。

【0031】

また、例えば、本開示の一態様に係る復号装置は、前記1つ以上のオフセット信号に含まれる1つのオフセット信号のビット長が、所定値以上だった場合、前記1つのオフセット信号を、1ビット以上の複数の信号に分割し、前記複数の信号のそれぞれの間の、値が0でない1ビット以上の信号が挿入された信号として読み解く。

【0032】

これにより、復号装置は、0でない1ビット以上の信号を記述することで、処理スライ

10

20

30

40

50

スを構成する複数のデータアクセス領域の先頭位置を表す信号を複数の信号に分割して、バイト境界から記述することができる。

【0033】

また、例えば、本開示の一態様に係る符号化方法は、可変長符号化処理において、1つのスライスを複数のデータアクセス領域に分割して符号化する際に、生成するビットストリーム上における前記複数のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセット信号を、ヘッダ領域に符号化するか否かを指定する信号を、シーケンスヘッダ、ピクチャヘッダまたは拡張補助領域に記述し、前記1つ以上のオフセット信号を、前記ヘッダ領域に符号化すると指定された場合、オフセット信号を、前記1つ以上のオフセット信号をヘッダ領域に符号化するか否かの指定に従って、ヘッダ領域に符号化し、前記1つ以上のオフセット信号として符号化される複数のオフセット信号のうち連続する2つのオフセット信号の間に値が0でない1ビット以上の信号を挿入し、前記1つ以上のオフセット信号を、前記ヘッダ領域に符号化しないと指定された場合、1つのスライスを複数の前記データアクセス領域に分割する場合でも、前記オフセット信号を前記ヘッダ領域に符号化しない。

10

【0034】

これにより、符号化方法は、上記符号化装置と同様の効果を奏することができる。

【0035】

また、例えば、本開示の一態様に係る復号方法は、可変長復号処理において、1つのスライスが複数のデータアクセス領域に分割されたビットストリームを復号する際に、前記ビットストリーム上における前記複数のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセット信号を、ヘッダ領域に含まれるか否かを指定する信号を、シーケンスヘッダ、ピクチャヘッダまたは拡張補助領域から読み解き、前記1つ以上のオフセット信号を、前記ヘッダ領域から復号すると指定された場合、前記1つ以上のオフセット信号をヘッダ領域から復号し、1つ以上の前記オフセット信号として復号される複数のオフセット信号のうち連続する2つのオフセット信号の間に挿入されている値が0でない1ビット以上の信号を復号し、前記1つ以上のオフセット信号を、前記ヘッダ領域から復号しないと指定された場合、1つのスライスを複数の前記データアクセス領域に分割する場合でも、前記オフセット信号を前記ビットストリームから復号しない。

20

【0036】

これにより、復号方法は、上記復号装置と同様の効果を奏することができる。

30

【0037】

また、例えば、本開示の一態様に係る符号化装置は、分割部と、イントラ予測部と、インター予測部と、ループフィルタ部と、変換部と、量子化部と、エントロピー符号化部とを備えてもよい。

【0038】

前記分割部は、ピクチャを複数のブロックに分割してもよい。前記イントラ予測部は、前記複数のブロックに含まれるブロックに対してイントラ予測を行ってもよい。前記インター予測部は、前記ブロックに対してインター予測を行ってもよい。前記変換部は、前記イントラ予測又は前記インター予測によって得られる予測画像と、原画像との予測誤差を変換して、変換係数を生成してもよい。前記量子化部は、前記変換係数を量子化して量子化係数を生成してもよい。前記エントロピー符号化部は、前記量子化係数を符号化して符号化ビットストリームを生成してもよい。前記ループフィルタ部は、前記ブロックの再構成画像にフィルタを適用してもよい。

40

【0039】

また、例えば、前記符号化装置は、複数のピクチャを含む動画像を符号化する符号化装置であってもよい。

【0040】

そして、エントロピー符号化部は、可変長符号化処理において、1つのスライスを複数のデータアクセス領域に分割して符号化する際に、生成するビットストリーム上における

50

前記複数のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセット信号を、ヘッダ領域に符号化するか否かを指定する信号を、シーケンスヘッダ、ピクチャヘッダまたは拡張補助領域に記述し、前記1つ以上のオフセット信号を、前記ヘッダ領域に符号化すると指定された場合、オフセット信号を、前記1つ以上のオフセット信号をヘッダ領域に符号化するか否かの指定に従って、ヘッダ領域に符号化し、前記1つ以上のオフセット信号として符号化される複数のオフセット信号のうち連続する2つのオフセット信号の間に値が0でない1ビット以上の信号を挿入し、前記1つ以上のオフセット信号を、前記ヘッダ領域に符号化しないと指定された場合、1つのスライスを複数の前記データアクセス領域に分割する場合でも、前記オフセット信号を前記ヘッダ領域に符号化しなくてもよい。

10

【0041】

また、例えば、本開示の一態様に係る復号装置は、エントロピー復号部と、逆量子化部と、逆変換部と、イントラ予測部と、インター予測部と、ループフィルタ部とを備えてもよい。

【0042】

前記エントロピー復号部は、符号化ビットストリームからピクチャ内のブロックの量子化係数を復号してもよい。前記逆量子化部は、前記量子化係数を逆量子化して変換係数を取得してもよい。前記逆変換部は、前記変換係数を逆変換して予測誤差を取得してもよい。前記イントラ予測部は、前記ブロックに対してイントラ予測を行ってもよい。前記インター予測部は、前記ブロックに対してインター予測を行ってもよい。前記フィルタ部は、前記イントラ予測又は前記インター予測によって得られる予測画像と前記予測誤差とを用いて生成される再構成画像にフィルタを適用してもよい。

20

【0043】

また、例えば、前記復号装置は、複数のピクチャを含む動画画像を復号する復号装置であってもよい。

【0044】

そして、エントロピー復号部は、可変長復号処理において、1つのスライスが複数のデータアクセス領域に分割されたビットストリームを復号する際に、前記ビットストリーム上における前記複数のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセット信号を、ヘッダ領域に含まれるか否かを指定する信号を、シーケンスヘッダ、ピクチャヘッダまたは拡張補助領域から読み解き、前記1つ以上のオフセット信号を、前記ヘッダ領域から復号すると指定された場合、前記1つ以上のオフセット信号をヘッダ領域から復号し、1つ以上の前記オフセット信号として復号される複数のオフセット信号のうち連続する2つのオフセット信号の間に挿入されている値が0でない1ビット以上の信号を復号し、前記1つ以上のオフセット信号を、前記ヘッダ領域から復号しないと指定された場合、1つのスライスを複数の前記データアクセス領域に分割する場合でも、前記オフセット信号を前記ビットストリームから復号しなくてもよい。

30

【0045】

さらに、これらの包括的又は具体的な態様は、システム、装置、方法、集積回路、コンピュータプログラム、又は、コンピュータ読み取り可能なCD-ROMなどの非一時的な記録媒体で実現されてもよく、システム、装置、方法、集積回路、コンピュータプログラム、及び、記録媒体の任意な組み合わせで実現されてもよい。

40

【0046】

以下、実施の形態について図面を参照しながら具体的に説明する。なお、以下で説明する実施の形態は、いずれも包括的または具体的な例を示すものである。以下の実施の形態で示される数値、形状、材料、構成要素、構成要素の配置位置及び接続形態、ステップ、ステップの関係及び順序などは、一例であり、請求の範囲を限定する主旨ではない。

【0047】

以下では、符号化装置および復号化装置の実施の形態を説明する。実施の形態は、本開示の各態様で説明する処理および／または構成を適用可能な符号化装置および復号化装置

50

の例である。処理および／または構成は、実施の形態とは異なる符号化装置および復号化装置においても実施可能である。例えば、実施の形態に対して適用される処理および／または構成に関して、例えば以下のいずれかを実施してもよい。

【0048】

(1) 本開示の各態様で説明する実施の形態の符号化装置または復号装置の複数の構成要素のうちいずれかは、本開示の各態様のいずれかで説明する他の構成要素に置き換えまたは組み合わせられてもよい。

【0049】

(2) 実施の形態の符号化装置または復号装置において、当該符号化装置または復号装置の複数の構成要素のうち一部の構成要素によって行われる機能または処理に、機能または処理の追加、置き換え、削除などの任意の変更がなされてもよい。例えば、いずれかの機能または処理は、本開示の各態様のいずれかで説明する他の機能または処理に、置き換えまたは組み合わせられてもよい。

【0050】

(3) 実施の形態の符号化装置または復号装置が実施する方法において、当該方法に含まれる複数の処理のうち一部の処理について、追加、置き換えおよび削除などの任意の変更がなされてもよい。例えば、方法におけるいずれかの処理は、本開示の各態様のいずれかで説明する他の処理に、置き換えまたは組み合わせられてもよい。

【0051】

(4) 実施の形態の符号化装置または復号装置を構成する複数の構成要素のうち一部の構成要素は、本開示の各態様のいずれかで説明する構成要素と組み合わせられてもよいし、本開示の各態様のいずれかで説明する機能の一部を備える構成要素と組み合わせられてもよいし、本開示の各態様で説明する構成要素が実施する処理の一部を実施する構成要素と組み合わせられてもよい。

【0052】

(5) 実施の形態の符号化装置または復号装置の機能の一部を備える構成要素、または、実施の形態の符号化装置または復号装置の処理の一部を実施する構成要素は、本開示の各態様のいずれかで説明する構成要素と、本開示の各態様でいずれかで説明する機能の一部を備える構成要素と、または、本開示の各態様のいずれかで説明する処理の一部を実施する構成要素と組み合わせまたは置き換えられてもよい。

【0053】

(6) 実施の形態の符号化装置または復号装置が実施する方法において、当該方法に含まれる複数の処理のいずれかは、本開示の各態様のいずれかで説明する処理に、または、同様のいずれかの処理に、置き換えまたは組み合わせられてもよい。

【0054】

(7) 実施の形態の符号化装置または復号装置が実施する方法に含まれる複数の処理のうち一部の処理は、本開示の各態様のいずれかで説明する処理と組み合わせられてもよい。

【0055】

(8) 本開示の各態様で説明する処理および／または構成の実施の仕方は、実施の形態の符号化装置または復号装置に限定されるものではない。例えば、処理および／または構成は、実施の形態において開示する動画像符号化または動画像復号とは異なる目的で利用される装置において実施されてもよい。

【0056】

[符号化装置]

まず、実施の形態に係る符号化装置を説明する。図1は、実施の形態に係る符号化装置100の機能構成を示すブロック図である。符号化装置100は、動画像をブロック単位で符号化する動画像符号化装置である。

【0057】

図1に示すように、符号化装置100は、画像をブロック単位で符号化する装置であっ

10

20

30

40

50

て、分割部 102 と、減算部 104 と、変換部 106 と、量子化部 108 と、エントロピー符号化部 110 と、逆量子化部 112 と、逆変換部 114 と、加算部 116 と、ブロックメモリ 118 と、ループフィルタ部 120 と、フレームメモリ 122 と、イントラ予測部 124 と、インター予測部 126 と、予測制御部 128 と、を備える。

【0058】

符号化装置 100 は、例えば、汎用プロセッサ及びメモリにより実現される。この場合、メモリに格納されたソフトウェアプログラムがプロセッサにより実行されたときに、プロセッサは、分割部 102、減算部 104、変換部 106、量子化部 108、エントロピー符号化部 110、逆量子化部 112、逆変換部 114、加算部 116、ループフィルタ部 120、イントラ予測部 124、インター予測部 126 及び予測制御部 128 として機能する。また、符号化装置 100 は、分割部 102、減算部 104、変換部 106、量子化部 108、エントロピー符号化部 110、逆量子化部 112、逆変換部 114、加算部 116、ループフィルタ部 120、イントラ予測部 124、インター予測部 126 及び予測制御部 128 に対応する専用の 1 以上の電子回路として実現されてもよい。

【0059】

以下に、符号化装置 100 の全体的な処理の流れを説明した後に、符号化装置 100 に含まれる各構成要素について説明する。

【0060】

〔符号化処理の全体フロー〕

図 2 は、符号化装置 100 による全体的な符号化処理の一例を示すフローチャートである。

【0061】

まず、符号化装置 100 の分割部 102 は、動画画である入力画像に含まれる各ピクチャを複数の固定サイズのブロック（例えば、 128×128 画素）に分割する（ステップ Sa__1）。そして、分割部 102 は、その固定サイズのブロックに対して分割パターン（ブロック形状ともいう）を選択する（ステップ Sa__2）。つまり、分割部 102 は、固定サイズのブロックを、その選択された分割パターンを構成する複数のブロックに、さらに分割する。そして、符号化装置 100 は、その複数のブロックのそれぞれについて、そのブロック（すなわち符号化対象ブロック）に対してステップ Sa__3～Sa__9 の処理を行う。

【0062】

つまり、イントラ予測部 124、インター予測部 126 および予測制御部 128 の全てまたは一部からなる予測処理部は、符号化対象ブロック（カレントブロックともいう）の予測信号（予測ブロックともいう）を生成する（ステップ Sa__3）。

【0063】

次に、減算部 104 は、符号化対象ブロックと予測ブロックとの差分を予測残差（差分ブロックともいう）として生成する（ステップ Sa__4）。

【0064】

次に、変換部 106 および量子化部 108 は、その差分ブロックに対して変換および量子化を行うことによって、複数の量子化係数を生成する（ステップ Sa__5）。なお、複数の量子化係数からなるブロックを係数ブロックともいう。

【0065】

次に、エントロピー符号化部 110 は、その係数ブロックと、予測信号の生成に関する予測パラメータとに対して符号化（具体的にはエントロピー符号化）を行うことによって、符号化信号を生成する（ステップ Sa__6）。なお、符号化信号は、符号化ビットストリーム、圧縮ビットストリーム、またはストリームともいう。

【0066】

次に、逆量子化部 112 および逆変換部 114 は、係数ブロックに対して逆量子化および逆変換を行うことによって、複数の予測残差（すなわち差分ブロック）を復元する（ステップ Sa__7）。

【0067】

次に、加算部116は、その復元された差分ブロックに予測ブロックを加算することによってカレントブロックを再構成画像（再構成ブロックまたは復号画像ブロックともいう）に再構成する（ステップSa__8）。これにより、再構成画像が生成される。

【0068】

この再構成画像が生成されると、ループフィルタ部120は、その再構成画像に対してフィルタリングを必要に応じて行う（ステップSa__9）。

【0069】

そして、符号化装置100は、ピクチャ全体の符号化が完了したか否かを判定し（ステップSa__10）、完了していないと判定する場合（ステップSa__10のNo）、ステップSa__2からの処理を繰り返し実行する。

10

【0070】

なお、上述の例では、符号化装置100は、固定サイズのブロックに対して1つの分割パターンを選択し、その分割パターンにしたがって各ブロックの符号化を行うが、複数の分割パターンのそれぞれにしたがって各ブロックの符号化を行ってもよい。この場合には、符号化装置100は、複数の分割パターンのそれぞれに対するコストを評価し、例えば最も小さいコストの分割パターンにしたがった符号化によって得られる符号化信号を、出力される符号化信号として選択してもよい。

【0071】

図示されているように、これらのステップSa__1～Sa__10の処理は、符号化装置100によってシーケンシャルに行われる。あるいは、それらの処理のうちの一部の複数の処理が並列に行われてもよく、それらの処理の順番の入れ替え等が行われてもよい。

20

【0072】

〔分割部〕

分割部102は、入力動画画像に含まれる各ピクチャを複数のブロックに分割し、各ブロックを減算部104に出力する。例えば、分割部102は、まず、ピクチャを固定サイズ（例えば128×128）のブロックに分割する。他の固定ブロックサイズが採用されてもよい。この固定サイズのブロックは、符号化ツリーユニット（CTU）と呼ばれることがある。そして、分割部102は、例えば再帰的な四分木（quad tree）及び／又は二分木（binary tree）ブロック分割に基づいて、固定サイズのブロックの各々を可変サイズ（例えば64×64以下）のブロックに分割する。すなわち、分割部102は、分割パターンを選択する。この可変サイズのブロックは、符号化ユニット（CU）、予測ユニット（PU）あるいは変換ユニット（TU）と呼ばれることがある。なお、種々の処理例では、CU、PU及びTUは区別される必要はなく、ピクチャ内の一部又はすべてのブロックがCU、PU、TUの処理単位となってもよい。

30

【0073】

図3は、実施の形態におけるブロック分割の一例を示す概念図である。図3において、実線は四分木ブロック分割によるブロック境界を表し、破線は二分木ブロック分割によるブロック境界を表す。

【0074】

ここでは、ブロック10は、128×128画素の正方形ブロック（128×128ブロック）である。この128×128ブロック10は、まず、4つの正方形の64×64ブロックに分割される（四分木ブロック分割）。

40

【0075】

左上の64×64ブロックは、さらに2つの矩形の32×64ブロックに垂直に分割され、左の32×64ブロックはさらに2つの矩形の16×64ブロックに垂直に分割される（二分木ブロック分割）。その結果、左上の64×64ブロックは、2つの16×64ブロック11、12と、32×64ブロック13とに分割される。

【0076】

右上の64×64ブロックは、2つの矩形の64×32ブロック14、15に水平に分

50

割される（二分木ブロック分割）。

【0077】

左下の64×64ブロックは、4つの正方形の32×32ブロックに分割される（四分木ブロック分割）。4つの32×32ブロックのうち左上のブロック及び右下のブロックはさらに分割される。左上の32×32ブロックは、2つの矩形の16×32ブロックに垂直に分割され、右の16×32ブロックはさらに2つの16×16ブロックに水平に分割される（二分木ブロック分割）。右下の32×32ブロックは、2つの32×16ブロックに水平に分割される（二分木ブロック分割）。その結果、左下の64×64ブロックは、16×32ブロック16と、2つの16×16ブロック17、18と、2つの32×32ブロック19、20と、2つの32×16ブロック21、22とに分割される。

10

【0078】

右下の64×64ブロック23は分割されない。

【0079】

以上のように、図3では、ブロック10は、再帰的な四分木及び二分木ブロック分割に基づいて、13個の可変サイズのブロック11～23に分割される。このような分割は、QTBT（quad-tree plus binary tree）分割と呼ばれることがある。

【0080】

なお、図3では、1つのブロックが4つ又は2つのブロックに分割されていたが（四分木又は二分木ブロック分割）、分割はこれらに限定されない。例えば、1つのブロックが3つのブロックに分割されてもよい（三分木ブロック分割）。このような三分木ブロック分割を含む分割は、MBT（multi-type tree）分割と呼ばれることがある。

20

【0081】

〔ピクチャの構成 スライス／タイル〕

ピクチャを並列にデコードするために、ピクチャはスライス単位またはタイル単位で構成される場合がある。スライス単位またはタイル単位からなるピクチャは、分割部102によって構成されてもよい。

【0082】

スライスは、ピクチャを構成する基本的な符号化の単位である。ピクチャは、例えば1つ以上のスライスから構成される。また、スライスは、1つ以上の連続するCTU（Coding Tree Unit）からなる。

30

【0083】

図4Aは、スライスの構成の一例を示す概念図である。例えば、ピクチャは、11×8個のCTUを含み、かつ、4つのスライス（スライス1～4）に分割される。スライス1は、16個のCTUからなり、スライス2は、21個のCTUからなり、スライス3は、29個のCTUからなり、スライス4は、22個のCTUからなる。ここで、ピクチャ内の各CTUは、いずれかのスライスに属する。スライスの形状は、ピクチャを水平方向に分割した形になる。スライスの境界は、画面端である必要はなく、画面内のCTUの境界のうちどこであってもよい。スライスの中のCTUの処理順（符号化順または復号順）は、例えばラスタ・スキャン順である。また、スライスは、ヘッダ情報と符号化データを含む。ヘッダ情報には、スライスの先頭のCTUアドレス、スライス・タイプなどそのスライスの特徴が記述されてもよい。

40

【0084】

タイルは、ピクチャを構成する矩形領域の単位である。各タイルにはTileIdと呼ばれる番号がラスタ・スキャン順に割り振られてもよい。

【0085】

図4Bは、タイルの構成の一例を示す概念図である。例えば、ピクチャは、11×8個のCTUを含み、かつ、4つの矩形領域のタイル（タイル1～4）に分割される。タイルが使用される場合、タイルが使用されない場合と比べてCTUの処理順が変更される。タイルが使用されない場合、ピクチャ内の複数のCTUはラスタ・スキャン順に処理される

50

。タイルが使用される場合には、複数のタイルのそれぞれにおいて、少なくとも1つのCTUがラスタ・スキャン順に処理される。例えば、図4Bに示すように、タイル1に含まれる複数のCTUの処理順は、タイル1の1行目左端からタイル1の1行目右端まで向かい、次に、タイル1の2行目左端からタイル1の2行目右端まで向かう順である。

【0086】

なお、1つのタイルは、1つ以上のスライスを含む場合があり、1つのスライスは、1つ以上のタイルを含む場合がある。

【0087】

〔減算部〕

減算部104は、分割部102から入力され、分割部102によって分割されたブロック単位で、原信号（原サンプル）から予測信号（以下に示す予測制御部128から入力される予測サンプル）を減算する。つまり、減算部104は、符号化対象ブロック（以下、カレントブロックという）の予測誤差（残差ともいう）を算出する。そして、減算部104は、算出された予測誤差（残差）を変換部106に出力する。

【0088】

原信号は、符号化装置100の入力信号であり、動画像を構成する各ピクチャの画像を表す信号（例えば輝度（luma）信号及び2つの色差（chroma）信号）である。以下において、画像を表す信号をサンプルということもある。

【0089】

〔変換部〕

変換部106は、空間領域の予測誤差を周波数領域の変換係数に変換し、変換係数を量子化部108に出力する。具体的には、変換部106は、例えば空間領域の予測誤差に対して所定の離散コサイン変換（DCT）又は離散サイン変換（DST）を行う。所定のDCT又はDSTは、予め定められていてもよい。

【0090】

なお、変換部106は、複数の変換タイプの中から適応的に変換タイプを選択し、選択された変換タイプに対応する変換基底関数（transform basis function）を用いて、予測誤差を変換係数に変換してもよい。このような変換は、EMT（explicit multiple core transform）又はAMT（adaptive multiple transform）と呼ばれることがある。

【0091】

複数の変換タイプは、例えば、DCT-II、DCT-V、DCT-VIII、DST-I及びDST-VIIを含む。図5Aは、変換タイプ例に対応する変換基底関数を示す表である。図5AにおいてNは入力画素の数を示す。これらの複数の変換タイプの中からの変換タイプの選択は、例えば、予測の種類（イントラ予測及びインター予測）に依存してもよいし、イントラ予測モードに依存してもよい。

【0092】

このようなEMT又はAMTを適用するか否かを示す情報（例えばEMTフラグまたはAMTフラグと呼ばれる）及び選択された変換タイプを示す情報は、通常、CUレベルで信号化される。なお、これらの情報の信号化は、CUレベルに限定される必要はなく、他のレベル（例えば、ビットシーケンスレベル、ピクチャレベル、スライスレベル、タイルレベル又はCTUレベル）であってもよい。

【0093】

また、変換部106は、変換係数（変換結果）を再変換してもよい。このような再変換は、AST（adaptive secondary transform）又はNSST（non-separable secondary transform）と呼ばれることがある。例えば、変換部106は、イントラ予測誤差に対応する変換係数のブロックに含まれるサブブロック（例えば4x4サブブロック）ごとに再変換を行う。NSSTを適用するか否かを示す情報及びNSSTに用いられる変換行列に関する情報は、通常、CUレベルで信号化される。なお、これらの情報の信号化は、CUレベルに限定される必要は

10

20

30

40

50

なく、他のレベル（例えば、シーケンスレベル、ピクチャレベル、スライスレベル、タイ
ルレベル又はCTUレベル）であってもよい。

【0094】

変換部106には、Separableな変換と、Non-Separableな変換
とが適用されてもよい。Separableな変換とは、入力の次元の数だけ方向ごとに
分離して複数回変換を行う方式であり、Non-Separableな変換とは、入力が
多次元であった際に2つ以上の次元をまとめて1次元とみなして、まとめて変換を行う方
式である。

【0095】

例えば、Non-Separableな変換の一例として、入力が4×4のブロックで
あった場合にはそれを16個の要素を持ったひとつの配列とみなし、その配列に対して1
6×16の変換行列で変換処理を行うようなものが挙げられる。

【0096】

また、Non-Separableな変換のさらなる例では、4×4の入力ブロックを
16個の要素を持ったひとつの配列とみなした後に、その配列に対してGivens回転
を複数回行うような変換（Hypercube Givens Transform）が行
われてもよい。

【0097】

変換部106での変換では、CU内の領域に応じて周波数領域に変換する基底のタイ
プを切替えることもできる。一例として、SVT（Spatially Varying T
ransform）がある。SVTでは、図5Bに示すように、水平あるいは垂直方向に
CUを2等分し、いずれか一方の領域のみ周波数領域への変換を行う。変換基底のタイ
プは領域毎に設定でき、例えば、DST7とDCT8が用いられる。本例ではCU内の2つ
の領域のうち、どちらか一方のみ変換を行い、もう一方は変換を行わないが、2つの領域
共に変換してもよい。また、分割方法も2等分だけでなく、4等分、あるいは分割を示す
情報を別途符号化してCU分割と同様にシグナリングするなど、より柔軟にすることもで
きる。なお、SVTは、SBT（Sub-block Transform）と呼ぶこと
もある。

【0098】

〔量子化部〕

量子化部108は、変換部106から出力された変換係数を量子化する。具体的には、
量子化部108は、カレントブロックの変換係数を所定の走査順序で走査し、走査された
変換係数に対応する量子化パラメータ（QP）に基づいて当該変換係数を量子化する。そ
して、量子化部108は、カレントブロックの量子化された変換係数（以下、量子化係数
という）をエントロピー符号化部110及び逆量子化部112に出力する。所定の走査順
序は、予め定められていてもよい。

【0099】

所定の走査順序は、変換係数の量子化／逆量子化のための順序である。例えば、所定の
走査順序は、周波数の昇順（低周波から高周波の順）又は降順（高周波から低周波の順）
で定義されてもよい。

【0100】

量子化パラメータ（QP）とは、量子化ステップ（量子化幅）を定義するパラメータで
ある。例えば、量子化パラメータの値が増加すれば量子化ステップも増加する。つまり、
量子化パラメータの値が増加すれば量子化誤差が増大する。

【0101】

また、量子化には、量子化マトリックスが使用される場合がある。例えば、4×4およ
び8×8などの周波数変換サイズと、イントラ予測およびインター予測などの予測モード
と、輝度および色差などの画素成分とに対応して数種類の量子化マトリックスが使われる
場合がある。なお、量子化とは、所定の間隔でサンプリングした値を所定のレベルに対
応づけてデジタル化することをいい、この技術分野では、丸め、ラウンディング、スケーリ

10

20

30

40

50

ングといった他の表現を用いて参照されてもよいし、丸め、ラウンディング、スケーリングを採用してもよい。所定の間隔及びレベルは、予め定められていてもよい。

【0102】

量子化マトリックスを使用する方法として、符号化装置側で直接設定された量子化マトリックスを使用する方法と、デフォルトの量子化マトリックス（デフォルトマトリックス）を使用する方法とがある。符号化装置側では、量子化マトリックスを直接設定することにより、画像の特徴に応じた量子化マトリックスを設定することができる。しかし、この場合、量子化マトリックスの符号化によって、符号量が増加するというデメリットがある。

【0103】

一方、量子化マトリックスを使用せず、高域成分の係数も低域成分の係数も同じように量子化する方法もある。なお、この方法は、係数が全て同じ値である量子化マトリックス（フラットなマトリックス）を用いる方法に等しい。

【0104】

量子化マトリックスは、例えば、SPS（シーケンスパラメータセット：Sequence Parameter Set）またはPPS（ピクチャパラメータセット：Picture Parameter Set）で指定されてもよい。SPSは、シーケンスに対して用いられるパラメータを含み、PPSは、ピクチャに対して用いられるパラメータを含む。SPSとPPSとは、単にパラメータセットと呼ばれる場合がある。

【0105】

〔エントロピー符号化部〕

エントロピー符号化部110は、量子化部108から入力された量子化係数に基づいて符号化信号（符号化ビットストリーム）を生成する。具体的には、エントロピー符号化部110は、例えば、量子化係数を二値化し、二値信号を算術符号化し、圧縮されたビットストリームまたはシーケンスを出力する。

【0106】

〔逆量子化部〕

逆量子化部112は、量子化部108から入力された量子化係数を逆量子化する。具体的には、逆量子化部112は、カレントブロックの量子化係数を所定の走査順序で逆量子化する。そして、逆量子化部112は、カレントブロックの逆量子化された変換係数を逆変換部114に出力する。所定の走査順序は、予め定められていてもよい。

【0107】

〔逆変換部〕

逆変換部114は、逆量子化部112から入力された変換係数を逆変換することにより予測誤差（残差）を復元する。具体的には、逆変換部114は、変換係数に対して、変換部106による変換に対応する逆変換を行うことにより、カレントブロックの予測誤差を復元する。そして、逆変換部114は、復元された予測誤差を加算部116に出力する。

【0108】

なお、復元された予測誤差は、通常、量子化により情報が失われているので、減算部104が算出した予測誤差と一致しない。すなわち、復元された予測誤差には、通常、量子化誤差が含まれている。

【0109】

〔加算部〕

加算部116は、逆変換部114から入力された予測誤差と予測制御部128から入力された予測サンプルとを加算することによりカレントブロックを再構成する。そして、加算部116は、再構成されたブロックをブロックメモリ118及びループフィルタ部120に出力する。再構成ブロックは、ローカル復号ブロックと呼ばれることもある。

【0110】

〔ブロックメモリ〕

ブロックメモリ118は、例えば、イントラ予測で参照されるブロックであって符号化対象ピクチャ（カレントピクチャという）内のブロックを格納するための記憶部である。

10

20

30

40

50

具体的には、ブロックメモリ 118 は、加算部 116 から出力された再構成ブロックを格納する。

【0111】

〔フレームメモリ〕

フレームメモリ 122 は、例えば、インター予測に用いられる参照ピクチャを格納するための記憶部であり、フレームバッファと呼ばれることもある。具体的には、フレームメモリ 122 は、ループフィルタ部 120 によってフィルタされた再構成ブロックを格納する。

【0112】

〔ループフィルタ部〕

ループフィルタ部 120 は、加算部 116 によって再構成されたブロックにループフィルタを施し、フィルタされた再構成ブロックをフレームメモリ 122 に出力する。ループフィルタとは、符号化ループ内で用いられるフィルタ（インループフィルタ）であり、例えば、デブロッキング・フィルタ（DF または DBF）、サンプルアダプティブオフセット（SAO）及びアダプティブループフィルタ（ALF）などを含む。

【0113】

ALF では、符号化歪みを除去するための最小二乗誤差フィルタが適用され、例えばカレントブロック内の 2×2 サブブロックごとに、局所的な勾配（gradient）の方向及び活性度（activity）に基づいて複数のフィルタの中から選択された 1 つのフィルタが適用される。

【0114】

具体的には、まず、サブブロック（例えば 2×2 サブブロック）が複数のクラス（例えば 15 又は 25 クラス）に分類される。サブブロックの分類は、勾配の方向及び活性度に基づいて行われる。例えば、勾配の方向値 D （例えば $0 \sim 2$ 又は $0 \sim 4$ ）と勾配の活性値 A （例えば $0 \sim 4$ ）とを用いて分類値 C （例えば $C = 5D + A$ ）が算出される。そして、分類値 C に基づいて、サブブロックが複数のクラスに分類される。

【0115】

勾配の方向値 D は、例えば、複数の方向（例えば水平、垂直及び 2 つの対角方向）の勾配を比較することにより導出される。また、勾配の活性値 A は、例えば、複数の方向の勾配を加算し、加算結果を量子化することにより導出される。

【0116】

このような分類の結果に基づいて、複数のフィルタの中からサブブロックのためのフィルタが決定される。

【0117】

ALF で用いられるフィルタの形状としては例えば円対称形状が利用される。図 6A ～ 図 6C は、ALF で用いられるフィルタの形状の複数の例を示す図である。図 6A は、 5×5 ダイヤモンド形状フィルタを示し、図 6B は、 7×7 ダイヤモンド形状フィルタを示し、図 6C は、 9×9 ダイヤモンド形状フィルタを示す。フィルタの形状を示す情報は、通常、ピクチャレベルで信号化される。なお、フィルタの形状を示す情報の信号化は、ピクチャレベルに限定される必要はなく、他のレベル（例えば、シーケンスレベル、スライスレベル、タイルレベル、CTU レベル又は CU レベル）であってもよい。

【0118】

ALF のオン／オフは、例えば、ピクチャレベル又は CU レベルで決定されてもよい。例えば、輝度については CU レベルで ALF を適用するか否かが決定されてもよく、色差についてはピクチャレベルで ALF を適用するか否かが決定されてもよい。ALF のオン／オフを示す情報は、通常、ピクチャレベル又は CU レベルで信号化される。なお、ALF のオン／オフを示す情報の信号化は、ピクチャレベル又は CU レベルに限定される必要はなく、他のレベル（例えば、シーケンスレベル、スライスレベル、タイルレベル又は CTU レベル）であってもよい。

【0119】

10

20

30

40

50

選択可能な複数のフィルタ（例えば１５又は２５までのフィルタ）の係数セットは、通常、ピクチャレベルで信号化される。なお、係数セットの信号化は、ピクチャレベルに限定される必要はなく、他のレベル（例えば、シーケンスレベル、スライスレベル、タイルレベル、ＣＴＵレベル、ＣＵレベル又はサブブロックレベル）であってもよい。

【０１２０】

〔ループフィルタ部 > デブロッキング・フィルタ〕

デブロッキング・フィルタでは、ループフィルタ部１２０は、再構成画像のブロック境界にフィルタ処理を行うことによって、そのブロック境界に生じる歪みを減少させる。

【０１２１】

図７は、デブロッキング・フィルタとして機能するループフィルタ部１２０の詳細な構成の一例を示すブロック図である。

【０１２２】

ループフィルタ部１２０は、境界判定部１２０１、フィルタ判定部１２０３と、フィルタ処理部１２０５と、処理判定部１２０８と、フィルタ特性決定部１２０７と、スイッチ１２０２、１２０４および１２０６とを備える。

【０１２３】

境界判定部１２０１は、デブロッキング・フィルタ処理される画素（すなわち対象画素）がブロック境界付近に存在しているか否かを判定する。そして、境界判定部１２０１は、その判定結果をスイッチ１２０２および処理判定部１２０８に出力する。

【０１２４】

スイッチ１２０２は、対象画素がブロック境界付近に存在していると境界判定部１２０１によって判定された場合には、フィルタ処理前の画像を、スイッチ１２０４に出力する。逆に、スイッチ１２０２は、境界判定部１２０１によって対象画素がブロック境界付近に存在していないと判定された場合には、フィルタ処理前の画像をスイッチ１２０６に出力する。

【０１２５】

フィルタ判定部１２０３は、対象画素の周辺にある少なくとも１つの周辺画素の画素値に基づいて、対象画素に対してデブロッキング・フィルタ処理を行うか否かを判定する。そして、フィルタ判定部１２０３は、その判定結果をスイッチ１２０４および処理判定部１２０８に出力する。

【０１２６】

スイッチ１２０４は、対象画素にデブロッキング・フィルタ処理を行うとフィルタ判定部１２０３によって判定された場合には、スイッチ１２０２を介して取得したフィルタ処理前の画像を、フィルタ処理部１２０５に出力する。逆に、スイッチ１２０４は、対象画素にデブロッキング・フィルタ処理を行わないとフィルタ判定部１２０３によって判定された場合には、スイッチ１２０２を介して取得したフィルタ処理前の画像をスイッチ１２０６に出力する。

【０１２７】

フィルタ処理部１２０５は、スイッチ１２０２および１２０４を介してフィルタ処理前の画像を取得した場合には、フィルタ特性決定部１２０７によって決定されたフィルタ特性を有するデブロッキング・フィルタ処理を、対象画素に対して実行する。そして、フィルタ処理部１２０５は、そのフィルタ処理後の画素をスイッチ１２０６に出力する。

【０１２８】

スイッチ１２０６は、処理判定部１２０８による制御に応じて、デブロッキング・フィルタ処理されていない画素と、フィルタ処理部１２０５によってデブロッキング・フィルタ処理された画素とを選択的に出力する。

【０１２９】

処理判定部１２０８は、境界判定部１２０１およびフィルタ判定部１２０３のそれぞれの判定結果に基づいて、スイッチ１２０６を制御する。つまり、処理判定部１２０８は、対象画素がブロック境界付近に存在していると境界判定部１２０１によって判定され、か

10

20

30

40

50

つ、対象画素にデブロッキング・フィルタ処理を行うとフィルタ判定部 1203 によって判定された場合には、デブロッキング・フィルタ処理された画素をスイッチ 1206 から出力させる。また、上述の場合以外では、処理判定部 1208 は、デブロッキング・フィルタ処理されていない画素をスイッチ 1206 から出力させる。このような画素の出力が繰り返し行われることによって、フィルタ処理後の画像がスイッチ 1206 から出力される。

【0130】

図 8 は、ブロック境界に対して対称なフィルタ特性を有するデブロッキング・フィルタの例を示す概念図である。

【0131】

デブロッキング・フィルタ処理では、例えば、画素値と量子化パラメータを用いて、特性の異なる 2 つのデブロッキング・フィルタ、すなわちストロングフィルタおよびウィークフィルタのうちの何れか 1 つが選択される。ストロングフィルタでは、図 8 に示すように、ブロック境界を挟んで画素 $p_0 \sim p_2$ と、画素 $q_0 \sim q_2$ とが存在する場合、画素 $q_0 \sim q_2$ のそれぞれの画素値は、例えば以下の式に示す演算を行うことによって、画素値 $q'_0 \sim q'_2$ に変更される。

【0132】

$$q'_0 = (p_1 + 2 \times p_0 + 2 \times q_0 + 2 \times q_1 + q_2 + 4) / 8$$

$$q'_1 = (p_0 + q_0 + q_1 + q_2 + 2) / 4$$

$$q'_2 = (p_0 + q_0 + q_1 + 3 \times q_2 + 2 \times q_3 + 4) / 8$$

【0133】

なお、上述の式において、 $p_0 \sim p_2$ および $q_0 \sim q_2$ は、画素 $p_0 \sim p_2$ および画素 $q_0 \sim q_2$ のそれぞれの画素値である。また、 q_3 は、画素 q_2 にブロック境界と反対側に隣接する画素 q_3 の画素値である。また、上述の各式の右辺において、デブロッキング・フィルタ処理に用いられる各画素の画素値に乗算される係数が、フィルタ係数である。

【0134】

さらに、デブロッキング・フィルタ処理では、演算後の画素値が閾値を超えて設定されないように、クリップ処理が行われてもよい。このクリップ処理では、上述の式による演算後の画素値は、量子化パラメータから決定される閾値を用いて、「演算対象画素値 $\pm 2 \times$ 閾値」にクリップされる。これにより、過度な平滑化を防ぐことができる。

【0135】

図 9 は、デブロッキング・フィルタ処理が行われるブロック境界を説明するための概念図である。図 10 は、Bs 値の一例を示す概念図である。

【0136】

デブロッキング・フィルタ処理が行われるブロック境界は、例えば、図 9 で示すような 8×8 画素ブロックの PU (Prediction Unit) または TU (Transform Unit) の境界である。デブロッキング・フィルタ処理は、4 行または 4 列を単位に行われ得る。まず、図 9 に示すブロック P およびブロック Q に対して、図 10 のように Bs (Boundary Strength) 値が決定される。

【0137】

図 10 の Bs 値にしたがい、同一の画像に属するブロック境界であっても、異なる強さのデブロッキング・フィルタ処理を行うか否かが決定される。色差信号に対するデブロッキング・フィルタ処理は、Bs 値が 2 の場合に行われる。輝度信号に対するデブロッキング・フィルタ処理は、Bs 値が 1 以上であって、所定の条件が満たされた場合に行われる。所定の条件は、予め定められていてもよい。なお、Bs 値の判定条件は図 10 に示したものに限定されず、他のパラメータに基づいて決定されてもよい。

【0138】

[予測処理部 (イントラ予測部・インター予測部・予測制御部)]

図 11 は、符号化装置 100 の予測処理部で行われる処理の一例を示すフローチャートである。なお、予測処理部は、イントラ予測部 124、インター予測部 126、および予

10

20

30

40

50

測制御部 128 の全てまたは一部の構成要素からなる。

【0139】

予測処理部は、カレントブロックの予測画像を生成する（ステップ S b _ 1）。この予測画像は、予測信号または予測ブロックともいう。なお、予測信号には、例えばイントラ予測信号またはインター予測信号がある。具体的には、予測処理部は、予測ブロックの生成、差分ブロックの生成、係数ブロックの生成、差分ブロックの復元、および復号画像ブロックの生成が行われることによって既に得られている再構成画像を用いて、カレントブロックの予測画像を生成する。

【0140】

再構成画像は、例えば、参照ピクチャの画像であってもよいし、カレントブロックを含むピクチャであるカレントピクチャ内の符号化済みのブロックの画像であってもよい。カレントピクチャ内の符号化済みのブロックは、例えばカレントブロックの隣接ブロックである。

10

【0141】

図 12 は、符号化装置 100 の予測処理部で行われる処理の他の例を示すフローチャートである。

【0142】

予測処理部は、第 1 の方式で予測画像を生成し（ステップ S c _ 1 a）、第 2 の方式で予測画像を生成し（ステップ S c _ 1 b）、第 3 の方式で予測画像を生成する（ステップ S c _ 1 c）。第 1 の方式、第 2 の方式、および第 3 の方式は、予測画像を生成するための互いに異なる方式であって、それぞれ例えば、インター予測方式、イントラ予測方式、および、それら以外の予測方式であってもよい。これらの予測方式では、上述の再構成画像を用いてもよい。

20

【0143】

次に、予測処理部は、ステップ S c _ 1 a、S c _ 1 b、および S c _ 1 c で生成された複数の予測画像のうちの何れか 1 つを選択する（ステップ S c _ 2）。この予測画像の選択、すなわち最終的な予測画像を得るための方式またはモードの選択は、生成された各予測画像に対するコストを算出し、そのコストに基づいて行われてもよい。または、その予測画像の選択は、符号化の処理に用いられるパラメータに基づいて行われてもよい。符号化装置 100 は、その選択された予測画像、方式またはモードを特定するための情報を符号化信号（符号化ビットストリームともいう）に信号化してもよい。その情報は、例えばフラグなどであってもよい。これにより、復号装置は、その情報に基づいて、符号化装置 100 において選択された方式またはモードにしたがって予測画像を生成することができる。なお、図 12 に示す例では、予測処理部は、各方式で予測画像を生成した後に、何れかの予測画像を選択する。しかし、予測処理部は、それらの予測画像を生成する前に、上述の符号化の処理に用いられるパラメータに基づいて、方式またはモードを選択し、その方式またはモードにしたがって予測画像を生成してもよい。

30

【0144】

例えば、第 1 の方式および第 2 の方式は、それぞれイントラ予測およびインター予測であって、予測処理部は、これらの予測方式にしたがって生成される予測画像から、カレントブロックに対する最終的な予測画像を選択してもよい。

40

【0145】

図 13 は、符号化装置 100 の予測処理部で行われる処理の他の例を示すフローチャートである。

【0146】

まず、予測処理部は、イントラ予測によって予測画像を生成し（ステップ S d _ 1 a）、インター予測によって予測画像を生成する（ステップ S d _ 1 b）。なお、イントラ予測によって生成された予測画像を、イントラ予測画像ともいい、インター予測によって生成された予測画像を、インター予測画像ともいう。

【0147】

50

次に、予測処理部は、イントラ予測画像およびインター予測画像のそれぞれを評価する（ステップS d__2）。この評価には、コストが用いられてもよい。つまり、予測処理部は、イントラ予測画像およびインター予測画像のそれぞれのコストCを算出する。このコストCは、R-D最適化モデルの式、例えば、 $C = D + \lambda \times R$ によって算出され得る。この式において、Dは、予測画像の符号化歪であって、例えば、カレントブロックの画素値と予測画像の画素値との差分絶対値和などによって表される。また、Rは、予測画像の発生符号量であって、具体的には、予測画像を生成するための動き情報などの符号化に必要な符号量などである。また、 λ は、例えばラグランジュの未定乗数である。

【0148】

そして、予測処理部は、イントラ予測画像およびインター予測画像から、最も小さいコストCが算出された予測画像を、カレントブロックの最終的な予測画像として選択する（ステップS d__3）。つまり、カレントブロックの予測画像を生成するための予測方式またはモードが選択される。

【0149】

[イントラ予測部]

イントラ予測部124は、ブロックメモリ118に格納されたカレントピクチャ内のブロックを参照してカレントブロックのイントラ予測（画面内予測ともいう）を行うことで、予測信号（イントラ予測信号）を生成する。具体的には、イントラ予測部124は、カレントブロックに隣接するブロックのサンプル（例えば輝度値、色差値）を参照してイントラ予測を行うことでイントラ予測信号を生成し、イントラ予測信号を予測制御部128

【0150】

例えば、イントラ予測部124は、規定の複数のイントラ予測モードのうちの1つを用いてイントラ予測を行う。複数のイントラ予測モードは、通常、1以上の非方向性予測モードと、複数の方向性予測モードと、を含む。規定の複数のモードは、予め規定されていてもよい。

【0151】

1以上の非方向性予測モードは、例えばH.265/H.265規格で規定されたPlanar予測モード及びDC予測モードを含む。

【0152】

複数の方向性予測モードは、例えばH.265/H.265規格で規定された33方向の予測モードを含む。なお、複数の方向性予測モードは、33方向に加えてさらに32方向の予測モード（合計で65個の方向性予測モード）を含んでもよい。図14は、イントラ予測において用いられ得る全67個のイントラ予測モード（2個の非方向性予測モード及び65個の方向性予測モード）を示す概念図である。実線矢印は、H.265/H.265規格で規定された33方向を表し、破線矢印は、追加された32方向を表す（2個の非方向性予測モードは図14には図示されていない）。

【0153】

種々の処理例では、色差ブロックのイントラ予測において、輝度ブロックが参照されてもよい。つまり、カレントブロックの輝度成分に基づいて、カレントブロックの色差成分が予測されてもよい。このようなイントラ予測は、CCLM（cross-component linear model）予測と呼ばれることがある。このような輝度ブロックを参照する色差ブロックのイントラ予測モード（例えばCCLMモードと呼ばれる）は、色差ブロックのイントラ予測モードの1つとして加えられてもよい。

【0154】

イントラ予測部124は、水平／垂直方向の参照画素の勾配に基づいてイントラ予測後の画素値を補正してもよい。このような補正をともしないイントラ予測は、PDPC（position dependent intra prediction combination）と呼ばれることがある。PDPCの適用の有無を示す情報（例えばPDPCフラグと呼ばれる）は、通常、CUレベルで信号化される。なお、この情報の信号化は、C

10

20

30

40

50

Ｕレベルに限定される必要はなく、他のレベル（例えば、シーケンスレベル、ピクチャレベル、スライスレベル、タイルレベル又はＣＴＵレベル）であってもよい。

【０１５５】

〔インター予測部〕

インター予測部１２６は、フレームメモリ１２２に格納された参照ピクチャであってカレントピクチャとは異なる参照ピクチャを参照してカレントブロックのインター予測（画面間予測ともいう）を行うことで、予測信号（インター予測信号）を生成する。インター予測は、カレントブロック又はカレントブロック内のカレントサブブロック（例えば４×４ブロック）の単位で行われる。例えば、インター予測部１２６は、カレントブロック又はカレントサブブロックについて参照ピクチャ内で動き探索（motion estimation）を行い、そのカレントブロック又はカレントサブブロックに最も一致する参照ブロック又はサブブロックを見つける。そして、インター予測部１２６は、参照ブロック又はサブブロックからカレントブロック又はサブブロックへの動き又は変化を補償する動き情報（例えば動きベクトル）を取得する。インター予測部１２６は、その動き情報に基づいて、動き補償（または動き予測）を行い、カレントブロック又はサブブロックのインター予測信号を生成する。インター予測部１２６は、生成されたインター予測信号を予測制御部１２８に出力する。

【０１５６】

動き補償に用いられた動き情報は、多様な形態でインター予測信号として信号化されてもよい。例えば、動きベクトルが信号化されてもよい。他の例として、動きベクトルと予測動きベクトル（motion vector predictor）との差分が信号化されてもよい。

【０１５７】

〔インター予測の基本フロー〕

図１５は、インター予測の基本的な流れの一例を示すフローチャートである。

【０１５８】

インター予測部１２６は、まず、予測画像を生成する（ステップＳｅ＿１～Ｓｅ＿３）。次に、減算部１０４は、カレントブロックと予測画像との差分を予測残差として生成する（ステップＳｅ＿４）。

【０１５９】

ここで、インター予測部１２６は、予測画像の生成では、カレントブロックの動きベクトル（ＭＶ）の決定（ステップＳｅ＿１およびＳｅ＿２）と、動き補償（ステップＳｅ＿３）とを行うことによって、その予測画像を生成する。また、インター予測部１２６は、ＭＶの決定では、候補動きベクトル（候補ＭＶ）の選択（ステップＳｅ＿１）と、ＭＶの導出（ステップＳｅ＿２）とを行うことによって、そのＭＶを決定する。候補ＭＶの選択は、例えば、候補ＭＶリストから少なくとも１つの候補ＭＶを選択することによって行われる。また、ＭＶの導出では、インター予測部１２６は、少なくとも１つの候補ＭＶから、さらに少なくとも１つの候補ＭＶを選択することによって、その選択された少なくとも１つの候補ＭＶを、カレントブロックのＭＶとして決定してもよい。あるいは、インター予測部１２６は、その選択された少なくとも１つの候補ＭＶのそれぞれについて、その候補ＭＶで指示される参照ピクチャの領域を探索することによって、カレントブロックのＭＶを決定してもよい。なお、この参照ピクチャの領域を探索することを、動き探索（motion estimation）と称してもよい。

【０１６０】

また、上述の例では、ステップＳｅ＿１～Ｓｅ＿３は、インター予測部１２６によって行われるが、例えばステップＳｅ＿１またはステップＳｅ＿２などの処理は、符号化装置１００に含まれる他の構成要素によって行われてもよい。

【０１６１】

〔動きベクトルの導出のフロー〕

図１６は、動きベクトル導出の一例を示すフローチャートである。

【0162】

インター予測部126は、動き情報（例えばMV）を符号化するモードで、カレントブロックのMVを導出する。この場合、例えば動き情報が予測パラメータとして符号化されて、信号化される。つまり、符号化された動き情報が、符号化信号（符号化ビットストリームともいう）に含まれる。

【0163】

あるいは、インター予測部126は、動き情報を符号化しないモードでMVを導出する。この場合には、動き情報は、符号化信号に含まれない。

【0164】

ここで、MV導出のモードには、後述のノーマルインターモード、マージモード、FRUCモードおよびアフィンモードなどがある。これらのモードのうち、動き情報を符号化するモードには、ノーマルインターモード、マージモード、およびアフィンモード（具体的には、アフィンインターモードおよびアフィンマージモード）などがある。なお、動き情報には、MVだけでなく、後述の予測動きベクトル選択情報が含まれてもよい。また、動き情報を符号化しないモードには、FRUCモードなどがある。インター予測部126は、これらの複数のモードから、カレントブロックのMVを導出するためのモードを選択し、その選択されたモードを用いてカレントブロックのMVを導出する。

【0165】

図17は、動きベクトル導出の他の例を示すフローチャートである。

【0166】

インター予測部126は、差分MVを符号化するモードで、カレントブロックのMVを導出する。この場合、例えば差分MVが予測パラメータとして符号化されて、信号化される。つまり、符号化された差分MVが、符号化信号に含まれる。この差分MVは、カレントブロックのMVと、その予測MVとの差である。

【0167】

あるいは、インター予測部126は、差分MVを符号化しないモードでMVを導出する。この場合には、符号化された差分MVは、符号化信号に含まれない。

【0168】

ここで、上述のようにMVの導出のモードには、後述のノーマルインター、マージモード、FRUCモードおよびアフィンモードなどがある。これらのモードのうち、差分MVを符号化するモードには、ノーマルインターモードおよびアフィンモード（具体的には、アフィンインターモード）などがある。また、差分MVを符号化しないモードには、FRUCモード、マージモードおよびアフィンモード（具体的には、アフィンマージモード）などがある。インター予測部126は、これらの複数のモードから、カレントブロックのMVを導出するためのモードを選択し、その選択されたモードを用いてカレントブロックのMVを導出する。

【0169】

〔動きベクトルの導出のフロー〕

図18は、動きベクトル導出の他の例を示すフローチャートである。MV導出のモード、すなわちインター予測モードには、複数のモードがあり、大きく分けて、差分MVを符号化するモードと、差分動きベクトルを符号化しないモードとがある。差分MVを符号化しないモードには、マージモード、FRUCモード、およびアフィンモード（具体的には、アフィンマージモード）がある。これらのモードの詳細については、後述するが、簡単には、マージモードは、周辺の符号化済みブロックから動きベクトルを選択することによって、カレントブロックのMVを導出するモードであり、FRUCモードは、符号化済み領域間で探索を行うことによって、カレントブロックのMVを導出するモードである。また、アフィンモードは、アフィン変換を想定して、カレントブロックを構成する複数のサブブロックそれぞれの動きベクトルを、カレントブロックのMVとして導出するモードである。

【0170】

具体的には、図示されるように、インター予測部126は、インター予測モード情報が0を示す場合（Sf__1で0）、マージモードにより動きベクトルを導出する（Sf__2）。また、インター予測部126は、インター予測モード情報が1を示す場合（Sf__1で1）、F R U Cモードにより動きベクトルを導出する（Sf__3）。また、インター予測部126は、インター予測モード情報が2を示す場合（Sf__1で2）、アフィンモード（具体的には、アフィンマージモード）により動きベクトルを導出する（Sf__4）。また、インター予測部126は、インター予測モード情報が3を示す場合（Sf__1で3）、差分MVを符号化するモード（例えば、ノーマルインターモード）により動きベクトルを導出する（Sf__5）。

【0171】

〔MV導出 > ノーマルインターモード〕

ノーマルインターモードは、候補MVによって示される参照ピクチャの領域から、カレントブロックの画像に類似するブロックに基づいて、カレントブロックのMVを導出するインター予測モードである。また、このノーマルインターモードでは、差分MVが符号化される。

【0172】

図19は、ノーマルインターモードによるインター予測の例を示すフローチャートである。

【0173】

インター予測部126は、まず、時間的または空間的にカレントブロックの周囲にある複数の符号化済みブロックのMVなどの情報に基づいて、そのカレントブロックに対して複数の候補MVを取得する（ステップSg__1）。つまり、インター予測部126は、候補MVリストを作成する。

【0174】

次に、インター予測部126は、ステップSg__1で取得された複数の候補MVの中から、N個（Nは2以上の整数）の候補MVのそれぞれを予測動きベクトル候補（予測MV候補ともいう）として、所定の優先順位に従って抽出する（ステップSg__2）。なお、その優先順位は、N個の候補MVのそれぞれに対して予め定められていてもよい。

【0175】

次に、インター予測部126は、そのN個の予測動きベクトル候補の中から1つの予測動きベクトル候補を、カレントブロックの予測動きベクトル（予測MVともいう）として選択する（ステップSg__3）。このとき、インター予測部126は、選択された予測動きベクトルを識別するための予測動きベクトル選択情報をストリームに符号化する。なお、ストリームは、上述の符号化信号または符号化ビットストリームである。

【0176】

次に、インター予測部126は、符号化済み参照ピクチャを参照し、カレントブロックのMVを導出する（ステップSg__4）。このとき、インター予測部126は、さらに、その導出されたMVと予測動きベクトルとの差分値を差分MVとしてストリームに符号化する。なお、符号化済み参照ピクチャは、符号化後に再構成された複数のブロックからなるピクチャである。

【0177】

最後に、インター予測部126は、その導出されたMVと符号化済み参照ピクチャとを用いてカレントブロックに対して動き補償を行くことにより、そのカレントブロックの予測画像を生成する（ステップSg__5）。なお、予測画像は、上述のインター予測信号である。

【0178】

また、符号化信号に含まれる、予測画像の生成に用いられたインター予測モード（上述の例ではノーマルインターモード）を示す情報は、例えば予測パラメータとして符号化される。

【0179】

10

20

30

40

50

なお、候補MVリストは、他のモードに用いられるリストと共通に用いられてもよい。また、候補MVリストに関する処理を、他のモードに用いられるリストに関する処理に適用してもよい。この候補MVリストに関する処理は、例えば、候補MVリストからの候補MVの抽出もしくは選択、候補MVの並び替え、または、候補MVの削除などである。

【0180】

〔MV導出 > マージモード〕

マージモードは、候補MVリストから候補MVをカレントブロックのMVとして選択することによって、そのMVを導出するインター予測モードである。

【0181】

図20は、マージモードによるインター予測の例を示すフローチャートである。

10

【0182】

インター予測部126は、まず、時間的または空間的にカレントブロックの周囲にある複数の符号化済みブロックのMVなどの情報に基づいて、そのカレントブロックに対して複数の候補MVを取得する（ステップSh__1）。つまり、インター予測部126は、候補MVリストを作成する。

【0183】

次に、インター予測部126は、ステップSh__1で取得された複数の候補MVの中から1つの候補MVを選択することによって、カレントブロックのMVを導出する（ステップSh__2）。このとき、インター予測部126は、選択された候補MVを識別するためのMV選択情報をストリームに符号化する。

20

【0184】

最後に、インター予測部126は、その導出されたMVと符号化済み参照ピクチャとを用いてカレントブロックに対して動き補償を行ことにより、そのカレントブロックの予測画像を生成する（ステップSh__3）。

【0185】

また、符号化信号に含められる、予測画像の生成に用いられたインター予測モード（上述の例ではマージモード）を示す情報は、例えば予測パラメータとして符号化される。

【0186】

図21は、マージモードによるカレントピクチャの動きベクトル導出処理の一例を説明するための概念図である。

30

【0187】

まず、予測MVの候補を登録した予測MVリストを生成する。予測MVの候補としては、対象ブロックの空間的に周辺に位置する複数の符号化済みブロックが持つMVである空間隣接予測MV、符号化済み参照ピクチャにおける対象ブロックの位置を投影した近辺のブロックが持つMVである時間隣接予測MV、空間隣接予測MVと時間隣接予測MVのMV値を組み合わせて生成したMVである結合予測MV、および値がゼロのMVであるゼロ予測MV等がある。

【0188】

次に、予測MVリストに登録されている複数の予測MVの中から1つの予測MVを選択することで、対象ブロックのMVとして決定する。

40

【0189】

さらに、可変長符号化部では、どの予測MVを選択したかを示す信号であるmerge__idxをストリームに記述して符号化する。

【0190】

なお、図21で説明した予測MVリストに登録する予測MVは一例であり、図中の個数とは異なる個数であったり、図中の予測MVの一部の種類を含まない構成であったり、図中の予測MVの種類以外の予測MVを追加した構成であったりしてもよい。

【0191】

マージモードにより導出した対象ブロックのMVを用いて、後述するDMVR（decoder motion vector refinement）処理を行うことによって

50

最終的なMVを決定してもよい。

【0192】

なお、予測MVの候補は、上述の候補MVであり、予測MVリストは、上述の候補MVリストである。また、候補MVリストを、候補リストと称してもよい。また、merge_idxは、MV選択情報である。

【0193】

〔MV導出 > FRUCモード〕

動き情報は符号化装置側から信号化されずに、復号装置側で導出されてもよい。なお、上述のように、H. 265/HEVC規格で規定されたマージモードが用いられてもよい。また例えば、復号装置側で動き探索を行うことにより動き情報が導出されてもよい。実施の形態において、復号装置側では、カレントブロックの画素値を用いずに動き探索が行われる。

【0194】

ここで、復号装置側で動き探索を行うモードについて説明する。この復号装置側で動き探索を行うモードは、PMMVD (pattern matched motion vector derivation) モード又はFRUC (frame rate up-conversion) モードと呼ばれることがある。

【0195】

フローチャートの形式でFRUC処理の一例を図22に示す。まず、カレントブロックに空間的又は時間的に隣接する符号化済みブロックの動きベクトルを参照して、各々が予測動きベクトル(MV)を有する複数の候補のリスト(すなわち、候補MVリストであって、マージリストと共通であってもよい)が生成される(ステップSi_1)。次に、候補MVリストに登録されている複数の候補MVの中からベスト候補MVを選択する(ステップSi_2)。例えば、候補MVリストに含まれる各候補MVの評価値が算出され、評価値に基づいて1つの候補MVが選択される。そして、選択された候補の動きベクトルに基づいて、カレントブロックのための動きベクトルが導出される(ステップSi_4)。具体的には、例えば、選択された候補の動きベクトル(ベスト候補MV)がそのままカレントブロックのための動きベクトルとして導出される。また例えば、選択された候補の動きベクトルに対応する参照ピクチャ内の位置の周辺領域において、パターンマッチングを行うことにより、カレントブロックのための動きベクトルが導出されてもよい。すなわち、ベスト候補MVの周辺の領域に対して、参照ピクチャにおけるパターンマッチングおよび評価値を用いた探索を行い、さらに評価値が良い値となるMVがあった場合は、ベスト候補MVを前記MVに更新して、それをカレントブロックの最終的なMVとしてもよい。より良い評価値を有するMVへの更新を行う処理を実施しない構成とすることも可能である。

【0196】

最後に、インター予測部126は、その導出されたMVと符号化済み参照ピクチャとを用いてカレントブロックに対して動き補償を行うことにより、そのカレントブロックの予測画像を生成する(ステップSi_5)。

【0197】

サブブロック単位で処理を行う場合も全く同様の処理としてもよい。

【0198】

評価値は、種々の方法によって算出されてもよい。例えば、動きベクトルに対応する参照ピクチャ内の領域の再構成画像と、所定の領域(その領域は、例えば、以下に示すように、他の参照ピクチャの領域またはカレントピクチャの隣接ブロックの領域であってもよい)の再構成画像とを比較する。所定の領域は予め定められていてもよい。

【0199】

そして、2つの再構成画像の画素値の差分を算出して、動きベクトルの評価値に用いてもよい。なお、差分値に加えてそれ以外の情報を用いて評価値を算出してよい。

【0200】

次に、パターンマッチングの例について詳細に説明する。まず、候補MVリスト（例えばマージリスト）に含まれる1つの候補MVを、パターンマッチングによる探索のスタートポイントとして選択する。例えば、パターンマッチングとしては、第1パターンマッチング又は第2パターンマッチングが用いられ得る。第1パターンマッチング及び第2パターンマッチングは、それぞれ、バイラテラルマッチング（*b i l a t e r a l m a t c h i n g*）及びテンプレートマッチング（*t e m p l a t e m a t c h i n g*）と呼ばれることがある。

【0201】

[MV導出 > FRUC > バイラテラルマッチング]

第1パターンマッチングでは、異なる2つの参照ピクチャ内の2つのブロックであってカレントブロックの動き軌道（*m o t i o n t r a j e c t o r y*）に沿う2つのブロックの間でパターンマッチングが行われる。したがって、第1パターンマッチングでは、上述した候補の評価値の算出のための所定の領域として、カレントブロックの動き軌道に沿う他の参照ピクチャ内の領域が用いられる。所定の領域は、予め定められていてもよい。

【0202】

図23は、動き軌道に沿う2つの参照ピクチャにおける2つのブロック間での第1パターンマッチング（バイラテラルマッチング）の一例を説明するための概念図である。図23に示すように、第1パターンマッチングでは、カレントブロック（*C u r b l o c k*）の動き軌道に沿う2つのブロックであって異なる2つの参照ピクチャ（*R e f 0*、*R e f 1*）内の2つのブロックのペアの中で最もマッチするペアを探索することにより2つの動きベクトル（*M V 0*、*M V 1*）が導出される。具体的には、カレントブロックに対して、候補MVで指定された第1の符号化済み参照ピクチャ（*R e f 0*）内の指定位置における再構成画像と、前記候補MVを表示時間間隔でスケーリングした対称MVで指定された第2の符号化済み参照ピクチャ（*R e f 1*）内の指定位置における再構成画像との差分を導出し、得られた差分値を用いて評価値を算出する。複数の候補MVの中で最も評価値が良い値となる候補MVを最終MVとして選択することが可能であり、良い結果をもたらし得る。

【0203】

連続的な動き軌道の仮定の下では、2つの参照ブロックを指し示す動きベクトル（*M V 0*、*M V 1*）は、カレントピクチャ（*C u r P i c*）と2つの参照ピクチャ（*R e f 0*、*R e f 1*）との間の時間的な距離（*T D 0*、*T D 1*）に対して比例する。例えば、カレントピクチャが時間的に2つの参照ピクチャの間に位置し、カレントピクチャから2つの参照ピクチャへの時間的な距離が等しい場合、第1パターンマッチングでは、鏡映対称な双方向の動きベクトルが導出される。

【0204】

[MV導出 > FRUC > テンプレートマッチング]

第2パターンマッチング（テンプレートマッチング）では、カレントピクチャ内のテンプレート（カレントピクチャ内でカレントブロックに隣接するブロック（例えば上及び／又は左隣接ブロック））と参照ピクチャ内のブロックとの間でパターンマッチングが行われる。したがって、第2パターンマッチングでは、上述した候補の評価値の算出のための所定の領域として、カレントピクチャ内のカレントブロックに隣接するブロックが用いられる。

【0205】

図24は、カレントピクチャ内のテンプレートと参照ピクチャ内のブロックとの間でのパターンマッチング（テンプレートマッチング）の一例を説明するための概念図である。図24に示すように、第2パターンマッチングでは、カレントピクチャ（*C u r P i c*）内でカレントブロック（*C u r b l o c k*）に隣接するブロックと最もマッチするブロックを参照ピクチャ（*R e f 0*）内で探索することによりカレントブロックの動きベクトルが導出される。具体的には、カレントブロックに対して、左隣接および上隣接の両方もしくはどちらか一方の符号化済み領域の再構成画像と、候補MVで指定された符号化済

み参照ピクチャ（Ref0）内の同等位置における再構成画像との差分を導出し、得られた差分値を用いて評価値を算出し、複数の候補MVの中で最も評価値が良い値となる候補MVをベスト候補MVとして選択することが可能である。

【0206】

このようなFRUCモードを適用するか否かを示す情報（例えばFRUCフラグと呼ばれる）は、CUレベルで信号化されてもよい。また、FRUCモードが適用される場合（例えばFRUCフラグが真の場合）、適用可能なパターンマッチングの方法（第1パターンマッチング又は第2パターンマッチング）を示す情報がCUレベルで信号化されてもよい。なお、これらの情報の信号化は、CUレベルに限定される必要はなく、他のレベル（例えば、シーケンスレベル、ピクチャレベル、スライスレベル、タイルレベル、CTUレベル又はサブブロックレベル）であってもよい。

10

【0207】

〔MV導出 > アフィンモード〕

次に、複数の隣接ブロックの動きベクトルに基づいてサブブロック単位で動きベクトルを導出するアフィンモードについて説明する。このモードは、アフィン動き補償予測（affine motion compensation prediction）モードと呼ばれることがある。

【0208】

図25Aは、複数の隣接ブロックの動きベクトルに基づくサブブロック単位の動きベクトルの導出の一例を説明するための概念図である。図25Aにおいて、カレントブロックは、16の4x4サブブロックを含む。ここでは、隣接ブロックの動きベクトルに基づいてカレントブロックの左上角制御ポイントの動きベクトル v_0 が導出され、同様に、隣接サブブロックの動きベクトルに基づいてカレントブロックの右上角制御ポイントの動きベクトル v_1 が導出される。そして、以下の式（1A）により、2つの動きベクトル v_0 及び v_1 が投影されてもよく、カレントブロック内の各サブブロックの動きベクトル（ v_x ， v_y ）が導出されてもよい。

20

【0209】

〔数1〕

$$\begin{cases} v_x = \frac{(v_{1x} - v_{0x})}{w}x - \frac{(v_{1y} - v_{0y})}{w}y + v_{0x} \\ v_y = \frac{(v_{1y} - v_{0y})}{w}x - \frac{(v_{1x} - v_{0x})}{w}y + v_{0y} \end{cases} \quad (1A)$$

30

【0210】

ここで、 x 及び y は、それぞれ、サブブロックの水平位置及び垂直位置を示し、 w は、所定の重み係数を示す。所定の重み係数は、予め決定されていてもよい。

【0211】

このようなアフィンモードを示す情報（例えばアフィンフラグと呼ばれる）は、CUレベルで信号化されてもよい。なお、このアフィンモードを示す情報の信号化は、CUレベルに限定される必要はなく、他のレベル（例えば、シーケンスレベル、ピクチャレベル、スライスレベル、タイルレベル、CTUレベル又はサブブロックレベル）であってもよい。

40

【0212】

また、このようなアフィンモードでは、左上及び右上角制御ポイントの動きベクトルの導出方法が異なるいくつかのモードを含んでもよい。例えば、アフィンモードには、アフィンインター（アフィンノーマルインターともいう）モードと、アフィンマージモードの2つのモードがある。

【0213】

〔MV導出 > アフィンモード〕

図25Bは、3つの制御ポイントを有するアフィンモードにおけるサブブロック単位の

50

動きベクトルの導出の一例を説明するための概念図である。図 2 5 B において、カレントブロックは、1 6 の 4 x 4 サブブロックを含む。ここでは、隣接ブロックの動きベクトルに基づいてカレントブロックの左上角制御ポイントの動きベクトル v_0 が導出され、同様に、隣接ブロックの動きベクトルに基づいてカレントブロックの右上角制御ポイントの動きベクトル v_1 、隣接ブロックの動きベクトルに基づいてカレントブロックの左下角制御ポイントの動きベクトル v_2 が導出される。そして、以下の式 (1 B) により、3 つの動きベクトル v_0 、 v_1 及び v_2 が投影されてもよく、カレントブロック内の各サブブロックの動きベクトル (v_x , v_y) が導出されてもよい。

【0 2 1 4】

【数 2】

$$\begin{cases} v_x = \frac{(v_{1x} - v_{0x})}{w}x - \frac{(v_{2x} - v_{0x})}{h}y + v_{0x} \\ v_y = \frac{(v_{1y} - v_{0y})}{w}x - \frac{(v_{2y} - v_{0y})}{h}y + v_{0y} \end{cases} \quad (1B)$$

10

【0 2 1 5】

ここで、 x 及び y は、それぞれ、サブブロック中心の水平位置及び垂直位置を示し、 w は、カレントブロックの幅、 h は、カレントブロックの高さを示す。

【0 2 1 6】

異なる制御ポイント数 (例えば、2 つと 3 つ) のアフィンモードは、C U レベルで切り替えて信号化されてもよい。なお、C U レベルで使用しているアフィンモードの制御ポイント数を示す情報を、他のレベル (例えば、シーケンスレベル、ピクチャレベル、スライスレベル、タイルレベル、C T U レベル又はサブブロックレベル) で信号化してもよい。

20

【0 2 1 7】

また、このような 3 つの制御ポイントを有するアフィンモードでは、左上、右上及び左下角制御ポイントの動きベクトルの導出方法が異なるいくつかのモードを含んでもよい。例えば、アフィンモードには、アフィンインター (アフィンノーマルインターともいう) モードと、アフィンマージモードの 2 つのモードがある。

【0 2 1 8】

[M V 導出 > アフィンマージモード]

30

図 2 6 A、図 2 6 B および図 2 6 C は、アフィンマージモードを説明するための概念図である。

【0 2 1 9】

アフィンマージモードでは、図 2 6 A に示すように、例えば、カレントブロックに隣接する符号化済みブロック A (左)、ブロック B (上)、ブロック C (右上)、ブロック D (左下) およびブロック E (左上) のうち、アフィンモードで符号化されたブロックに対応する複数の動きベクトルに基づいて、カレントブロックの制御ポイントのそれぞれの予測動きベクトルが算出される。具体的には、符号化済みブロック A (左)、ブロック B (上)、ブロック C (右上)、ブロック D (左下) およびブロック E (左上) の順序でこれらのブロックが検査され、アフィンモードで符号化された最初の有効なブロックが特定される。この特定されたブロックに対応する複数の動きベクトルに基づいて、カレントブロックの制御ポイントの予測動きベクトルが算出される。

40

【0 2 2 0】

例えば、図 2 6 B に示すように、カレントブロックの左に隣接するブロック A が 2 つの制御ポイントを有するアフィンモードで符号化されている場合は、ブロック A を含む符号化済みブロックの左上角および右上角の位置に投影した動きベクトル v_3 および v_4 が導出される。そして、導出された動きベクトル v_3 および v_4 から、カレントブロックの左上角の制御ポイントの予測動きベクトル v_0 と、右上角の制御ポイントの予測動きベクトル v_1 が算出される。

【0 2 2 1】

50

例えば、図 2 6 C に示すように、カレントブロックの左に隣接するブロック A が 3 つの制御ポイントを有するアフィンモードで符号化されている場合は、ブロック A を含む符号化済みブロックの左上角、右上角および左下角の位置に投影した動きベクトル v_3 、 v_4 および v_5 が導出される。そして、導出された動きベクトル v_3 、 v_4 および v_5 から、カレントブロックの左上角の制御ポイントの予測動きベクトル v_0 と、右上角の制御ポイントの予測動きベクトル v_1 と、左下角の制御ポイントの予測動きベクトル v_2 が算出される。

【0 2 2 2】

なお、後述する図 2 9 のステップ S j _ 1 におけるカレントブロックの制御ポイントのそれぞれの予測動きベクトルの導出に、この予測動きベクトル導出方法を用いてもよい。

10

【0 2 2 3】

図 2 7 は、アフィンマージモードの一例を示すフローチャートである。

【0 2 2 4】

アフィンマージモードでは、図示されるように、まず、インター予測部 1 2 6 は、カレントブロックの制御ポイントのそれぞれの予測 MV を導出する（ステップ S k _ 1）。制御ポイントは、図 2 5 A に示すように、カレントブロックの左上角および右上角のポイント、或いは図 2 5 B に示すように、カレントブロックの左上角、右上角および左下角のポイントである。

【0 2 2 5】

つまり、インター予測部 1 2 6 は、図 2 6 A に示すように、符号化済みブロック A（左）、ブロック B（上）、ブロック C（右上）、ブロック D（左下）およびブロック E（左上）の順序にこれらのブロックを検査し、アフィンモードで符号化された最初の有効なブロックを特定する。

20

【0 2 2 6】

そして、ブロック A が特定されブロック A が 2 つの制御ポイントを有する場合、図 2 6 B に示すように、インター予測部 1 2 6 は、ブロック A を含む符号化済みブロックの左上角および右上角の動きベクトル v_3 および v_4 から、カレントブロックの左上角の制御ポイントの動きベクトル v_0 と、右上角の制御ポイントの動きベクトル v_1 とを算出する。例えば、インター予測部 1 2 6 は、符号化済みブロックの左上角および右上角の動きベクトル v_3 および v_4 を、カレントブロックに投影することによって、カレントブロックの左上角の制御ポイントの予測動きベクトル v_0 と、右上角の制御ポイントの予測動きベクトル v_1 とを算出する。

30

【0 2 2 7】

或いは、ブロック A が特定されブロック A が 3 つの制御ポイントを有する場合、図 2 6 C に示すように、インター予測部 1 2 6 は、ブロック A を含む符号化済みブロックの左上角、右上角および左下角の動きベクトル v_3 、 v_4 および v_5 から、カレントブロックの左上角の制御ポイントの動きベクトル v_0 と、右上角の制御ポイントの動きベクトル v_1 、左下角の制御ポイントの動きベクトル v_2 とを算出する。例えば、インター予測部 1 2 6 は、符号化済みブロックの左上角、右上角および左下角の動きベクトル v_3 、 v_4 および v_5 を、カレントブロックに投影することによって、カレントブロックの左上角の制御ポイントの予測動きベクトル v_0 と、右上角の制御ポイントの予測動きベクトル v_1 、左下角の制御ポイントの動きベクトル v_2 とを算出する。

40

【0 2 2 8】

次に、インター予測部 1 2 6 は、カレントブロックに含まれる複数のサブブロックのそれぞれについて、動き補償を行う。すなわち、インター予測部 1 2 6 は、その複数のサブブロックのそれぞれについて、2 つの予測動きベクトル v_0 および v_1 と上述の式（1 A）、或いは 3 つの予測動きベクトル v_0 、 v_1 および v_2 と上述の式（1 B）とを用いて、そのサブブロックの動きベクトルをアフィン MV として算出する（ステップ S k _ 2）。そして、インター予測部 1 2 6 は、それらのアフィン MV および符号化済み参照ピクチャを用いてそのサブブロックに対して動き補償を行う（ステップ S k _ 3）。その結果、カ

50

レントブロックに対して動き補償が行われ、そのカレントブロックの予測画像が生成される。

【0229】

[MV導出 > アフィンインターモード]

図28Aは、2つの制御ポイントを有するアフィンインターモードを説明するための概念図である。

【0230】

このアフィンインターモードでは、図28Aに示すように、カレントブロックに隣接する符号化済みブロックA、ブロックBおよびブロックCの動きベクトルから選択された動きベクトルが、カレントブロックの左上角の制御ポイントの予測動きベクトル v_0 として用いられる。同様に、カレントブロックに隣接する符号化済みブロックDおよびブロックEの動きベクトルから選択された動きベクトルが、カレントブロックの右上角の制御ポイントの予測動きベクトル v_1 として用いられる。

10

【0231】

図28Bは、3つの制御ポイントを有するアフィンインターモードを説明するための概念図である。

【0232】

このアフィンインターモードでは、図28Bに示すように、カレントブロックに隣接する符号化済みブロックA、ブロックBおよびブロックCの動きベクトルから選択された動きベクトルが、カレントブロックの左上角の制御ポイントの予測動きベクトル v_0 として用いられる。同様に、カレントブロックに隣接する符号化済みブロックDおよびブロックEの動きベクトルから選択された動きベクトルが、カレントブロックの右上角の制御ポイントの予測動きベクトル v_1 として用いられる。更に、カレントブロックに隣接する符号化済みブロックFおよびブロックGの動きベクトルから選択された動きベクトルが、カレントブロックの左下角の制御ポイントの予測動きベクトル v_2 として用いられる。

20

【0233】

図29は、アフィンインターモードの一例を示すフローチャートである。

【0234】

図示されるように、アフィンインターモードでは、まず、インター予測部126は、カレントブロックの2つまたは3つの制御ポイントのそれぞれの予測MV (v_0 , v_1) または (v_0 , v_1 , v_2) を導出する (ステップS_j__1)。制御ポイントは、図25Aまたは図25Bに示すように、カレントブロックの左上角、右上角或いは左下角のポイントである。

30

【0235】

つまり、インター予測部126は、図28Aまたは図28Bに示すカレントブロックの各制御ポイント近傍の符号化済みブロックのうちの何れかのブロックの動きベクトルを選択することによって、カレントブロックの制御ポイントの予測動きベクトル (v_0 , v_1) または (v_0 , v_1 , v_2) を導出する。このとき、インター予測部126は、選択された2つの動きベクトルを識別するための予測動きベクトル選択情報をストリームに符号化する。

40

【0236】

例えば、インター予測部126は、カレントブロックに隣接する符号化済みブロックからどのブロックの動きベクトルを制御ポイントの予測動きベクトルとして選択するかを、コスト評価等を用いて決定し、どの予測動きベクトルを選択したかを示すフラグをビットストリームに記述してもよい。

【0237】

次に、インター予測部126は、ステップS_j__1で選択または導出された予測動きベクトルをそれぞれ更新しながら (ステップS_j__2)、動き探索を行う (ステップS_j__3およびS_j__4)。つまり、インター予測部126は、更新される予測動きベクトルに対応する各サブブロックの動きベクトルをアフィンMVとして、上述の式(1A)または

50

式(1B)を用いて算出する(ステップS_j__3)。そして、インター予測部126は、それらのアフィンMVおよび符号化済み参照ピクチャを用いて各サブブロックに対して動き補償を行う(ステップS_j__4)。その結果、インター予測部126は、動き探索ループにおいて、例えば最も小さいコストが得られる予測動きベクトルを、制御ポイントの動きベクトルとして決定する(ステップS_j__5)。このとき、インター予測部126は、さらに、その決定されたMVと予測動きベクトルとのそれぞれの差分値を差分MVとしてストリームに符号化する。

【0238】

最後に、インター予測部126は、その決定されたMVと符号化済み参照ピクチャとを用いてカレントブロックに対して動き補償を行ことにより、そのカレントブロックの予測画像を生成する(ステップS_j__6)。

【0239】

[MV導出 > アフィンインターモード]

異なる制御ポイント数(例えば、2つと3つ)のアフィンモードをCUレベルで切り替えて信号化する場合、符号化済みブロックとカレントブロックで制御ポイントの数が異なる場合がある。図30Aおよび図30Bは、符号化済みブロックとカレントブロックで制御ポイントの数が異なる場合の、制御ポイントの予測ベクトル導出方法を説明するための概念図である。

【0240】

例えば、図30Aに示すように、カレントブロックが左上角、右上角および左下角の3つの制御ポイントを有し、カレントブロックの左に隣接するブロックAが2つの制御ポイントを有するアフィンモードで符号化されている場合は、ブロックAを含む符号化済みブロックの左上角および右上角の位置に投影した動きベクトル v_3 および v_4 が導出される。そして、導出された動きベクトル v_3 および v_4 から、カレントブロックの左上角の制御ポイントの予測動きベクトル v_0 と、右上角の制御ポイントの予測動きベクトル v_1 が算出される。更に、導出された動きベクトル v_0 および v_1 から、左下角の制御ポイントの予測動きベクトル v_2 が算出される。

【0241】

例えば、図30Bに示すように、カレントブロックが左上角および右上角の2つの制御ポイントを有し、カレントブロックの左に隣接するブロックAが3つの制御ポイントを有するアフィンモードで符号化されている場合は、ブロックAを含む符号化済みブロックの左上角、右上角および左下角の位置に投影した動きベクトル v_3 、 v_4 および v_5 が導出される。そして、導出された動きベクトル v_3 、 v_4 および v_5 から、カレントブロックの左上角の制御ポイントの予測動きベクトル v_0 と、右上角の制御ポイントの予測動きベクトル v_1 が算出される。

【0242】

図29のステップS_j__1におけるカレントブロックの制御ポイントのそれぞれの予測動きベクトルの導出に、この予測動きベクトル導出方法を用いてもよい。

【0243】

[MV導出 > DMVR]

図31Aは、マージモードおよびDMVRの関係を示すフローチャートである。

【0244】

インター予測部126は、マージモードでカレントブロックの動きベクトルを導出する(ステップS_l__1)。次に、インター予測部126は、動きベクトルの探索、すなわち動き探索を行うか否かを判定する(ステップS_l__2)。ここで、インター予測部126は、動き探索を行わないと判定すると(ステップS_l__2のNo)、ステップS_l__1で導出された動きベクトルを、カレントブロックに対する最終の動きベクトルとして決定する(ステップS_l__4)。すなわち、この場合には、マージモードでカレントブロックの動きベクトルが決定される。

【0245】

一方、ステップ S l__1 で動き探索を行うと判定すると（ステップ S l__2 の Y e s）、インター予測部 1 2 6 は、ステップ S l__1 で導出された動きベクトルによって示される参照ピクチャの周辺領域を探索することによって、カレントブロックに対して最終の動きベクトルを導出する（ステップ S l__3）。すなわち、この場合には、D M V R でカレントブロックの動きベクトルが決定される。

【0 2 4 6】

図 3 1 B は、M V を決定するための D M V R 処理の一例を説明するための概念図である。

【0 2 4 7】

まず、（例えばマージモードにおいて）カレントブロックに設定された最適 M V P を、候補 M V とする。そして、候補 M V（L 0）に従って、L 0 方向の符号化済みピクチャである第 1 参照ピクチャ（L 0）から参照画素を特定する。同様に、候補 M V（L 1）に従って、L 1 方向の符号化済みピクチャである第 2 参照ピクチャ（L 1）から参照画素を特定する。これらの参照画素の平均をとることでテンプレートを生成する。

【0 2 4 8】

次に、前記テンプレートを用いて、第 1 参照ピクチャ（L 0）および第 2 参照ピクチャ（L 1）の候補 M V の周辺領域をそれぞれ探索し、コストが最小となる M V を最終的な M V として決定する。なお、コスト値は、例えば、テンプレートの各画素値と探索領域の各画素値との差分値および候補 M V 値等を用いて算出してもよい。

【0 2 4 9】

なお、典型的には、符号化装置と、後述の復号化装置とでは、ここで説明した処理の構成および動作は基本的に共通である。

【0 2 5 0】

ここで説明した処理例そのものでなくても、候補 M V の周辺を探索して最終的な M V を導出することができる処理であれば、どのような処理を用いてもよい。

【0 2 5 1】

〔動き補償 > B I O / O B M C〕

動き補償では、予測画像を生成し、その予測画像を補正するモードがある。そのモードは、例えば、後述の B I O および O B M C である。

【0 2 5 2】

図 3 2 は、予測画像の生成の一例を示すフローチャートである。

【0 2 5 3】

インター予測部 1 2 6 は、予測画像を生成し（ステップ S m__1）、例えば上述の何れかのモードによってその予測画像を補正する（ステップ S m__2）。

【0 2 5 4】

図 3 3 は、予測画像の生成の他の例を示すフローチャートである。

【0 2 5 5】

インター予測部 1 2 6 は、カレントブロックの動きベクトルを決定する（ステップ S n__1）。次に、インター予測部 1 2 6 は、予測画像を生成し（ステップ S n__2）、補正処理を行うか否かを判定する（ステップ S n__3）。ここで、インター予測部 1 2 6 は、補正処理を行うと判定すると（ステップ S n__3 の Y e s）、その予測画像を補正することによって最終的な予測画像を生成する（ステップ S n__4）。一方、インター予測部 1 2 6 は、補正処理を行わないと判定すると（ステップ S n__3 の N o）、その予測画像を補正することなく最終的な予測画像として出力する（ステップ S n__5）。

【0 2 5 6】

また、動き補償では、予測画像を生成するときに輝度を補正するモードがある。そのモードは、例えば、後述の L I C である。

【0 2 5 7】

図 3 4 は、予測画像の生成の他の例を示すフローチャートである。

【0 2 5 8】

インター予測部 1 2 6 は、カレントブロックの動きベクトルを導出する（ステップ S o

10

20

30

40

50

__1)。次に、インター予測部126は、輝度補正処理を行うか否かを判定する（ステップS o__2）。ここで、インター予測部126は、輝度補正処理を行うと判定すると（ステップS o__2のY e s）、輝度補正を行いながら予測画像を生成する（ステップS o__3）。つまり、L I Cによって予測画像が生成される。一方、インター予測部126は、輝度補正処理を行わないと判定すると（ステップS o__2のN o）、輝度補正を行うことなく通常の動き補償によって予測画像を生成する（ステップS o__4）。

【0259】

〔動き補償 > O B M C〕

動き探索により得られたカレントブロックの動き情報だけでなく、隣接ブロックの動き情報も用いて、インター予測信号が生成されてもよい。具体的には、（参照ピクチャ内の）動き探索により得られた動き情報に基づく予測信号と、（カレントピクチャ内の）隣接ブロックの動き情報に基づく予測信号と、を重み付け加算することにより、カレントブロック内のサブブロック単位でインター予測信号が生成されてもよい。このようなインター予測（動き補償）は、O B M C（o v e r l a p p e d b l o c k m o t i o n c o m p e n s a t i o n）と呼ばれることがある。

【0260】

O B M Cモードでは、O B M Cのためのサブブロックのサイズを示す情報（例えばO B M Cブロックサイズと呼ばれる）は、シーケンスレベルで信号化されてもよい。さらに、O B M Cモードを適用するか否かを示す情報（例えばO B M Cフラグと呼ばれる）は、C Uレベルで信号化されてもよい。なお、これらの情報の信号化のレベルは、シーケンスレベル及びC Uレベルに限定される必要はなく、他のレベル（例えばピクチャレベル、スライスレベル、タイルレベル、C T Uレベル又はサブブロックレベル）であってもよい。

【0261】

O B M Cモードの例について、より具体的に説明する。図35及び図36は、O B M C処理による予測画像補正処理の概要を説明するためのフローチャート及び概念図である。

【0262】

まず、図36に示すように、処理対象（カレント）ブロックに割り当てられた動きベクトル（M V）を用いて通常の動き補償による予測画像（P r e d）を取得する。図36において、矢印“M V”は参照ピクチャを指し、予測画像を得るためにカレントピクチャのカレントブロックが何を参照しているかを示している。

【0263】

次に、符号化済みの左隣接ブロックに対して既に導出された動きベクトル（M V__L）を符号化対象ブロックに適用（再利用）して予測画像（P r e d__L）を取得する。動きベクトル（M V__L）は、カレントブロックから参照ピクチャを指す矢印“M V__L”によって示される。そして、2つの予測画像P r e dとP r e d__Lとを重ね合わせることで予測画像の1回目の補正を行う。これは、隣接ブロック間の境界を混ぜ合わせる効果を有する。

【0264】

同様に、符号化済みの上隣接ブロックに対して既に導出された動きベクトル（M V__U）を符号化対象ブロックに適用（再利用）して予測画像（P r e d__U）を取得する。動きベクトル（M V__U）は、カレントブロックから参照ピクチャを指す矢印“M V__U”によって示される。そして、予測画像P r e d__Uを1回目の補正を行った予測画像（例えば、P r e dとP r e d__L）を重ね合わせることで予測画像の2回目の補正を行う。これは、隣接ブロック間の境界を混ぜ合わせる効果を有する。2回目の補正によって得られた予測画像は、隣接ブロックとの境界が混ぜ合わされた（スムージングされた）、カレントブロックの最終的な予測画像である。

【0265】

なお、上述の例は、左隣接および上隣接のブロックを用いた2パスの補正方法であるが、その補正方法は、右隣接および／または下隣接のブロックも用いた3パスまたはそれ以上のパスの補正方法であってもよい。

【0266】

なお、重ね合わせを行う領域はブロック全体の画素領域ではなく、ブロック境界近傍の一部の領域のみであってもよい。

【0267】

なお、ここでは1枚の参照ピクチャから、追加的な予測画像 $Pred_L$ および $Pred_U$ を重ね合わせることで1枚の予測画像 $Pred$ を得るためのOBMCの予測画像補正処理について説明した。しかし、複数の参照画像に基づいて予測画像が補正される場合には、同様の処理が複数の参照ピクチャのそれぞれに適用されてもよい。このような場合、複数の参照ピクチャに基づくOBMCの画像補正を行うことによって、各々の参照ピクチャから、補正された予測画像を取得した後に、その取得された複数の補正予測画像をさらに重ね合わせることで最終的な予測画像を取得する。

10

【0268】

なお、OBMCでは、対象ブロックの単位は、予測ブロック単位であっても、予測ブロックをさらに分割したサブブロック単位であってもよい。

【0269】

OBMC処理を適用するかどうかの判定の方法として、例えば、OBMC処理を適用するかどうかを示す信号である $obmc_flag$ を用いる方法がある。具体的な一例としては、符号化装置は、対象ブロックが動きの複雑な領域に属しているかどうかを判定してもよい。符号化装置は、動きの複雑な領域に属している場合は、 $obmc_flag$ として値1を設定してOBMC処理を適用して符号化を行い、動きの複雑な領域に属していない場合は、 $obmc_flag$ として値0を設定してOBMC処理を適用せずにブロックの符号化を行う。一方、復号化装置では、ストリーム（例えば圧縮シーケンス）に記述された $obmc_flag$ を復号することで、その値に応じてOBMC処理を適用するかどうかを切替えて復号を行う。

20

【0270】

インター予測部126は、上述の例では、矩形のカレントブロックに対して1つの矩形の予測画像を生成する。しかし、インター予測部126は、その矩形のカレントブロックに対して矩形と異なる形状の複数の予測画像を生成し、それらの複数の予測画像を結合することによって、最終的な矩形の予測画像を生成してもよい。矩形と異なる形状は、例えば三角形であってもよい。

30

【0271】

図37は、2つの三角形の予測画像の生成を説明するための概念図である。

【0272】

インター予測部126は、カレントブロック内の三角形の第1パーティションに対して、その第1パーティションの第1MVを用いて動き補償を行うことによって、三角形の予測画像を生成する。同様に、インター予測部126は、カレントブロック内の三角形の第2パーティションに対して、その第2パーティションの第2MVを用いて動き補償を行うことによって、三角形の予測画像を生成する。そして、インター予測部126は、これらの予測画像を結合することによって、カレントブロックと同じ矩形の予測画像を生成する。

【0273】

なお、図37に示す例では、第1パーティションおよび第2パーティションはそれぞれ三角形であるが、台形であってもよく、それぞれ互いに異なる形状であってもよい。さらに、図37に示す例では、カレントブロックが2つのパーティションから構成されているが、3つ以上のパーティションから構成されていてもよい。

40

【0274】

また、第1パーティションおよび第2パーティションは重複していてもよい。すなわち、第1パーティションおよび第2パーティションは同じ画素領域を含んでいてもよい。この場合、第1パーティションにおける予測画像と第2パーティションにおける予測画像とを用いてカレントブロックの予測画像を生成してもよい。

【0275】

50

また、この例では2つのパーティションともにインター予測で予測画像が生成される例を示したが、少なくとも1つのパーティションについてイントラ予測によって予測画像を生成してもよい。

【0276】

[動き補償 > BIO]

次に、動きベクトルを導出する方法について説明する。まず、等速直線運動を仮定したモデルに基づいて動きベクトルを導出するモードについて説明する。このモードは、BIO (bi-directional optical flow) モードと呼ばれることがある。

【0277】

図38は、等速直線運動を仮定したモデルを説明するための概念図である。図38において、 (v_x, v_y) は、速度ベクトルを示し、 τ_0 、 τ_1 は、それぞれ、カレントピクチャ (Cur Pic) と2つの参照ピクチャ (Ref0, Ref1) との間の時間的な距離を示す。 (MV_x0, MV_y0) は、参照ピクチャ Ref0 に対応する動きベクトルを示し、 (MV_x1, MV_y1) は、参照ピクチャ Ref1 に対応する動きベクトルを示す。

【0278】

このとき速度ベクトル (v_x, v_y) の等速直線運動の仮定の下では、 (MV_x0, MV_y0) 及び (MV_x1, MV_y1) は、それぞれ、 $(v_x \tau_0, v_y \tau_0)$ 及び $(-v_x \tau_1, -v_y \tau_1)$ と表され、以下のオプティカルフロー等式 (2) が採用されてもよい。

【0279】

【数3】

$$\partial I^{(k)} / \partial t + v_x \partial I^{(k)} / \partial x + v_y \partial I^{(k)} / \partial y = 0. \quad (2)$$

【0280】

ここで、 $I(k)$ は、動き補償後の参照画像 k ($k=0, 1$) の輝度値を示す。このオプティカルフロー等式は、 (i) 輝度値の時間微分と、 (ii) 水平方向の速度及び参照画像の空間勾配の水平成分の積と、 (iii) 垂直方向の速度及び参照画像の空間勾配の垂直成分の積と、の和が、ゼロと等しいことを示す。このオプティカルフロー等式とエルミート補間 (Hermite interpolation) との組み合わせに基づいて、マージリスト等から得られるブロック単位の動きベクトルが画素単位で補正されてもよい。

【0281】

なお、等速直線運動を仮定したモデルに基づく動きベクトルの導出とは異なる方法で、復号装置側で動きベクトルが導出されてもよい。例えば、複数の隣接ブロックの動きベクトルに基づいてサブブロック単位で動きベクトルが導出されてもよい。

【0282】

[動き補償 > LIC]

次に、LIC (local illumination compensation) 処理を用いて予測画像 (予測) を生成するモードの一例について説明する。

【0283】

図39は、LIC処理による輝度補正処理を用いた予測画像生成方法の一例を説明するための概念図である。

【0284】

まず、符号化済みの参照ピクチャからMVを導出して、カレントブロックに対応する参照画像を取得する。

【0285】

次に、カレントブロックに対して、参照ピクチャとカレントピクチャとで輝度値がどの

10

20

30

40

50

ように変化したかを示す情報を抽出する。この抽出は、カレントピクチャにおける符号化済み左隣接参照領域（周辺参照領域）および符号化済み上隣参照領域（周辺参照領域）の輝度画素値と、導出されたMVで指定された参照ピクチャ内の同等位置における輝度画素値とに基づいて行われる。そして、輝度値がどのように変化したかを示す情報を用いて、輝度補正パラメータを算出する。

【0286】

MVで指定された参照ピクチャ内の参照画像に対して前記輝度補正パラメータを適用する輝度補正処理を行うことで、カレントブロックに対する予測画像を生成する。

【0287】

なお、図39における前記周辺参照領域の形状は一例であり、これ以外の形状を用いてもよい。

【0288】

また、ここでは1枚の参照ピクチャから予測画像を生成する処理について説明したが、複数枚の参照ピクチャから予測画像を生成する場合も同様であり、各々の参照ピクチャから取得した参照画像に、上述と同様の方法で輝度補正処理を行ってから予測画像を生成してもよい。

【0289】

LIC処理を適用するかどうかの判定の方法として、例えば、LIC処理を適用するかどうかを示す信号である`lic_flag`を用いる方法がある。具体的な一例としては、符号化装置において、カレントブロックが、輝度変化が発生している領域に属しているかどうかを判定し、輝度変化が発生している領域に属している場合は`lic_flag`として値1を設定してLIC処理を適用して符号化を行い、輝度変化が発生している領域に属していない場合は`lic_flag`として値0を設定してLIC処理を適用せずに符号化を行う。一方、復号化装置では、ストリームに記述された`lic_flag`を復号化することで、その値に応じてLIC処理を適用するかどうかを切替えて復号を行ってもよい。

【0290】

LIC処理を適用するかどうかの判定の別の方法として、例えば、周辺ブロックでLIC処理を適用したかどうかに従って判定する方法もある。具体的な一例としては、カレントブロックがマージモードであった場合、マージモード処理におけるMVの導出の際に選択した周辺の符号化済みブロックがLIC処理を適用して符号化したかどうかを判定する。その結果に応じてLIC処理を適用するかどうかを切替えて符号化を行う。なお、この例の場合でも、同じ処理が復号装置側の処理に適用される。

【0291】

LIC処理（輝度補正処理）の態様について図39を用いて説明したが、以下、その詳細を説明する。

【0292】

まず、インター予測部126は、符号化済みピクチャである参照ピクチャから符号化対象ブロックに対応する参照画像を取得するための動きベクトルを導出する。

【0293】

次に、インター予測部126は、符号化対象ブロックに対して、左隣接および上隣接の符号化済み周辺参照領域の輝度画素値と、動きベクトルで指定された参照ピクチャ内の同等位置における輝度画素値とを用いて、参照ピクチャと符号化対象ピクチャとで輝度値がどのように変化したかを示す情報を抽出して輝度補正パラメータを算出する。例えば、符号化対象ピクチャ内の周辺参照領域内のある画素の輝度画素値を p_0 とし、当該画素と同等位置の、参照ピクチャ内の周辺参照領域内の画素の輝度画素値を p_1 とする。インター予測部126は、周辺参照領域内の複数の画素に対して、 $A \times p_1 + B = p_0$ を最適化する係数 A 及び B を輝度補正パラメータとして算出する。

【0294】

次に、インター予測部126は、動きベクトルで指定された参照ピクチャ内の参照画像に対して輝度補正パラメータを用いて輝度補正処理を行うことで、符号化対象ブロックに

10

20

30

40

50

対する予測画像を生成する。例えば、参照画像内の輝度画素値を p_2 とし、輝度補正処理後の予測画像の輝度画素値を p_3 とする。インター予測部 126 は、参照画像内の各画素に対して、 $A \times p_2 + B = p_3$ を算出することで輝度補正処理後の予測画像を生成する。

【0295】

なお、図 39 における周辺参照領域の形状は一例であり、これ以外の形状を用いてもよい。また、図 39 に示す周辺参照領域の一部が用いられてもよい。例えば、上隣接画素および左隣接画素のそれぞれから間引いた所定数の画素を含む領域を周辺参照領域として用いてもよい。また、周辺参照領域は、符号化対象ブロックに隣接する領域に限らず、符号化対象ブロックに隣接しない領域であってもよい。画素に関する所定数は、予め定められていてもよい。

【0296】

また、図 39 に示す例では、参照ピクチャ内の周辺参照領域は、符号化対象ピクチャ内の周辺参照領域から、符号化対象ピクチャの動きベクトルで指定される領域であるが、他の動きベクトルで指定される領域であってもよい。例えば、当該他の動きベクトルは、符号化対象ピクチャ内の周辺参照領域の動きベクトルであってもよい。

【0297】

なお、ここでは、符号化装置 100 における動作を説明したが、復号装置 200 における動作も典型的には同様である。

【0298】

なお、LIC 処理は輝度のみではなく、色差に適用してもよい。このとき、 Y 、 C_b 、および C_r のそれぞれに対して個別に補正パラメータを導出してもよいし、いずれかに対して共通の補正パラメータを用いてもよい。

【0299】

また、LIC 処理はサブブロック単位で適用してもよい。例えば、カレントサブブロックの周辺参照領域と、カレントサブブロックの MV で指定された参照ピクチャ内の参照サブブロックの周辺参照領域を用いて補正パラメータを導出してもよい。

【0300】

〔予測制御部〕

予測制御部 128 は、イントラ予測信号（イントラ予測部 124 から出力される信号）及びインター予測信号（インター予測部 126 から出力される信号）のいずれかを選択し、選択した信号を予測信号として減算部 104 及び加算部 116 に出力する。

【0301】

図 1 に示すように、種々の符号化装置例では、予測制御部 128 は、エントロピー符号化部 110 に入力される予測パラメータを出力してもよい。エントロピー符号化部 110 は、予測制御部 128 から入力されるその予測パラメータ、量子化部 108 から入力される量子化係数に基づいて、符号化ビットストリーム（またはシーケンス）を生成してもよい。予測パラメータは復号装置に使用されてもよい。復号装置は、符号化ビットストリームを受信して復号し、イントラ予測部 124、インター予測部 126 および予測制御部 128 において行われる予測処理と同じ処理を行ってもよい。予測パラメータは、選択予測信号（例えば、動きベクトル、予測タイプ、または、イントラ予測部 124 またはインター予測部 126 で用いられた予測モード）、または、イントラ予測部 124、インター予測部 126 および予測制御部 128 において行われる予測処理に基づく、あるいはその予測処理を示す、任意のインデックス、フラグ、もしくは値を含んでいてもよい。

【0302】

〔符号化装置の実装例〕

図 40 は、符号化装置 100 の実装例を示すブロック図である。符号化装置 100 は、プロセッサ a1 及びメモリ a2 を備える。例えば、図 1 に示された符号化装置 100 の複数の構成要素は、図 40 に示されたプロセッサ a1 及びメモリ a2 によって実装される。

【0303】

プロセッサ a1 は、情報処理を行う回路であり、メモリ a2 にアクセス可能な回路であ

10

20

30

40

50

る。例えば、プロセッサ a 1 は、動画像を符号化する専用又は汎用の電子回路である。プロセッサ a 1 は、CPU のようなプロセッサであってもよい。また、プロセッサ a 1 は、複数の電子回路の集合体であってもよい。また、例えば、プロセッサ a 1 は、図 1 等 に示された符号化装置 100 の複数の構成要素のうち、複数の構成要素の役割を果たしてもよい。

【0304】

メモリ a 2 は、プロセッサ a 1 が動画像を符号化するための情報が記憶される専用又は汎用のメモリである。メモリ a 2 は、電子回路であってもよく、プロセッサ a 1 に接続されていてもよい。また、メモリ a 2 は、プロセッサ a 1 に含まれていてもよい。また、メモリ a 2 は、複数の電子回路の集合体であってもよい。また、メモリ a 2 は、磁気ディスク又は光ディスク等であってもよいし、ストレージ又は記録媒体等と表現されてもよい。また、メモリ a 2 は、不揮発性メモリでもよいし、揮発性メモリでもよい。

10

【0305】

例えば、メモリ a 2 には、符号化される動画像が記憶されてもよいし、符号化された動画像に対応するビット列が記憶されてもよい。また、メモリ a 2 には、プロセッサ a 1 が動画像を符号化するためのプログラムが記憶されていてもよい。

【0306】

また、例えば、メモリ a 2 は、図 1 等 に示された符号化装置 100 の複数の構成要素のうち、情報を記憶するための構成要素の役割を果たしてもよい。例えば、メモリ a 2 は、図 1 に示されたブロックメモリ 118 及びフレームメモリ 122 の役割を果たしてもよい。より具体的には、メモリ a 2 には、再構成済みブロック及び再構成済みピクチャ等が記憶されてもよい。

20

【0307】

なお、符号化装置 100 において、図 1 等 に示された複数の構成要素の全てが実装されなくてもよいし、上述された複数の処理の全てが行われなくてもよい。図 1 等 に示された複数の構成要素の一部は、他の装置に含まれていてもよいし、上述された複数の処理の一部は、他の装置によって実行されてもよい。

【0308】

〔復号装置〕

次に、例えば上記の符号化装置 100 から出力された符号化信号（符号化ビットストリーム）を復号可能な復号装置について説明する。図 4 1 は、実施の形態に係る復号装置 200 の機能構成を示すブロック図である。復号装置 200 は、動画像をブロック単位で復号する動画像復号装置である。

30

【0309】

図 4 1 に示すように、復号装置 200 は、エントロピー復号部 202 と、逆量子化部 204 と、逆変換部 206 と、加算部 208 と、ブロックメモリ 210 と、ループフィルタ部 212 と、フレームメモリ 214 と、イントラ予測部 216 と、インター予測部 218 と、予測制御部 220 と、を備える。

【0310】

復号装置 200 は、例えば、汎用プロセッサ及びメモリにより実現される。この場合、メモリに格納されたソフトウェアプログラムがプロセッサにより実行されたときに、プロセッサは、エントロピー復号部 202、逆量子化部 204、逆変換部 206、加算部 208、ループフィルタ部 212、イントラ予測部 216、インター予測部 218 及び予測制御部 220 として機能する。また、復号装置 200 は、エントロピー復号部 202、逆量子化部 204、逆変換部 206、加算部 208、ループフィルタ部 212、イントラ予測部 216、インター予測部 218 及び予測制御部 220 に対応する専用の 1 以上の電子回路として実現されてもよい。

40

【0311】

以下に、復号装置 200 の全体的な処理の流れを説明した後に、復号装置 200 に含まれる各構成要素について説明する。

50

【0312】

〔復号処理の全体フロー〕

図42は、復号装置200による全体的な復号処理の一例を示すフローチャートである。

【0313】

まず、復号装置200のエントロピー復号部202は、固定サイズのブロック（例えば、 128×128 画素）の分割パターンを特定する（ステップSp__1）。この分割パターンは、符号化装置100によって選択された分割パターンである。そして、復号装置200は、その分割パターンを構成する複数のブロックのそれぞれに対してステップSp__2～Sp__6の処理を行う。

【0314】

つまり、エントロピー復号部202は、復号対象ブロック（カレントブロックともいう）の符号化された量子化係数および予測パラメータを復号（具体的にはエントロピー復号）する（ステップSp__2）。

【0315】

次に、逆量子化部204および逆変換部206は、複数の量子化係数に対して逆量子化および逆変換を行うことによって、複数の予測残差（すなわち差分ブロック）を復元する（ステップSp__3）。

【0316】

次に、イントラ予測部216、インター予測部218および予測制御部220の全てまたは一部からなる予測処理部は、カレントブロックの予測信号（予測ブロックともいう）を生成する（ステップSp__4）。

【0317】

次に、加算部208は、差分ブロックに予測ブロックを加算することによってカレントブロックを再構成画像（復号画像ブロックともいう）に再構成する（ステップSp__5）。

【0318】

そして、この再構成画像が生成されると、ループフィルタ部212は、その再構成画像に対してフィルタリングを行う（ステップSp__6）。

【0319】

そして、復号装置200は、ピクチャ全体の復号が完了したか否かを判定し（ステップSp__7）、完了していないと判定する場合（ステップSp__7のNo）、ステップSp__1からの処理を繰り返し実行する。

【0320】

図示されたように、ステップSp__1～Sp__7の処理は、復号装置200によってシークエンシャルに行われる。あるいは、それらの処理のうちの一部の複数の処理が並列に行われてもよく、順番の入れ替え等が行われてもよい。

【0321】

〔エントロピー復号部〕

エントロピー復号部202は、符号化ビットストリームをエントロピー復号する。具体的には、エントロピー復号部202は、例えば、符号化ビットストリームから二値信号に算術復号する。そして、エントロピー復号部202は、二値信号を多値化（*debinarize*）する。エントロピー復号部202は、ブロック単位で量子化係数を逆量子化部204に出力する。エントロピー復号部202は、実施の形態におけるイントラ予測部216、インター予測部218および予測制御部220に、符号化ビットストリーム（図1参照）に含まれている予測パラメータを出力してもよい。イントラ予測部216、インター予測部218および予測制御部220は、符号化装置側におけるイントラ予測部124、インター予測部126および予測制御部128で行われる処理と同じ予測処理を実行することができる。

【0322】

〔逆量子化部〕

逆量子化部204は、エントロピー復号部202からの入力である復号対象ブロック（

10

20

30

40

50

以下、カレントブロックという)の量子化係数を逆量子化する。具体的には、逆量子化部204は、カレントブロックの量子化係数の各々について、当該量子化係数に対応する量子化パラメータに基づいて当該量子化係数を逆量子化する。そして、逆量子化部204は、カレントブロックの逆量子化された量子化係数(つまり変換係数)を逆変換部206に出力する。

【0323】

〔逆変換部〕

逆変換部206は、逆量子化部204からの入力である変換係数を逆変換することにより予測誤差を復元する。

【0324】

例えば符号化ビットストリームから読み解かれた情報がEMT又はAMTを適用することを示す場合(例えばAMTフラグが真)、逆変換部206は、読み解かれた変換タイプを示す情報に基づいてカレントブロックの変換係数を逆変換する。

【0325】

また例えば、符号化ビットストリームから読み解かれた情報がNSTを適用することを示す場合、逆変換部206は、変換係数に逆再変換を適用する。

【0326】

〔加算部〕

加算部208は、逆変換部206からの入力である予測誤差と予測制御部220からの入力である予測サンプルとを加算することによりカレントブロックを再構成する。そして、加算部208は、再構成されたブロックをブロックメモリ210及びループフィルタ部212に出力する。

【0327】

〔ブロックメモリ〕

ブロックメモリ210は、イントラ予測で参照されるブロックであって復号対象ピクチャ(以下、カレントピクチャという)内のブロックを格納するための記憶部である。具体的には、ブロックメモリ210は、加算部208から出力された再構成ブロックを格納する。

【0328】

〔ループフィルタ部〕

ループフィルタ部212は、加算部208によって再構成されたブロックにループフィルタを施し、フィルタされた再構成ブロックをフレームメモリ214及び表示装置等に出力する。

【0329】

符号化ビットストリームから読み解かれたALFのオン／オフを示す情報がALFのオンを示す場合、局所的な勾配の方向及び活性度に基づいて複数のフィルタの中から1つのフィルタが選択され、選択されたフィルタが再構成ブロックに適用される。

【0330】

〔フレームメモリ〕

フレームメモリ214は、インター予測に用いられる参照ピクチャを格納するための記憶部であり、フレームバッファと呼ばれることもある。具体的には、フレームメモリ214は、ループフィルタ部212によってフィルタされた再構成ブロックを格納する。

【0331】

〔予測処理部(イントラ予測部・インター予測部・予測制御部)〕

図43は、復号装置200の予測処理部で行われる処理の一例を示すフローチャートである。なお、予測処理部は、イントラ予測部216、インター予測部218、および予測制御部220の全てまたは一部の構成要素からなる。

【0332】

予測処理部は、カレントブロックの予測画像を生成する(ステップSq_1)。この予測画像は、予測信号または予測ブロックともいう。なお、予測信号には、例えばイントラ

10

20

30

40

50

予測信号またはインター予測信号がある。具体的には、予測処理部は、予測ブロックの生成、差分ブロックの生成、係数ブロックの生成、差分ブロックの復元、および復号画像ブロックの生成が行われることによって既に得られている再構成画像を用いて、カレントブロックの予測画像を生成する。

【0333】

再構成画像は、例えば、参照ピクチャの画像であってもよいし、カレントブロックを含むピクチャであるカレントピクチャ内の復号済みのブロックの画像であってもよい。カレントピクチャ内の復号済みのブロックは、例えばカレントブロックの隣接ブロックである。

【0334】

図44は、復号装置200の予測処理部で行われる処理の他の例を示すフローチャートである。

【0335】

予測処理部は、予測画像を生成するための方式またはモードを判定する（ステップS_r__1）。例えば、この方式またはモードは、例えば予測パラメータなどに基づいて判定されてもよい。

【0336】

予測処理部は、予測画像を生成するためのモードとして第1の方式を判定した場合には、その第1の方式にしたがって予測画像を生成する（ステップS_r__2a）。また、予測処理部は、予測画像を生成するためのモードとして第2の方式を判定した場合には、その第2の方式にしたがって予測画像を生成する（ステップS_r__2b）。また、予測処理部は、予測画像を生成するためのモードとして第3の方式を判定した場合には、その第3の方式にしたがって予測画像を生成する（ステップS_r__2c）。

【0337】

第1の方式、第2の方式、および第3の方式は、予測画像を生成するための互いに異なる方式であって、それぞれ例えば、インター予測方式、イントラ予測方式、および、それら以外の予測方式であってもよい。これらの予測方式では、上述の再構成画像を用いてもよい。

【0338】

〔イントラ予測部〕

イントラ予測部216は、符号化ビットストリームから読み解かれたイントラ予測モードに基づいて、ブロックメモリ210に格納されたカレントピクチャ内のブロックを参照してイントラ予測を行うことで、予測信号（イントラ予測信号）を生成する。具体的には、イントラ予測部216は、カレントブロックに隣接するブロックのサンプル（例えば輝度値、色差値）を参照してイントラ予測を行うことでイントラ予測信号を生成し、イントラ予測信号を予測制御部220に出力する。

【0339】

なお、色差ブロックのイントラ予測において輝度ブロックを参照するイントラ予測モードが選択されている場合は、イントラ予測部216は、カレントブロックの輝度成分に基づいて、カレントブロックの色差成分を予測してもよい。

【0340】

また、符号化ビットストリームから読み解かれた情報がPDP Cの適用を示す場合、イントラ予測部216は、水平／垂直方向の参照画素の勾配に基づいてイントラ予測後の画素値を補正する。

【0341】

〔インター予測部〕

インター予測部218は、フレームメモリ214に格納された参照ピクチャを参照して、カレントブロックを予測する。予測は、カレントブロック又はカレントブロック内のサブブロック（例えば4×4ブロック）の単位で行われる。例えば、インター予測部218は、符号化ビットストリーム（例えば、エンロピー復号部202から出力される予測パラメータ）から読み解かれた動き情報（例えば動きベクトル）を用いて動き補償を行うこ

10

20

30

40

50

とでカレントブロック又はサブブロックのインター予測信号を生成し、インター予測信号を予測制御部 220 に出力する。

【0342】

符号化ビットストリームから読み解かれた情報が O B M C モードを適用することを示す場合、インター予測部 218 は、動き探索により得られたカレントブロックの動き情報だけでなく、隣接ブロックの動き情報も用いて、インター予測信号を生成する。

【0343】

また、符号化ビットストリームから読み解かれた情報が F R U C モードを適用することを示す場合、インター予測部 218 は、符号化ストリームから読み解かれたパターンマッチングの方法（バイラテラルマッチング又はテンプレートマッチング）に従って動き探索を行うことにより動き情報を導出する。そして、インター予測部 218 は、導出された動き情報を用いて動き補償（予測）を行う。

【0344】

また、インター予測部 218 は、B I O モードが適用される場合に、等速直線運動を仮定したモデルに基づいて動きベクトルを導出する。また、符号化ビットストリームから読み解かれた情報がアフィン動き補償予測モードを適用することを示す場合には、インター予測部 218 は、複数の隣接ブロックの動きベクトルに基づいてサブブロック単位で動きベクトルを導出する。

【0345】

〔MV 導出 > ノーマルインターモード〕

符号化ビットストリームから読み解かれた情報がノーマルインターモードを適用することを示す場合、インター予測部 218 は、符号化ストリームから読み解かれた情報に基づいて、MV を導出し、その MV を用いて動き補償（予測）を行う。

【0346】

図 45 は、復号装置 200 におけるノーマルインターモードによるインター予測の例を示すフローチャートである。

【0347】

復号装置 200 のインター予測部 218 は、ブロックごとに、そのブロックに対して動き補償を行う。インター予測部 218 は、時間的または空間的にカレントブロックの周囲にある複数の復号済みブロックの MV などの情報に基づいて、そのカレントブロックに対して複数の候補 MV を取得する（ステップ S s __ 1）。つまり、インター予測部 218 は、候補 MV リストを作成する。

【0348】

次に、インター予測部 218 は、ステップ S s __ 1 で取得された複数の候補 MVの中から、N 個（N は 2 以上の整数）の候補 MV のそれぞれを予測動きベクトル候補（予測 MV 候補ともいう）として、所定の優先順位に従って抽出する（ステップ S s __ 2）。なお、その優先順位は、N 個の予測 MV 候補のそれぞれに対して予め定められていてもよい。

【0349】

次に、インター予測部 218 は、入力されたストリーム（すなわち符号化ビットストリーム）から予測動きベクトル選択情報を復号し、その復号された予測動きベクトル選択情報を用いて、その N 個の予測 MV 候補の中から 1 つの予測 MV 候補を、カレントブロックの予測動きベクトル（予測 MV ともいう）として選択する（ステップ S s __ 3）。

【0350】

次に、インター予測部 218 は、入力されたストリームから差分 MV を復号し、その復号された差分 MV である差分値と、選択された予測動きベクトルとを加算することによって、カレントブロックの MV を導出する（ステップ S s __ 4）。

【0351】

最後に、インター予測部 218 は、その導出された MV と復号済み参照ピクチャとを用いてカレントブロックに対して動き補償を行うことにより、そのカレントブロックの予測画像を生成する（ステップ S s __ 5）。

10

20

30

40

50

【0352】

〔予測制御部〕

予測制御部220は、イントラ予測信号及びインター予測信号のいずれかを選択し、選択した信号を予測信号として加算部208に出力する。全体的に、復号装置側の予測制御部220、イントラ予測部216およびインター予測部218の構成、機能、および処理は、符号化装置側の予測制御部128、イントラ予測部124およびインター予測部126の構成、機能、および処理と対応していてもよい。

【0353】

〔復号装置の実装例〕

図46は、復号装置200の実装例を示すブロック図である。復号装置200は、プロセッサb1及びメモリb2を備える。例えば、図41に示された復号装置200の複数の構成要素は、図46に示されたプロセッサb1及びメモリb2によって実装される。

【0354】

プロセッサb1は、情報処理を行う回路であり、メモリb2にアクセス可能な回路である。例えば、プロセッサb1は、符号化された動画像（すなわち符号化ビットストリーム）を復号する専用又は汎用の電子回路である。プロセッサb1は、CPUのようなプロセッサであってもよい。また、プロセッサb1は、複数の電子回路の集合体であってもよい。また、例えば、プロセッサb1は、図41等々に示された復号装置200の複数の構成要素のうち、複数の構成要素の役割を果たしてもよい。

【0355】

メモリb2は、プロセッサb1が符号化ビットストリームを復号するための情報が記憶される専用又は汎用のメモリである。メモリb2は、電子回路であってもよく、プロセッサb1に接続されていてもよい。また、メモリb2は、プロセッサb1に含まれていてもよい。また、メモリb2は、複数の電子回路の集合体であってもよい。また、メモリb2は、磁気ディスク又は光ディスク等であってもよいし、ストレージ又は記録媒体等と表現されてもよい。また、メモリb2は、不揮発性メモリでもよいし、揮発性メモリでもよい。

【0356】

例えば、メモリb2には、動画像が記憶されてもよいし、符号化ビットストリームが記憶されてもよい。また、メモリb2には、プロセッサb1が符号化ビットストリームを復号するためのプログラムが記憶されていてもよい。

【0357】

また、例えば、メモリb2は、図41等々に示された復号装置200の複数の構成要素のうち、情報を記憶するための構成要素の役割を果たしてもよい。具体的には、メモリb2は、図41に示されたブロックメモリ210及びフレームメモリ214の役割を果たしてもよい。より具体的には、メモリb2には、再構成済みブロック及び再構成済みピクチャ等が記憶されてもよい。

【0358】

なお、復号装置200において、図41等々に示された複数の構成要素の全てが実装されなくてもよいし、上述された複数の処理の全てが行われなくてもよい。図41等々に示された複数の構成要素の一部は、他の装置に含まれていてもよいし、上述された複数の処理の一部は、他の装置によって実行されてもよい。

【0359】

〔各用語の定義〕

各用語は一例として、以下のような定義であってもよい。

【0360】

ピクチャは、モノクロフォーマットにおける複数の輝度サンプルの配列、又は、4:2:0、4:2:2及び4:4:4のカラーフォーマットにおける複数の輝度サンプルの配列及び複数の色差サンプルの2つの対応配列である。ピクチャは、フレーム又はフィールドであってもよい。

【0361】

フレームは、複数のサンプル行 0、2、4、・・・が生じるトップフィールド、及び、複数のサンプル行 1、3、5、・・・が生じるボトムフィールドの組成物である。

【0362】

スライスは、1つの独立スライスセグメント、及び、（もしあれば）同じアクセスユニット内の（もしあれば）次の独立スライスセグメントに先行する全ての後続の従属スライスセグメントに含まれる整数個の符号化ツリーユニットである。

【0363】

タイルは、ピクチャにおける特定のタイル列及び特定のタイル行内の複数の符号化ツリーブロックの矩形領域である。タイルは、タイルのエッジを跨ぐループフィルタが依然として適用されてもよいが、独立して復号及び符号化され得ることが意図された、フレームの矩形領域であってもよい。

【0364】

ブロックは、複数のサンプルの $M \times N$ (N 行 M 列) 配列、又は、複数の変換係数の $M \times N$ 配列である。ブロックは、1つの輝度及び2つの色差の複数の行列からなる複数の画素の正方形又は矩形の領域であってもよい。

【0365】

CTU（符号化ツリーユニット）は、3つのサンプル配列を有するピクチャの複数の輝度サンプルの符号化ツリーブロックであってもよいし、複数の色差サンプルの2つの対応符号化ツリーブロックであってもよい。あるいは、CTUは、モノクロピクチャと、3つの分離されたカラー平面及び複数のサンプルの符号化に用いられるシンタックス構造を用いて符号化されるピクチャとのいずれかの複数のサンプルの符号化ツリーブロックであってもよい。

【0366】

スーパーブロックは、1つ又は2つのモード情報ブロックを構成し、又は、再帰的に4つの 32×32 ブロックに分割され、さらに分割され得る 64×64 画素の正方形ブロックであってもよい。

【0367】

〔エントリーポイントオフセットの説明〕

まず、エントリーポイントオフセット (`entry_point_offset`) について説明する。エントリーポイントオフセットは、スライスを構成する複数のデータアクセス領域の先頭位置を示す情報である。シンタックスにおいて、エントリーポイントオフセットは、`entry_point_offset` と記載される。なお、シンタックスにおいて、エントリーポイントオフセットは、`entry_point_offset` と異なる名称で表現されてもよい。

【0368】

図47は、`entry_point_offset` を説明するための図である。図47を用いて、エントリーポイントオフセットについて、さらに詳しく説明する。スライスは、複数のデータアクセス領域に分割された状態で、ビットストリームに符号化される場合がある。例えば、データアクセス領域の例は、ブリック (`Brick`) またはタイルである。タイルは、スライスが矩形の領域に分割されたものである。また、例えば、データアクセス領域は、WPP (`Wavefront Parallel Processing`) と呼ばれる、スライスが、並列復号処理を可能とするために分割された領域であってもよい。例えば、WPPは、カレントピクチャの1CTUラインによって定義される領域であってもよい。

【0369】

復号において、復号装置200が、ビットストリームに符号化された、複数のデータアクセス領域のうちの任意のデータアクセス領域にアクセスするためには、復号装置200はアクセスの対象となるデータアクセス領域の先頭に位置するデータの、ビットストリーム上の位置を判別する必要がある。復号装置200が当該判別を行うために、符号化装置100が、ビットストリームに符号化された、複数のデータアクセス領域それぞれの先頭

10

20

30

40

50

を示すデータの位置を、スライスヘッダに記述した信号が、エントリーポイントオフセットである。

【0370】

図47に示される例においては、符号化装置100によって、スライスがブリックに分割されて符号化される例が示されている。このとき、Brick0の先頭バイトの位置は、Brick0がスライスヘッダの直後にあることにより特定が可能であるため、符号化装置100は、Brick0の先頭バイトの位置をエントリーポイントオフセットとしてスライスヘッダに記述しない。しかしながら、符号化装置100は、Brick1およびBrick2それぞれの先頭バイトの位置を、スライスヘッダに記述して符号化する。そのため、復号装置200は、Brick1およびBrick2それぞれの先頭バイトの位置を特定することができる。これにより、復号装置200は、Brick1およびBrick2それぞれに独立してアクセスすることができ、Brick1およびBrick2それぞれに対して、独立して復号処理を開始することが可能となる。

10

【0371】

次に、エントリーポイントオフセットのシンタックス構成の例について説明する。図48は、entry_point_offsetのシンタックス構成の例を示す図である。図48に示される例では、エントリーポイントオフセットに関わる情報は、符号化装置100によってスライスヘッダに記述されている。なお、エントリーポイントオフセットは、符号化装置100によって、スライスヘッダではなく、その他のヘッダ領域または拡張補助情報領域に記述されていてもよい。

20

【0372】

図48に示されるシンタックスを、符号化装置100の行う処理を通して説明する。まず、符号化装置100は、エントリーポイントオフセットを示す情報を符号化する条件が満たされているか否かを判定する。当該条件とは、データアクセス領域が処理対象スライス内に複数あるか否かであってもよい。具体的には、当該条件を満たす場合とは、処理対象スライスが複数のブリックで構成されている場合、または、処理対象スライスが、WPPが用いられる処理に対応している場合等である。

【0373】

次に、符号化装置100は、シンタックスにおいて、num_entry_point_offsetと表現される情報を符号化する。num_entry_point_offsetは、符号化装置100がスライスヘッダに記述するエントリーポイントオフセットの個数を示す情報である。具体的には、num_entry_point_offsetは、処理対象スライスを構成するブリックの個数から1を引いた個数、または、処理対象スライスを、WPPを用いて処理する際に、処理対象スライスを分割したデータ領域の個数から1を引いた個数等を示す情報である。ここで、num_entry_point_offsetは、上記以外の数値を示す情報であってもよい。

30

【0374】

なお、処理対象スライスに含まれるデータアクセス領域の個数があらかじめ判明している場合は、num_entry_point_offsetを表す信号をスライスヘッダに符号化しなくてもよい。ここで、処理対象スライスに含まれるデータアクセス領域の個数があらかじめ判明している場合とは、例えば、処理対象スライスに含まれるブリックの個数があらかじめ判明している、または、既定値である場合等である。

40

【0375】

次に、符号化装置100は、シンタックスにおいてoffset_lengthと表現される情報を符号化する。offset_lengthは、符号化装置100が処理対象スライスのスライスヘッダに記述する、エントリーポイントオフセット1つ分のビット長を指定する情報である。例えば、符号化装置100は、データアクセス領域が比較的大きなサイズになる可能性が高いときに、offset_lengthとして、長いビット長を指定し、反対に、データアクセス領域が比較的大きなサイズになる可能性が低いときには、短いビット長を指定する。そして、符号化装置100は、指定されたエントリーポイ

50

ントオフセットの個数分だけ、`entry__point__offset`の値を符号化する。
【0376】

なお、図48に示される例では、処理対象スライスのエントリーポイントオフセットに関する情報が、符号化装置100が、処理対象スライスのスライスヘッダに記述する最後の信号となるため、符号化装置100は、バイトアラインを行ってから、スライスヘッダ領域の記述を終了する。具体的には、符号化装置100がビットストリームに記述した最後の信号のビット位置がバイト境界でなかった場合、符号化装置100は、次のバイト境界に達するまで0または1の値のビットを記述する処理を行う。なお、符号化装置100がビットストリームに記述した最後の信号のビット位置がバイト境界であった場合でも、符号化装置100は、さらに8ビットの0または1の値の信号を記述して、記述された信号が次のバイト境界に達するような処理を行ってもよい。

10

【0377】

なお、図48で示されたシンタックスは一例であり、符号化装置100は、記載されている一部のシンタックスを変更したり、図48で示されたシンタックス以外のシンタックスを追加したり、シンタックスの順序を変更したり、シンタックスにおける条件判定等の処理を変更してもよい。

【0378】

〔エントリーポイントオフセットの課題〕

次に、符号化または復号において、エントリーポイントオフセットが用いられる場合に生じる問題点について、説明する。図49は、`entry__point__offset`を表す信号を符号化する方法の例を説明するための図である。図47および図48で説明したように、符号化装置100が、エントリーポイントオフセットに関する情報をスライスヘッダに記述するためには、符号化装置100がスライスヘッダを生成する前に、符号化装置100が処理対象スライスを構成する各データアクセス領域の先頭バイト位置を特定する必要がある。つまり、符号化装置100がスライスヘッダを生成する前に、符号化装置100が処理対象スライスを構成する複数のデータアクセス領域を符号化した後の符号量を、符号化装置100が特定する必要がある。

20

【0379】

しかしながら、スライスヘッダは、処理対象スライスを構成する複数のデータアクセス領域が記述されているスライスデータの前に符号化装置100によって符号化されるため、符号化装置100がスライスヘッダを生成する時点では、符号化装置100は、エントリーポイントオフセットに関する確定した情報を得ることができない。そこで、符号化装置100は、例えば、図49に示される方法で、処理対象スライスのスライスヘッダを生成する。

30

【0380】

まず、符号化装置100は、スライスヘッダを生成する際に、`entry__point__offset`を表す信号として、所定のビット長を有するダミーのビット列をスライスヘッダに記述して、符号化し、符号化したデータをストリームバッファに出力する。続いて、符号化装置100は、スライスデータの符号化を行い、符号化したデータをストリームバッファに出力する。そして、符号化装置100は、スライスデータの符号化が完了したことにより確定した`entry__point__offset`を表す信号を、`entry__point__offset`を表す信号としてすでに記述されているダミーのビット列と置き換える。このとき、符号化装置100は、処理対象スライスを構成する複数のデータアクセス領域（例えば、図49で示される`Brick0`、`Brick1`、および、`Brick2`）の符号量が確定した時点で、当該符号量から`entry__point__offset`の値を決定する。

40

【0381】

ところが、符号化装置100が図49に示される方法を用いて符号化を行った場合に、2つの問題が発生する。まず、符号化装置100が図49に示される方法を用いて符号化を行った場合に発生する第1の問題について説明する。図50は、`entry__point`

50

t__offsetに関する課題を説明するための図である。

【0382】

図48で説明されたエントリーポイントオフセットのシンタックス構成によると、entry__point__offsetを表す信号が記述されるビット列の先頭位置がバイト境界でない場合が生じることが想定される。そのため、符号化装置100は、ストリームバッファ上に格納されているスライスヘッダのうちの上述のダミーのビット列の位置を、バイト位置および1バイト中のビット位置を管理する必要がある、管理処理が複雑化するという問題が発生する。また、符号化装置100は、ストリームバッファ上に格納されたスライスヘッダを表すデータの書き換えをバイト単位でしか行うことができない。そのため、符号化装置100が上述のダミーのビット列を書き換える場合、entry__point__offsetを表す信号以外の信号まで書き換えてしまう可能性がある。よって、符号化装置100は、entry__point__offsetを表す信号以外の信号まで書き換えてしまうことを避けるために、複雑なデータ書き換え処理を行う必要に迫られるという問題が発生する。

10

【0383】

次に、符号化装置100が図49に示される方法を用いて符号化を行った場合に発生する第2の問題について説明する。図51は、entry__point__offsetに関する課題を説明するための別の図である。符号化装置100が生成するビットストリームについては、NALユニットと呼ばれる単位でデータ構造が管理される。例えば、NALユニットは、1つのスライスであり、または、1つのパラメータセットであってもよい。NALユニットが1つのスライスの場合は、NALユニットは、例えば、ヘッダ領域とデータ領域とを含む。また、NALユニットが1つのパラメータセットの場合は、パラメータセットは、例えば、SPSまたはPPSである。一般に、NALユニットの先頭バイトを復号装置200が一意に特定することができるように、符号化装置100は、NALユニットの先頭にスタートコードと呼ばれる特定のビット列を記述し、それらを符号化する。

20

【0384】

ここで、スタートコードは、例えば、図51に示される例のように、値が0の2バイトのビット列と値が1の1バイトのビット列とから構成される。一般的に、スタートコードと同一のビット列が偶然スタートコード以外のデータ領域で出現した場合、復号装置200は、本来のスタートコードを表すビット列、および、スタートコード以外の情報を表す、スタートコードと偶然同一となったビット列を見分けることができない。このような問題を解消するため、符号化装置100は、スタートコードを表すデータ領域以外のデータ領域で、偶然スタートコードと同一のビット列が出現した場合は、そのビット列の途中に、符号化情報を持たないエミュレーション防止バイトと呼ばれる1バイトのビット列を挿入する。そして、復号装置200は、復号の際に、エミュレーション防止バイトを、ビットストリームから取り除いてから、シンタックスの復号を行う。図51に示される例では、エミュレーション防止バイトは、値が3の1バイトのビット列であり、値が0の2バイトのビット列と、値が1の1バイトのビット列との間に挿入されている。

30

【0385】

図48で説明されたentry__point__offsetを表す信号が記述されるビット列の領域においても、偶然スタートコードと同一のビット列が出現する可能性があり、その場合には、符号化装置100は、エミュレーション防止バイトを挿入することが必要となる。特に、図49で説明された処理方法が用いられた場合、符号化装置100が正しい値のentry__point__offsetを上書きする際に、置き換えられる前のダミーのビット列を記述したビット列を符号化したデータ領域に対してエミュレーション防止バイトを挿入してしまうと、挿入されたエミュレーション防止バイトのバイト数だけ上書きするデータが延長され、符号化装置100は、後続する本来上書きを行ってはいけないスライスデータの先頭の領域まで、誤って上書きしてしまうことになる。

40

【0386】

[エントリーポイントオフセットのシンタックスの第1の例]

50

図52は、実施の形態における`entry__point__offset`のシンタックス構成の第1の例を示す図である。図48で説明されたシンタックス構成の例との違いは、`offset__length`の後にバイトアラインメントが行われている点と、`entry__point__offset`の後に、シンタックスにおいて`entry__point__separate__one__bit`と表現される情報が符号化されている点である。上述の点以外は、図48と共通であるため、ここでは説明を省略する。

【0387】

符号化装置100は、`offset__length`の後に、バイトアラインメントを行うことで、`offset__length`に後続する`entry__point__offset`を表す信号の記述を、常にバイト境界から開始することができる。そのため、符号化装置100は、図50で説明された課題を解決することが可能となる。なお、符号化装置100が実施するバイトアラインメントの処理は、`entry__point__offset`を表す信号の記述の後で行われるバイトアラインメントの処理と共通であってもよい、異なってもよい。

【0388】

例えば、バイトアラインメントの処理は、バイト境界に達するまで埋めるビットの値が0か1かで異なってもよいし、バイトアラインメントの処理を行う時点でビット列を記述する位置が既にバイト境界であった場合に、さらに8ビットを書き込んで次のバイト境界まで、次にビット列を記述する位置を進めるか否かが異なってもよい。なお、エントリーポイントオフセットに関する情報を表す信号が、スライスヘッダの最後のデータを表す信号でない場合でも、エントリーポイントオフセットに関する情報を表す信号の直後に、バイトアラインメントの処理を行う構成としてもよい。

【0389】

`entry__point__separate__one__bit`は、値1を持つ1ビットの信号であり、`entry__point__offset`を符号化する度に、符号化装置100によって挿入される。これにより、符号化装置100は、1つ分の`entry__point__offset`のビット長よりも長く、値0のビットが連続することを回避することが可能となる。また、スタートコードで定義された、0が連続するビット列のビット数よりも、`entry__point__offset`を表す信号のビット長が短ければ、図51で説明された課題を解決することが可能となる。なお、`entry__point__separate__one__bit`は、値が0でないビットで構成されたビット列の信号であれば、2ビット以上の信号であってもよいし、他の意味を持たせた信号に置き換えられてもよい。

【0390】

なお、図52の例では、符号化装置100が、`entry__point__separate__one__bit`を、`entry__point__offset`を表す信号の後に挿入しているが、逆に、符号化装置100が、`entry__point__separate__one__bit`を、`entry__point__offset`を表す信号の前に挿入する構成としてもよい。

【0391】

なお、図52の例では、`offset__length`を表す信号の後の、バイトアラインメントの処理と、`entry__point__offset`を表す信号の後の、`entry__point__separate__one__bit`を表す信号の符号化との、2つの処理を同時に適用した例を説明したが、符号化装置100は、いずれか一方のみを適用する構成としてもよい。

【0392】

なお、このシンタックス構成は一例であり、記載されている一部のシンタックスを変更したり、これ以外のシンタックスを追加したり、シンタックスの順番を変えたり、条件判定等の処理を変更してもよい。なお、ここではシンタックスの符号化処理を用いて説明を行ったが、復号処理においても、シンタックスに信号を符号化するか、シンタックスから

10

20

30

40

50

信号を復号するかの差異があるだけであり、符号化装置 100 と復号装置 200 とで基本的な処理方法は共通である。

【0393】

〔エン트리ポイントオフセットのシンタックスの第1の例の効果〕

図52で説明された構成によると、図49で説明された方法で符号化処理を行う際に発生する、図50および図51で説明された課題を解決することができる。よって、符号化装置100は、処理を簡略化させることが可能となり、符号化装置100の回路規模の削減および処理サイクル数の削減が実現できる可能性が高くなる。

【0394】

〔エン트리ポイントオフセットのシンタックスの第2の例〕

図53は、実施の形態における `entry__point__offset` のシンタックス構成の第2の例を示す図である。図53で説明されるシンタックス構成の第2の例の、図52で説明されたシンタックス構成の第1の例との差異は、図52で説明された例における `entry__point__offset` を表す信号が、`entry__point__offset__prefix` と `entry__point__offset__suffix` との2つの信号に分割されており、さらに、それぞれの後に、`entry__point__separate__one__bit` という信号が符号化されている点である。図53で説明されるシンタックス構成の第2の例の上記以外の構成は、図52で説明されたシンタックス構成の第1の例と共通であるため、ここでは説明を省略する。

【0395】

`entry__point__offset__prefix` を表す信号は、符号化対象とする `entry__point__offset` を表す信号の上位側の N ビットの信号であり、`entry__point__offset__suffix` は、符号化対象とする `entry__point__offset` を表す信号の下位側の M ビットの信号である。そして、 $N+M$ で表されるビット数は、`offset__length` で指定された `entry__point__offset` のビット長と等しくなる。例えば、 N をスタートコードのエミュレーションが発生する可能性のある、値0のビットが連続するビット列のビット数よりも小さいビット長とし、 M を `offset__length` で指定されたビット数から N を引いた値のビット長としてもよい。

【0396】

具体的には、 $N+M$ が32ビットであった場合、 N を22ビット、 M を10ビットとしてもよい。また、例えば、 N を `offset__length` で指定されたエン트리ポイントオフセットのビット長を2で割ったビット長とし、 M をその残りのビット長としてもよい。具体的には、 $N+M$ が27ビットであった場合、 N を14ビット、 M を13ビットとしてもよい。

【0397】

なお、図53の例では、`entry__point__offset__suffix` を表す信号と、`entry__point__offset__suffix` を表す信号の後に続く、`entry__point__separate__one__bit` を表す信号とが、常に符号化される構成としている。しかしながら、特定の条件を満たす場合は、`entry__point__offset__suffix` を表す信号と、`entry__point__offset__suffix` を表す信号との後に続く、`entry__point__separate__one__bit` を表す信号とを符号化しない構成としてもよい。

【0398】

例えば、`offset__length` で指定された `entry__point__offset` が表す信号のビット長が、スタートコードのエミュレーションが発生する可能性のある、値0のビットが連続するビット列のビット数よりも小さいビット長であった場合は、それらの信号を符号化しない構成としてもよい。ここで、想定される場合とは、例えばビット長が22以下であった場合である。

【0399】

なお、図53の例では、`entry__point__offset`を表す信号を`entry__point__offset__prefix`を表す信号と`entry__point__offset__suffix`を表す信号との2つに分割する構成としているが、`entry__point__offset`を表す信号を3つ以上に分割する構成としてもよい。その場合も分割した信号のそれぞれに対して、符号化装置100が、`entry__point__separate__one__bit`を表す信号を符号化する構成としてもよい。

【0400】

なお、図53の例では、符号化装置100が、`entry__point__separate__one__bit`を、`entry__point__offset__prefix`を表す信号と`entry__point__offset__suffix`を表す信号の後にそれぞれ挿入しているが、逆に、符号化装置100が、`entry__point__separate__one__bit`を、`entry__point__offset__prefix`を表す信号と`entry__point__offset__suffix`を表す信号の前にそれぞれ挿入する構成としてもよい。

【0401】

なお、このシンタックス構成は一例であり、符号化装置100は、記載されている一部のシンタックスを変更したり、上記以外のシンタックスを追加したり、シンタックスの順番を変えたり、条件判定等の処理を変更してもよい。なお、ここではシンタックスの符号化処理を用いて、第2の例の説明を行ったが、復号処理においてもシンタックスに信号を符号化するか、シンタックスから信号を復号するかの差異があるだけであり、符号化装置100と復号装置200とで基本的な処理方法は共通である。

【0402】

〔エン트리ポイントオフセットのシンタックスの第2の例の効果〕

図53で説明された構成によると、符号化装置100が、図49で説明された方式で符号化処理を行う際に発生する図50および図51で説明された課題について、`entry__point__offset`を表す信号に割当てられたビット長がどのような長さであった場合でも、符号化装置100は当該課題を回避することができる。よって、符号化装置100は、処理を簡略化させることが可能となり、符号化装置100の回路規模の削減および処理サイクル数の削減が実現される可能性が高まる。

【0403】

〔エン트리ポイントオフセットのシンタックスの第3の例〕

図54は、実施の形態における`entry__point__offset`のシンタックス構成の第3の例を示す図である。図48で説明された`entry__point__offset`のシンタックス構成の例との違いは、以下の通りである。図48の例では、エン트리ポイントオフセットに関する信号を符号化するか否かの判定条件として、`entry__point__offset`を符号化する条件を満たしているか否かを判定する構成であったのに対して、図54の例では、さらにSPSで指定した`present__flag`が有効か否かについての判定も行う構成となっている。図54の例で説明されたシンタックスの第3の例の構成は、上記以外は図48で説明された例のシンタックス構成の例と共通であるため、ここでは説明を省略する。

【0404】

`present__flag`は、`entry__point__offset`を符号化する条件を満たされているか否かに関わらず、符号化装置100が、`entry__point__offset`を符号化するか否かを切換え可能とする信号である。ここで、`present__flag`を表す信号は、SPS（シーケンスヘッダ）に記述される信号である。なお、`present__flag`を表す信号は、スライスよりも上位のヘッダ領域であれば、SPSではない領域に記述されてもよい。SPSではない領域とは、例えば、ピクチャヘッダ、拡張補助情報領域等である。

【0405】

これにより、例えば、データアクセス領域が処理対象スライス内に複数個あった場合で

10

20

30

40

50

も、`present_flag`によって`entry_point_offset`を表す信号を符号化しないという指定を行っている場合は、符号化装置100は、エントリーポイントオフセットに関する信号が符号化されていないストリームを生成することが可能となる。例えば、復号装置200において、データアクセス領域に独立してアクセスする必要がないシステムの場合には、符号化装置100においてエントリーポイントオフセットに関する信号を符号化しない構成とすることで、システムの簡略化が図られることが可能となる。

【0406】

なお、符号化装置100は、図54で説明された`entry_point_offset`のシンタックス構成の第3の例と、図52で説明された`entry_point_offset`のシンタックス構成の第1の例または図53で説明された`entry_point_offset`のシンタックス構成の第2の例とが組み合わせた構成でもよい。また、図54で説明されたシンタックス構成は一例であり、符号化装置100は、記載されている一部のシンタックスを変更したり、上記以外のシンタックスを追加したり、シンタックスの順番を変更したり、条件判定等の処理を変更したりしてもよい。なお、ここではシンタックスの符号化処理を用いて説明を行ったが、復号処理においてもシンタックスに信号を符号化するか、シンタックスから信号を復号するかの差異があるだけであり、基本的な処理方法は、符号化装置100と復号装置200とで共通である。

【0407】

〔エントリーポイントオフセットのシンタックスの第3の例の効果〕

図54で説明された構成によると、復号装置200において、データアクセス領域に独立してアクセスする必要がないシステムの場合には、符号化装置100においてエントリーポイントオフセットに関する信号を符号化しない構成とすることが可能となり、図50および図51で説明された課題を考慮する必要がない。よって、符号化装置100は、処理を簡略化させることが可能となり、符号化装置100の回路規模の削減および処理サイクル数の削減が実現される可能性が高まる。

【0408】

〔実装〕

図55は、実施の形態における符号化装置の動作例を示すフローチャートである。例えば、図40に示された符号化装置100は、一次変換が行われた予測残差信号に対して二次変換処理を適用する変換処理を行う際に、図55に示された動作を行う。具体的には、プロセッサ1は、メモリ2を用いて、以下の動作を行う。

【0409】

まず、符号化装置100は、可変長符号化処理において、1つのスライスを複数のデータアクセス領域に分割して符号化する際に、生成するビットストリーム上における複数のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセット信号を、ヘッダ領域に符号化するか否かを指定する信号を、シーケンスヘッダ、ピクチャヘッダまたは拡張補助領域に記述する（ステップS100）。

【0410】

次に、符号化装置100は、オフセット信号を、ヘッダ領域に符号化すると指定されたか否かを判定する（ステップS101）。

【0411】

続いて、符号化装置100は、1つ以上のオフセット信号を、ヘッダ領域に符号化すると指定された場合（ステップS101でYes）、1つ以上のオフセット信号を、ヘッダ領域に符号化する（ステップS102）。

【0412】

そして、符号化装置100は、1つ以上のオフセット信号として符号化される複数のオフセット信号のうち連続する2つのオフセット信号の間に値が0でない1ビット以上の信号を挿入する（ステップS103）。

【0413】

また、符号化装置 100 は、1 つ以上のオフセット信号を、ヘッダ領域に符号化しないと指定された場合（ステップ S 101 で No）、1 つのスライスを複数のデータアクセス領域に分割する場合でも、オフセット信号を前記ヘッダ領域に符号化しない（ステップ S 104）。

【0414】

また、符号化装置 100 は、ヘッダ領域に符号化する 1 つ以上のオフセット信号のそれぞれを示す信号を、バイトアラインされた位置から記述してもよい。

【0415】

また、符号化装置 100 は、ヘッダ領域に符号化する 1 つ以上のオフセット信号のそれぞれを表す信号に続いて、1 ビット以上の信号を挿入し、バイトアラインしてもよい。

10

【0416】

また、符号化装置 100 は、1 つ以上のオフセット信号に含まれる 1 つのオフセット信号を 1 ビット以上の複数の信号に分割し、複数の信号のうち連続する 2 つの信号の間に、値が 0 でない 1 ビット以上の信号を挿入してもよい。

【0417】

また、符号化装置 100 は、1 つ以上のオフセット信号に含まれる 1 つのオフセット信号のビット長が、所定値以上だった場合、1 つのオフセット信号を、複数の信号に分割し、複数の信号のうち連続する 2 つの信号の間に、値が 0 でない 1 ビット以上の信号を挿入してもよい。

【0418】

20

図 56 は、実施の形態における復号装置の動作例を示すフローチャートである。例えば、図 46 に示された復号装置 200 は、二次変換を適用した変換係数に対して、さらに、一次変換を適用する逆変換処理を行う際に、図 56 に示された動作を行う。具体的には、プロセッサ b1 は、メモリ b2 を用いて、以下の動作を行う。

【0419】

まず、復号装置 200 は、可変長復号処理において、1 つのスライスを複数のデータアクセス領域に分割して符号化する際に、生成するビットストリーム上における複数のデータアクセス領域のそれぞれの先頭位置を特定するための 1 つ以上のオフセット信号を、ヘッダ領域から復号するか否かを指定する信号を、シーケンスヘッダ、ピクチャヘッダまたは拡張補助領域から読み解く（ステップ S 200）。

30

【0420】

次に、復号装置 200 は、オフセット信号を、ヘッダ領域から復号すると指定されたか否かを判定する（ステップ S 201）。

【0421】

続いて、復号装置 200 は、1 つ以上のオフセット信号を、ヘッダ領域から復号すると指定された場合（ステップ S 201 で Yes）、1 つ以上のオフセット信号を、ヘッダ領域から復号する（ステップ S 202）。

【0422】

そして、復号装置 200 は、1 つ以上のオフセット信号として符号化される複数のオフセット信号のうち連続する 2 つのオフセット信号の間から値が 0 でない 1 ビット以上の信号を復号する（ステップ S 203）。

40

【0423】

また、復号装置 200 は、1 つ以上のオフセット信号を、ヘッダ領域から復号しないと指定された場合（ステップ S 201 で No）、1 つのスライスを複数のデータアクセス領域に分割する場合でも、オフセット信号をヘッダ領域から復号しない（ステップ S 204）。

【0424】

また、復号装置 200 は、ヘッダ領域から復号する 1 つ以上のオフセット信号のそれぞれを示す信号を、バイトアラインされた位置から読み解いてもよい。

【0425】

50

また、復号装置 200 は、ヘッダ領域に復号する 1 つ以上の前記オフセット信号のそれぞれを表す信号に続いて、1 ビット以上の信号を挿入された信号として読み解き、バイトアラインしてもよい。

【0426】

また、復号装置 200 は、1 つ以上のオフセット信号に含まれる 1 つのオフセット信号を、複数の信号に分割し、複数の信号のうち連続する 2 つの信号の間に、値が 0 でない 1 ビット以上の信号が挿入された信号として読み解いてもよい。

【0427】

また、復号装置 200 は、1 つ以上のオフセット信号に含まれる 1 つのオフセット信号のビット長が、所定値以上だった場合、1 つのオフセット信号を、1 ビット以上の複数の信号に分割し、複数の信号のそれぞれの間の、値が 0 でない 1 ビット以上の信号が挿入された信号として読み解いてもよい。

【0428】

[補足]

本実施の形態における符号化装置 100 及び復号装置 200 は、それぞれ、画像符号化装置及び画像復号装置として利用されてもよいし、動画像符号化装置及び動画像復号装置として利用されてもよい。

【0429】

なお、上記各実施の形態において、各構成要素は、専用のハードウェアで構成されるか、各構成要素に適したソフトウェアプログラムを実行することによって実現されてもよい。各構成要素は、CPU 又はプロセッサなどのプログラム実行部が、ハードディスク又は半導体メモリなどの記録媒体に記録されたソフトウェアプログラムを読み出して実行することによって実現されてもよい。

【0430】

具体的には、符号化装置 100 及び復号装置 200 のそれぞれは、処理回路 (Processing Circuitry) と、当該処理回路に電気的に接続された、当該処理回路からアクセス可能な記憶装置 (Storage) とを備えていてもよい。例えば、処理回路はプロセッサ a1 又は b1 に対応し、記憶装置はメモリ a2 又は b2 に対応する。

【0431】

処理回路は、専用のハードウェア及びプログラム実行部の少なくとも一方を含み、記憶装置を用いて処理を実行する。また、記憶装置は、処理回路がプログラム実行部を含む場合には、当該プログラム実行部により実行されるソフトウェアプログラムを記憶する。

【0432】

ここで、本実施の形態の符号化装置 100 又は復号装置 200 などを実現するソフトウェアは、次のようなプログラムである。

【0433】

すなわち、このプログラムは、コンピュータに、可変長符号化処理において、1 つのスライスを複数のデータアクセス領域に分割して符号化する際に、生成するビットストリーム上における複数のデータアクセス領域のそれぞれの先頭位置を特定するための 1 つ以上のオフセット信号を、ヘッダ領域に符号化するか否かを指定する信号を、シーケンスヘッダ、ピクチャヘッダまたは拡張補助領域に記述し、1 つ以上のオフセット信号を、ヘッダ領域に符号化すると指定された場合、オフセット信号を、1 つ以上のオフセット信号をヘッダ領域に符号化するか否かの指定に従って、ヘッダ領域に符号化し、1 つ以上のオフセット信号として符号化される複数のオフセット信号のうち連続する 2 つのオフセット信号の間に値が 0 でない 1 ビット以上の信号を挿入し、1 つ以上のオフセット信号を、ヘッダ領域に符号化しないと指定された場合、1 つのスライスを複数のデータアクセス領域に分割する場合でも、オフセット信号をヘッダ領域に符号化しない符号化方法を実行させてもよい。

【0434】

あるいは、このプログラムは、コンピュータに、可変長復号処理において、1 つのスラ

10

20

30

40

50

イスが複数のデータアクセス領域に分割されたビットストリームを復号する際に、ビットストリーム上における複数のデータアクセス領域のそれぞれの先頭位置を特定するための1つ以上のオフセット信号を、ヘッダ領域に含まれるか否かを指定する信号を、シーケンスヘッダ、ピクチャヘッダまたは拡張補助領域から読み解き、1つ以上のオフセット信号を、ヘッダ領域から復号すると指定された場合、1つ以上のオフセット信号をヘッダ領域から復号し、1つ以上のオフセット信号として復号される複数のオフセット信号のうち連続する2つのオフセット信号の間に挿入されている値が0でない1ビット以上の信号を復号し、1つ以上のオフセット信号を、ヘッダ領域から復号しないと指定された場合、1つのスライスが複数のデータアクセス領域に分割する場合でも、オフセット信号をビットストリームから復号しない復号方法を実行させてもよい。

10

【0435】

また、各構成要素は、上述の通り、回路であってもよい。これらの回路は、全体として1つの回路を構成してもよいし、それぞれ別々の回路であってもよい。また、各構成要素は、汎用的なプロセッサで実現されてもよいし、専用のプロセッサで実現されてもよい。

【0436】

また、特定の構成要素が実行する処理を別の構成要素が実行してもよい。また、処理を実行する順番が変更されてもよいし、複数の処理が並行して実行されてもよい。また、符号化復号装置が、符号化装置100及び復号装置200を備えていてもよい。

【0437】

また、説明に用いられた第1及び第2等の序数は、適宜、付け替えられてもよい。また、構成要素などに対して、序数が新たに与えられてもよいし、取り除かれてもよい。

20

【0438】

以上、符号化装置100及び復号装置200の態様について、実施の形態に基づいて説明したが、符号化装置100及び復号装置200の態様は、この実施の形態に限定されるものではない。本開示の趣旨を逸脱しない限り、当業者が思いつく各種変形を本実施の形態に施したものや、異なる実施の形態における構成要素を組み合わせて構築される形態も、符号化装置100及び復号装置200の態様の範囲内に含まれてもよい。

【0439】

ここで開示された1以上の態様を本開示における他の態様の少なくとも一部と組み合わせて実施してもよい。また、ここで開示された1以上の態様のフローチャートに記載の一部の処理、装置の一部の構成、シンタックスの一部などを他の態様と組み合わせて実施してもよい。

30

【0440】

〔実施及び応用〕

以上の各実施の形態において、機能的又は作用的なブロックの各々は、通常、MPU (micro processing unit) 及びメモリ等によって実現可能である。また、機能ブロックの各々による処理は、ROM等の記録媒体に記録されたソフトウェア (プログラム) を読み出して実行するプロセッサなどのプログラム実行部として実現されてもよい。当該ソフトウェアは、配布されてもよい。当該ソフトウェアは、半導体メモリなどの様々な記録媒体に記録されてもよい。なお、各機能ブロックをハードウェア (専用回路) によって実現することも可能である。ハードウェア及びソフトウェアの様々な組み合わせが採用され得る。

40

【0441】

各実施の形態において説明した処理は、単一の装置 (システム) を用いて集中処理することによって実現してもよく、又は、複数の装置を用いて分散処理することによって実現してもよい。また、上記プログラムを実行するプロセッサは、単数であってもよく、複数であってもよい。すなわち、集中処理を行ってもよく、又は分散処理を行ってもよい。

【0442】

本開示の態様は、以上の実施例に限定されることなく、種々の変更が可能であり、それらも本開示の態様の範囲内に包含される。

50

【0443】

さらにここで、上記各実施の形態で示した動画像符号化方法（画像符号化方法）又は動画像復号化方法（画像復号方法）の応用例、及び、その応用例を実施する種々のシステムを説明する。このようなシステムは、画像符号化方法を用いた画像符号化装置、画像復号方法を用いた画像復号装置、又は、両方を備える画像符号化復号装置を有することを特徴としてもよい。このようなシステムの他の構成について、場合に応じて適切に変更することができる。

【0444】

〔使用例〕

図57は、コンテンツ配信サービスを実現する適切なコンテンツ供給システムex100の全体構成を示す図である。通信サービスの提供エリアを所望の大きさに分割し、各セル内にそれぞれ、図示された例における固定無線局である基地局ex106、ex107、ex108、ex109、ex110が設置されている。

【0445】

このコンテンツ供給システムex100では、インターネットex101に、インターネットサービスプロバイダex102又は通信網ex104、及び基地局ex106～ex110を介して、コンピュータex111、ゲーム機ex112、カメラex113、家電ex114、及びスマートフォンex115などの各機器が接続される。当該コンテンツ供給システムex100は、上記のいずれかの装置を組合せて接続するようにしてもよい。種々の実施において、基地局ex106～ex110を介さずに、各機器が電話網又は近距離無線等を介して直接的又は間接的に相互に接続されていてもよい。さらに、ストリーミングサーバex103は、インターネットex101等を介して、コンピュータex111、ゲーム機ex112、カメラex113、家電ex114、及びスマートフォンex115などの各機器と接続されてもよい。また、ストリーミングサーバex103は、衛星ex116を介して、飛行機ex117内のホットスポット内の端末等と接続されてもよい。

【0446】

なお、基地局ex106～ex110の代わりに、無線アクセスポイント又はホットスポット等が用いられてもよい。また、ストリーミングサーバex103は、インターネットex101又はインターネットサービスプロバイダex102を介さずに直接通信網ex104と接続されてもよいし、衛星ex116を介さずに直接飛行機ex117と接続されてもよい。

【0447】

カメラex113はデジタルカメラ等の静止画撮影、及び動画撮影が可能な機器である。また、スマートフォンex115は、2G、3G、3.9G、4G、そして今後は5Gと呼ばれる移動通信システムの方式に対応したスマートフォン機、携帯電話機、又はPHS（Personal Handy-phone System）等である。

【0448】

家電ex114は、冷蔵庫、又は家庭用燃料電池コージェネレーションシステムに含まれる機器等である。

【0449】

コンテンツ供給システムex100では、撮影機能を有する端末が基地局ex106等を通じてストリーミングサーバex103に接続されることで、ライブ配信等が可能になる。ライブ配信では、端末（コンピュータex111、ゲーム機ex112、カメラex113、家電ex114、スマートフォンex115、及び飛行機ex117内の端末等）は、ユーザが当該端末を用いて撮影した静止画又は動画コンテンツに対して上記各実施の形態で説明した符号化処理を行ってもよく、符号化により得られた映像データと、映像に対応する音を符号化した音データと多重化してもよく、得られたデータをストリーミングサーバex103に送信してもよい。即ち、各端末は、本開示の一態様に係る画像符号化装置として機能する。

【0450】

一方、ストリーミングサーバex103は要求のあったクライアントに対して送信されたコンテンツデータをストリーム配信する。クライアントは、上記符号化処理されたデータを復号化することが可能な、コンピュータex111、ゲーム機ex112、カメラex113、家電ex114、スマートフォンex115、又は飛行機ex117内の端末等である。配信されたデータを受信した各機器は、受信したデータを復号化処理して再生してもよい。即ち、各機器は、本開示の一態様に係る画像復号装置として機能してもよい。

【0451】

〔分散処理〕

また、ストリーミングサーバex103は複数のサーバ又は複数のコンピュータであって、データを分散して処理したり記録したり配信するものであってもよい。例えば、ストリーミングサーバex103は、CDN (Contents Delivery Network) により実現され、世界中に分散された多数のエッジサーバとエッジサーバ間をつなぐネットワークによりコンテンツ配信が実現されていてもよい。CDNでは、クライアントに応じて物理的に近いエッジサーバが動的に割り当てられ得る。そして、当該エッジサーバにコンテンツがキャッシュ及び配信されることで遅延を減らすことができる。また、いくつかのタイプのエラーが発生した場合又はトラフィックの増加などにより通信状態が変わる場合に複数のエッジサーバで処理を分散したり、他のエッジサーバに配信主体を切り替えたり、障害が生じたネットワークの部分を迂回して配信を続けることができるので、高速かつ安定した配信が実現できる。

【0452】

また、配信自体の分散処理にとどまらず、撮影したデータの符号化処理を各端末で行ってもよいし、サーバ側で行ってもよいし、互いに分担して行ってもよい。一例として、一般に符号化処理では、処理ループが2度行われる。1度目のループでフレーム又はシーン単位での画像の複雑さ、又は、符号量が検出される。また、2度目のループでは画質を維持して符号化効率を向上させる処理が行われる。例えば、端末が1度目の符号化処理を行い、コンテンツを受け取ったサーバ側が2度目の符号化処理を行うことで、各端末での処理負荷を減らしつつもコンテンツの質と効率を向上させることができる。この場合、ほぼリアルタイムで受信して復号する要求があれば、端末が行った一度目の符号化済みデータを他の端末で受信して再生することもできるので、より柔軟なリアルタイム配信も可能になる。

【0453】

他の例として、カメラex113等は、画像から特徴量（特徴又は特性の量）を抽出し、特徴量に関するデータをメタデータとして圧縮してサーバに送信する。サーバは、例えば特徴量からオブジェクトの重要性を判断して量子化精度を切り替えるなど、画像の意味（又は内容の重要性）に応じた圧縮を行う。特徴量データはサーバでの再度の圧縮時の動きベクトル予測の精度及び効率向上に特に有効である。また、端末でVLC（可変長符号化）などの簡易的な符号化を行い、サーバでCABAC（コンテキスト適応型二値算術符号化方式）など処理負荷の大きな符号化を行ってもよい。

【0454】

さらに他の例として、スタジアム、ショッピングモール、又は工場などにおいては、複数の端末によりほぼ同一のシーンが撮影された複数の映像データが存在する場合がある。この場合には、撮影を行った複数の端末と、必要に応じて撮影をしていない他の端末及びサーバを用いて、例えばGOP (Group of Picture) 単位、ピクチャ単位、又はピクチャを分割したタイル単位などで符号化処理をそれぞれ割り当てて分散処理を行う。これにより、遅延を減らし、よりリアルタイム性を実現できる。

【0455】

複数の映像データはほぼ同一シーンであるため、各端末で撮影された映像データを互いに参照し合えるように、サーバで管理及び／又は指示をしてもよい。また、各端末からの符号化済みデータを、サーバが受信し複数のデータ間で参照関係を変更、又はピクチャ自

10

20

30

40

50

体を補正或いは差し替えて符号化しなおしてもよい。これにより、一つ一つのデータの質と効率を高めたストリームを生成できる。

【0456】

さらに、サーバは、映像データの符号化方式を変更するトランスコードを行ったうえで映像データを配信してもよい。例えば、サーバは、MPEG系の符号化方式をVP系（例えばVP9）に変換してもよいし、H.264をH.265に変換等してもよい。

【0457】

このように、符号化処理は、端末、又は1以上のサーバにより行うことが可能である。よって、以下では、処理を行う主体として「サーバ」又は「端末」等の記載を用いるが、サーバで行われる処理の一部又は全てが端末で行われてもよいし、端末で行われる処理の一部又は全てがサーバで行われてもよい。また、これらに関しては、復号処理についても同様である。

【0458】

〔3D、マルチアングル〕

互いにほぼ同期した複数のカメラex113及び／又はスマートフォンex115などの端末により撮影された異なるシーン、又は、同一シーンを異なるアングルから撮影した画像或いは映像を統合して利用することが増えてきている。各端末で撮影した映像は、別途取得した端末間の相対的な位置関係、又は、映像に含まれる特徴点が一致する領域などに基づいて統合され得る。

【0459】

サーバは、2次元の動画像を符号化するだけでなく、動画像のシーン解析などに基づいて自動的に、又は、ユーザが指定した時刻において、静止画を符号化し、受信端末に送信してもよい。サーバは、さらに、撮影端末間の相対的な位置関係を取得できる場合には、2次元の動画像だけでなく、同一シーンが異なるアングルから撮影された映像に基づき、当該シーンの3次元形状を生成できる。サーバは、ポイントクラウドなどにより生成した3次元のデータを別途符号化してもよいし、3次元データを用いて人物又はオブジェクトを認識或いは追跡した結果に基づいて、受信端末に送信する映像を、複数の端末で撮影した映像から、選択、又は、再構成して生成してもよい。

【0460】

このようにして、ユーザは、各撮影端末に対応する各映像を任意に選択してシーンを楽しむこともできるし、複数画像又は映像を用いて再構成された3次元データから選択視点の映像を切り出したコンテンツを楽しむこともできる。さらに、映像と共に、音も複数の相異なるアングルから収録され、サーバは、特定のアングル又は空間からの音に対応する映像と多重化して、多重化された映像と音とを送信してもよい。

【0461】

また、近年ではVirtual Reality（VR）及びAugmented Reality（AR）など、現実世界と仮想世界とを対応付けたコンテンツも普及してきている。VRの画像の場合、サーバは、右目用及び左目用の視点画像をそれぞれ作成し、Multi-View Coding（MVC）などにより各視点映像間で参照を許容する符号化を行ってもよいし、互いに参照せずに別ストリームとして符号化してもよい。別ストリームの復号時には、ユーザの視点に応じて仮想的な3次元空間が再現されるように互いに同期させて再生するとよい。

【0462】

ARの画像の場合には、サーバは、現実空間のカメラ情報に、仮想空間上の仮想物体情報を、3次元的位置又はユーザの視点の動きに基づいて重畳してもよい。復号装置は、仮想物体情報及び3次元データを取得又は保持し、ユーザの視点の動きに応じて2次元画像を生成し、スムーズにつなげることで重畳データを作成してもよい。または、復号装置は仮想物体情報の依頼に加えてユーザの視点の動きをサーバに送信してもよい。サーバは、サーバに保持される3次元データから受信した視点の動きに合わせて重畳データを作成し、重畳データを符号化して復号装置に配信してもよい。なお、重畳データは、典型的には

10

20

30

40

50

、RGB以外に透過度を示す α 値を有し、サーバは、3次元データから作成されたオブジェクト以外の部分の α 値が0などに設定し、当該部分が透過する状態で、符号化してもよい。もしくは、サーバは、クロマキーのように所定の値のRGB値を背景に設定し、オブジェクト以外の部分は背景色にしたデータを生成してもよい。所定の値のRGB値は、予め定められていてもよい。

【0463】

同様に配信されたデータの復号処理はクライアント（例えば、端末）で行っても、サーバ側で行ってもよいし、互いに分担して行ってもよい。一例として、ある端末が、一旦サーバに受信リクエストを送り、そのリクエストに応じたコンテンツを他の端末で受信し復号処理を行い、ディスプレイを有する装置に復号済みの信号が送信されてもよい。通信可能な端末自体の性能によらず処理を分散して適切なコンテンツを選択することで画質のよいデータを再生することができる。また、他の例として大きなサイズの画像データをTV等で受信しつつ、鑑賞者の個人端末にピクチャが分割されたタイルなど一部の領域が復号されて表示されてもよい。これにより、全体像を共有化しつつ、自身の担当分野又はより詳細に確認したい領域を手元で確認することができる。

【0464】

屋内外の近距離、中距離、又は長距離の無線通信が複数使用可能な状況下で、MPEG-DASHなどの配信システム規格を利用して、シームレスにコンテンツを受信することが可能かもしれない。ユーザは、ユーザの端末、屋内外に配置されたディスプレイなどの復号装置又は表示装置を自由に選択しながらリアルタイムで切り替えてもよい。また、自身の位置情報などを用いて、復号する端末及び表示する端末を切り替えながら復号を行うことができる。これにより、ユーザが目的地へ移動している間に、表示可能なデバイスが埋め込まれた隣の建物の壁面又は地面の一部に情報をマップ及び表示することが可能になる。また、符号化データが受信端末から短時間でアクセスできるサーバにキャッシュされている、又は、コンテンツ・デリバリー・サービスにおけるエッジサーバにコピーされている、などの、ネットワーク上での符号化データへのアクセス容易性に基づいて、受信データのビットレートを切り替えることも可能である。

【0465】

【スケーラブル符号化】

コンテンツの切り替えに関して、図58に示す、上記各実施の形態で示した動画像符号化方法を応用して圧縮符号化されたスケーラブルなストリームを用いて説明する。サーバは、個別のストリームとして内容は同じで質の異なるストリームを複数有していても構わないが、図示するようにレイヤに分けて符号化を行うことで実現される時間的／空間的スケーラブルなストリームの特徴を活かして、コンテンツを切り替える構成であってもよい。つまり、復号側が性能という内的要因と通信帯域の状態などの外的要因とに応じてどのレイヤを復号するかを決定することで、復号側は、低解像度のコンテンツと高解像度のコンテンツとを自由に切り替えて復号できる。例えばユーザが移動中にスマートフォンex115で視聴していた映像の続きを、例えば帰宅後にインターネットTV等の機器で視聴したい場合には、当該機器は、同じストリームを異なるレイヤまで復号すればよいので、サーバ側の負担を軽減できる。

【0466】

さらに、上記のように、レイヤ毎にピクチャが符号化されており、ベースレイヤの上位のエンハンスメントレイヤでスケーラビリティを実現する構成以外に、エンハンスメントレイヤが画像の統計情報などに基づくメタ情報を含んでいてもよい。復号側が、メタ情報に基づきベースレイヤのピクチャを超解像することで高画質化したコンテンツを生成してもよい。超解像は、解像度を維持及び／又は拡大しつつ、SN比を向上してもよい。メタ情報は、超解像処理に用いるような線形或いは非線形のフィルタ係数を特定するため情報、又は、超解像処理に用いるフィルタ処理、機械学習或いは最小2乗演算におけるパラメータ値を特定する情報などを含む。

【0467】

または、画像内のオブジェクトなどの意味合いに応じてピクチャがタイル等に分割される構成が提供されてもよい。復号側が、復号するタイルを選択することで一部の領域だけを復号する。さらに、オブジェクトの属性（人物、車、ボールなど）と映像内の位置（同一画像における座標位置など）とをメタ情報として格納することで、復号側は、メタ情報に基づいて所望のオブジェクトの位置を特定し、そのオブジェクトを含むタイルを決定できる。例えば、図59に示すように、メタ情報は、HEVCにおけるSEI（*supplemental enhancement information*）メッセージなど、画素データとは異なるデータ格納構造を用いて格納されてもよい。このメタ情報は、例えば、メインオブジェクトの位置、サイズ、又は色彩などを示す。

【0468】

ストリーム、シーケンス又はランダムアクセス単位など、複数のピクチャから構成される単位でメタ情報が格納されてもよい。復号側は、特定人物が映像内に出現する時刻などを取得でき、ピクチャ単位の情報と時間情報を合わせることで、オブジェクトが存在するピクチャを特定でき、ピクチャ内でのオブジェクトの位置を決定できる。

【0469】

〔Webページの最適化〕

図60は、コンピュータex111等におけるwebページの表示画面例を示す図である。図61は、スマートフォンex115等におけるwebページの表示画面例を示す図である。図60及び図61に示すようにwebページが、画像コンテンツへのリンクであるリンク画像を複数含む場合があり、閲覧するデバイスによってその見え方は異なってもよい。画面上に複数のリンク画像が見える場合には、ユーザが明示的にリンク画像を選択するまで、又は画面の中央付近にリンク画像が近付く或いはリンク画像の全体が画面内に入るまで、表示装置（復号装置）は、リンク画像として各コンテンツが有する静止画又はIピクチャを表示してもよいし、複数の静止画又はIピクチャ等でgifアニメのような映像を表示してもよいし、ベースレイヤのみを受信し、映像を復号及び表示してもよい。

【0470】

ユーザによりリンク画像が選択された場合、表示装置は、例えばベースレイヤを最優先にしつつ復号を行う。なお、webページを構成するHTMLにスケーラブルなコンテンツであることを示す情報があれば、表示装置は、エンハンスメントレイヤまで復号してもよい。さらに、リアルタイム性を担保するために、選択される前又は通信帯域が非常に厳しい場合には、表示装置は、前方参照のピクチャ（Iピクチャ、Pピクチャ、前方参照のみのBピクチャ）のみを復号及び表示することで、先頭ピクチャの復号時刻と表示時刻との間の遅延（コンテンツの復号開始から表示開始までの遅延）を低減できる。またさらに、表示装置は、ピクチャの参照関係を敢えて無視して、全てのBピクチャ及びPピクチャを前方参照にして粗く復号し、時間が経ち受信したピクチャが増えるにつれて正常の復号を行ってもよい。

【0471】

〔自動走行〕

また、車の自動走行又は走行支援のため2次元又は3次元の地図情報などのような静止画又は映像データを送受信する場合、受信端末は、1以上のレイヤに属する画像データに加えて、メタ情報として天候又は工事の情報なども受信し、これらに対応付けて復号してもよい。なお、メタ情報は、レイヤに属してもよいし、単に画像データと多重化されてもよい。

【0472】

この場合、受信端末を含む車、ドローン又は飛行機などが移動するため、受信端末は、当該受信端末の位置情報を送信することで、基地局ex106～ex110を切り替えながらシームレスな受信及び復号の実行を実現できる。また、受信端末は、ユーザの選択、ユーザの状況及び／又は通信帯域の状態に応じて、メタ情報をどの程度受信するか、又は地図情報をどの程度更新していくかを動的に切り替えることが可能になる。

【0473】

コンテンツ供給システム ex 100 では、ユーザが送信した符号化された情報をリアルタイムでクライアントが受信して復号し、再生することができる。

【0474】

〔個人コンテンツの配信〕

また、コンテンツ供給システム ex 100 では、映像配信業者による高画質で長時間のコンテンツのみならず、個人による低画質で短時間のコンテンツのユニキャスト、又はマルチキャスト配信が可能である。このような個人のコンテンツは今後も増加していくと考えられる。個人コンテンツをより優れたコンテンツにするために、サーバは、編集処理を行ってから符号化処理を行ってもよい。これは、例えば、以下のような構成を用いて実現

10

【0475】

撮影時にリアルタイム又は蓄積して撮影後に、サーバは、原画データ又は符号化済みデータから撮影エラー、シーン探索、意味の解析、及びオブジェクト検出などの認識処理を行う。そして、サーバは、認識結果に基づいて手動又は自動で、ピントずれ又は手ブレなどを補正したり、明度が他のピクチャに比べて低い又は焦点が合っていないシーンなどの重要性の低いシーンを削除したり、オブジェクトのエッジを強調したり、色合いを変化させるなどの編集を行う。サーバは、編集結果に基づいて編集後のデータを符号化する。また撮影時刻が長すぎると視聴率が下がることも知られており、サーバは、撮影時間に応じて特定の時間範囲内のコンテンツになるように上記のように重要性が低いシーンのみならず動きが少ないシーンなどを、画像処理結果に基づき自動でクリップしてもよい。または、サーバは、シーンの意味解析の結果に基づいてダイジェストを生成して符号化してもよい。

20

【0476】

個人コンテンツには、そのままでは著作権、著作者人格権、又は肖像権等の侵害となるものが写り込んでいるケースもあり、共有する範囲が意図した範囲を超えてしまうなど個人にとって不都合な場合もある。よって、例えば、サーバは、画面の周辺部の人の顔、又は家の中などを敢えて焦点が合わない画像に変更して符号化してもよい。さらに、サーバは、符号化対象画像内に、予め登録した人物とは異なる人物の顔が映っているかどうかを認識し、映っている場合には、顔の部分にモザイクをかけるなどの処理を行ってもよい。または、符号化の前処理又は後処理として、著作権などの観点からユーザが画像を加工したい人物又は背景領域を指定してもよい。サーバは、指定された領域を別の映像に置き換える、又は焦点をぼかすなどの処理を行ってもよい。人物であれば、動画像において人物をトラッキングして、人物の顔の部分の映像を置き換えることができる。

30

【0477】

データ量の小さい個人コンテンツの視聴はリアルタイム性の要求が強いため、帯域幅にもよるが、復号装置は、まずベースレイヤを最優先で受信して復号及び再生を行ってもよい。復号装置は、この間にエンハンスメントレイヤを受信し、再生がループされる場合など2回以上再生される場合に、エンハンスメントレイヤも含めて高画質の映像を再生してもよい。このようにスケーラブルな符号化が行われているストリームであれば、未選択時又は見始めた段階では粗い動画だが、徐々にストリームがスマートになり画像がよくなるような体験を提供することができる。スケーラブル符号化以外にも、1回目に再生される粗いストリームと、1回目の動画を参照して符号化される2回目のストリームとが1つのストリームとして構成されていても同様の体験を提供できる。

40

【0478】

〔その他の実施応用例〕

また、これらの符号化又は復号処理は、一般的に各端末が有する L S I ex 500 において処理される。L S I (large scale integration circuitry) ex 500 (図57参照) は、ワンチップであっても複数チップからなる構成であってもよい。なお、動画像符号化又は復号用のソフトウェアをコンピュータ ex 11

50

1等で読み取り可能な何らかの記録メディア（ＣＤ－ＲＯＭ、フレキシブルディスク、又はハードディスクなど）に組み込み、そのソフトウェアを用いて符号化又は復号処理を行ってもよい。さらに、スマートフォンex115がカメラ付きである場合には、そのカメラで取得した動画データを送信してもよい。このときの動画データはスマートフォンex115が有するLSIex500で符号化処理されたデータであってもよい。

【0479】

なお、LSIex500は、アプリケーションソフトをダウンロードしてアクティベートする構成であってもよい。この場合、端末は、まず、当該端末がコンテンツの符号化方式に対応しているか、又は、特定サービスの実行能力を有するかを判定する。端末がコンテンツの符号化方式に対応していない場合、又は、特定サービスの実行能力を有さない場合、端末は、コーデック又はアプリケーションソフトをダウンロードし、その後、コンテンツ取得及び再生してもよい。

【0480】

また、インターネットex101を介したコンテンツ供給システムex100に限らず、デジタル放送用システムにも上記各実施の形態の少なくとも動画像符号化装置（画像符号化装置）又は動画像復号化装置（画像復号装置）のいずれかを組み込むことができる。衛星などを利用して放送用の電波に映像と音が多重化された多重化データを載せて送受信するため、コンテンツ供給システムex100のユニキャストがし易い構成に対してマルチキャスト向きであるという違いがあるが符号化処理及び復号処理に関しては同様の応用が可能である。

【0481】

〔ハードウェア構成〕

図62は、図57に示されたスマートフォンex115のさらに詳細を示す図である。また、図63は、スマートフォンex115の構成例を示す図である。スマートフォンex115は、基地局ex110との間で電波を送受信するためのアンテナex450と、映像及び静止画を撮ることが可能なカメラ部ex465と、カメラ部ex465で撮像した映像、及びアンテナex450で受信した映像等が復号されたデータを表示する表示部ex458とを備える。スマートフォンex115は、さらに、タッチパネル等である操作部ex466と、音声又は音響を出力するためのスピーカ等である音声出力部ex457と、音声を入力するためのマイク等である音声入力部ex456と、撮影した映像或いは静止画、録音した音声、受信した映像或いは静止画、メール等の符号化されたデータ、又は、復号化されたデータを保存可能なメモリ部ex467と、ユーザを特定し、ネットワークをはじめ各種データへのアクセスの認証をするためのSIMex468とのインタフェース部であるスロット部ex464とを備える。なお、メモリ部ex467の代わりに外付けメモリが用いられてもよい。

【0482】

表示部ex458及び操作部ex466等を統括的に制御し得る主制御部ex460と、電源回路部ex461、操作入力制御部ex462、映像信号処理部ex455、カメラインタフェース部ex463、ディスプレイ制御部ex459、変調／復調部ex452、多重／分離部ex453、音声信号処理部ex454、スロット部ex464、及びメモリ部ex467とが同期バスex470を介して接続されている。

【0483】

電源回路部ex461は、ユーザの操作により電源キーがオン状態にされると、スマートフォンex115を動作可能な状態に起動し、バッテリーパックから各部に対して電力を供給する。

【0484】

スマートフォンex115は、CPU、ROM及びRAM等を有する主制御部ex460の制御に基づいて、通話及データ通信等の処理を行う。通話時は、音声入力部ex456で收音した音声信号を音声信号処理部ex454でデジタル音声信号に変換し、変調／復調部ex452でスペクトラム拡散処理を施し、送信／受信部ex451でデジタルア

ナログ変換処理及び周波数変換処理を施し、その結果の信号を、アンテナ e x 4 5 0 を介して送信する。また受信データを増幅して周波数変換処理及びアナログデジタル変換処理を施し、変調／復調部 e x 4 5 2 でスペクトラム逆拡散処理し、音声信号処理部 e x 4 5 4 でアナログ音声信号に変換した後、これを音声出力部 e x 4 5 7 から出力する。データ通信モード時は、本体部の操作部 e x 4 6 6 等の操作に基づいてテキスト、静止画、又は映像データが操作入力制御部 e x 4 6 2 を介して主制御部 e x 4 6 0 の制御下で送出され得る。同様の送受信処理が行われる。データ通信モード時に映像、静止画、又は映像と音声を送信する場合、映像信号処理部 e x 4 5 5 は、メモリ部 e x 4 6 7 に保存されている映像信号又はカメラ部 e x 4 6 5 から入力された映像信号を上記各実施の形態で示した動画像符号化方法によって圧縮符号化し、符号化された映像データを多重／分離部 e x 4 5 3 に送出する。音声信号処理部 e x 4 5 4 は、映像又は静止画をカメラ部 e x 4 6 5 で撮像中に音声入力部 e x 4 5 6 で收音した音声信号を符号化し、符号化された音声データを多重／分離部 e x 4 5 3 に送出する。多重／分離部 e x 4 5 3 は、符号化済み映像データと符号化済み音声データを所定の方式で多重化し、変調／復調部（変調／復調回路部） e x 4 5 2、及び送信／受信部 e x 4 5 1 で変調処理及び変換処理を施してアンテナ e x 4 5 0 を介して送信する。所定の方式は、予め定められていてもよい。

【0485】

電子メール又はチャットに添付された映像、又はウェブページにリンクされた映像を受信した場合等において、アンテナ e x 4 5 0 を介して受信された多重化データを復号するために、多重／分離部 e x 4 5 3 は、多重化データを分離することにより、多重化データを映像データのビットストリームと音声データのビットストリームとに分け、同期バス e x 4 7 0 を介して符号化された映像データを映像信号処理部 e x 4 5 5 に供給するとともに、符号化された音声データを音声信号処理部 e x 4 5 4 に供給する。映像信号処理部 e x 4 5 5 は、上記各実施の形態で示した動画像符号化方法に対応した動画像復号化方法によって映像信号を復号し、ディスプレイ制御部 e x 4 5 9 を介して表示部 e x 4 5 8 から、リンクされた動画像ファイルに含まれる映像又は静止画が表示される。音声信号処理部 e x 4 5 4 は、音声信号を復号し、音声出力部 e x 4 5 7 から音声が出力される。リアルタイムストリーミングがますます普及しだしているため、ユーザの状況によっては音声の再生が社会的にふさわしくないこともあり得る。そのため、初期値としては、音声信号は再生せず映像データのみを再生する構成の方が望ましく、ユーザが映像データをクリックするなど操作を行った場合にのみ音声を同期して再生してもよい。

【0486】

またここではスマートフォン e x 1 1 5 を例に説明したが、端末としては符号化器及び復号化器を両方持つ送受信型端末の他に、符号化器のみを有する送信端末、及び、復号化器のみを有する受信端末という他の実装形式が考えられる。デジタル放送用システムにおいて、映像データに音声データが多重化された多重化データを受信又は送信するとして説明した。ただし、多重化データには、音声データ以外に映像に関連する文字データなどが多重化されてもよい。また、多重化データではなく映像データ自体が受信又は送信されてもよい。

【0487】

なお、CPUを含む主制御部 e x 4 6 0 が符号化又は復号処理を制御するとして説明したが、種々の端末はGPUを備えることも多い。よって、CPUとGPUで共通化されたメモリ、又は共通に使用できるようにアドレスが管理されているメモリにより、GPUの性能を活かして広い領域を一括して処理する構成でもよい。これにより符号化時間を短縮でき、リアルタイム性を確保し、低遅延を実現できる。特に動き探索、デブロックフィルタ、SAO (Sample Adaptive Offset)、及び変換・量子化の処理を、CPUではなく、GPUでピクチャなどの単位で一括して行うと効率的である。

【産業上の利用可能性】

【0488】

本開示は、例えば、テレビジョン受像機、デジタルビデオレコーダー、カーナビゲーシ

10

20

30

40

50

ョン、携帯電話、デジタルカメラ、デジタルビデオカメラ、テレビ会議システム、又は、電子ミラー等に利用可能である。

【符号の説明】

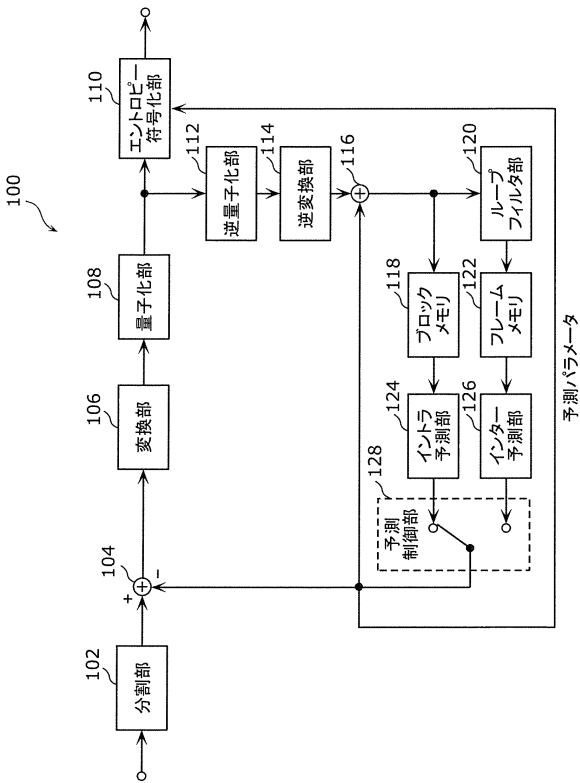
【0489】

100	符号化装置	
102	分割部	
104	減算部	
106	変換部	
108	量子化部	
110	エントロピー符号化部	10
112、204	逆量子化部	
114、206	逆変換部	
116、208	加算部	
118、210	ブロックメモリ	
120、212	ループフィルタ部	
122、214	フレームメモリ	
124、216	イントラ予測部	
126、218	インター予測部	
128、220	予測制御部	
200	復号装置	20
202	エントロピー復号部	
1201	境界判定部	
1202、1204、1206	スイッチ	
1203	フィルタ判定部	
1205	フィルタ処理部	
1207	フィルタ特性決定部	
1208	処理判定部	
a1、b1	プロセッサ	
a2、b2	メモリ	30

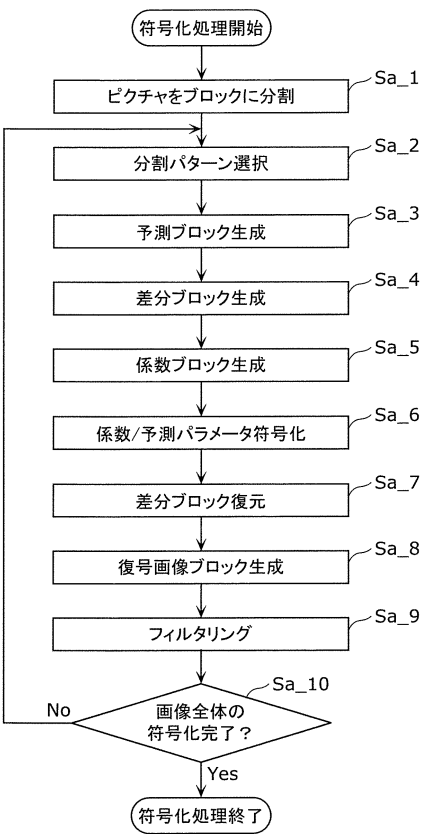
40

50

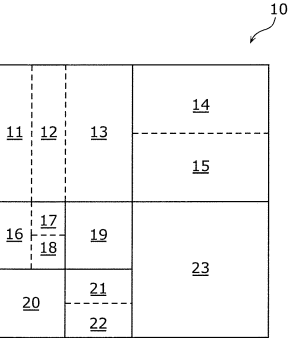
【図面】
【図 1】



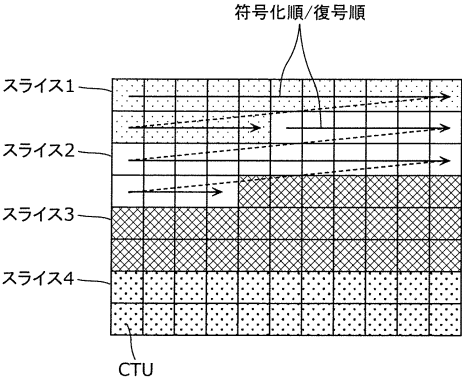
【図 2】



【図 3】



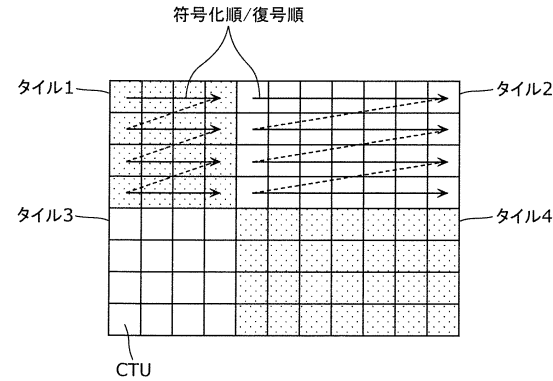
【図 4 A】



10

20

【図 4 B】



【図 5 A】

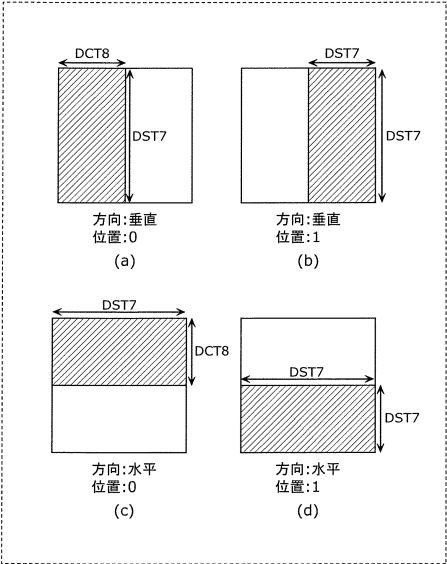
変換タイプ	基底関数 $T_i(j)$, $i, j=0, 1, \dots, N-1$
DCT-II	$T_i(j) = \omega_0 \cdot \sqrt{\frac{2}{N}} \cdot \cos\left(\frac{n \cdot i \cdot (2j+1)}{2N}\right)$ where $\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} & i = 0 \\ 1 & i \neq 0 \end{cases}$
DCT-V	$T_i(j) = \omega_0 \cdot \omega_1 \cdot \sqrt{\frac{2}{2N-1}} \cdot \cos\left(\frac{2n \cdot i \cdot j}{2N-1}\right)$ where $\omega_0 = \begin{cases} \sqrt{\frac{2}{N}} & i = 0 \\ 1 & i \neq 0 \end{cases}$, $\omega_1 = \begin{cases} \sqrt{\frac{2}{N}} & j = 0 \\ 1 & j \neq 0 \end{cases}$
DCT-VIII	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \cos\left(\frac{n \cdot (2i+1) \cdot (2j+1)}{4N+2}\right)$
DST-I	$T_i(j) = \sqrt{\frac{2}{N+1}} \cdot \sin\left(\frac{n \cdot (i+1) \cdot (j+1)}{N+1}\right)$
DST-VII	$T_i(j) = \sqrt{\frac{4}{2N+1}} \cdot \sin\left(\frac{n \cdot (2i+1) \cdot (j+1)}{2N+1}\right)$

30

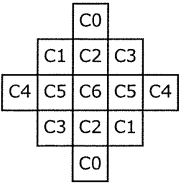
40

50

【図 5 B】



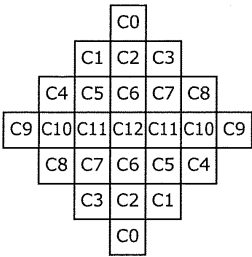
【図 6 A】



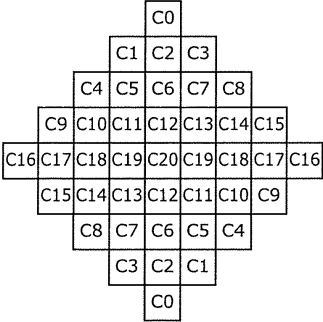
10

20

【図 6 B】



【図 6 C】

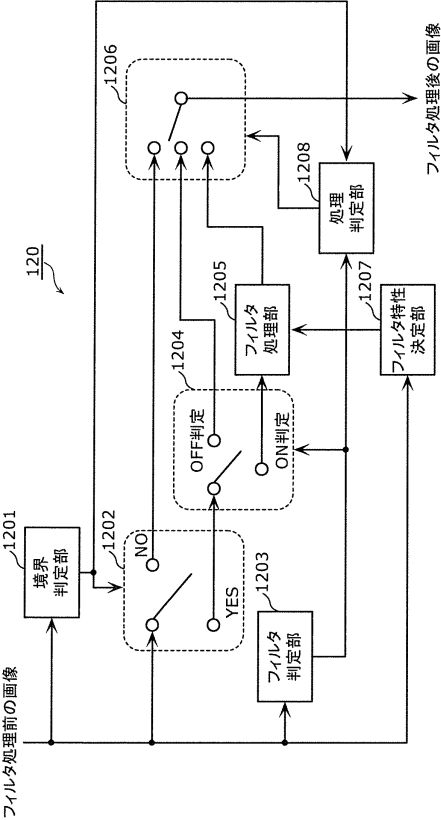


30

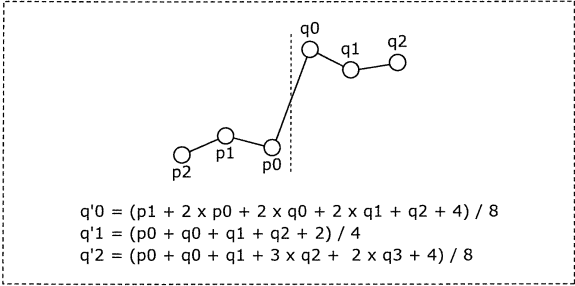
40

50

【図 7】



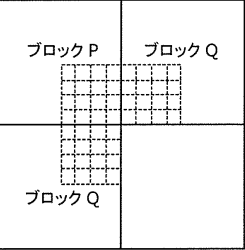
【図 8】



10

20

【図 9】



【図 10】

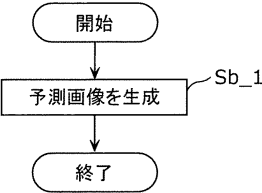
Bs 値を決定するブロック境界を挟むブロックPと Q の条件	Bs 値
・ 少なくとも一方のブロックが画面内予測ブロックである	2
・ 少なくとも片方のブロックに非零の直交変換係数が存在し、TU 境界である	1
・ 境界を挟む 2 つのブロックの動きベクトルの差の絶対値が 1 画素以上である	1
・ 境界を挟む 2 つのブロックの動き補償の参照画像が異なるか、動きベクトルの数が異なる	1
・ 上記以外	0

30

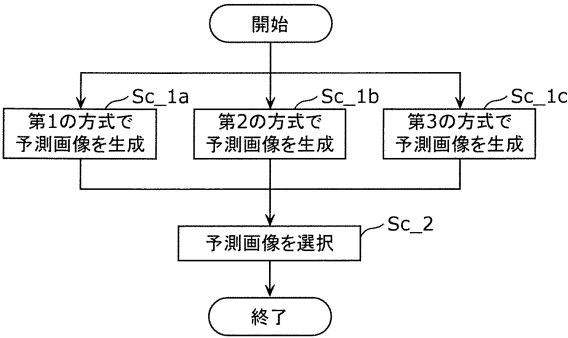
40

50

【図 1 1】

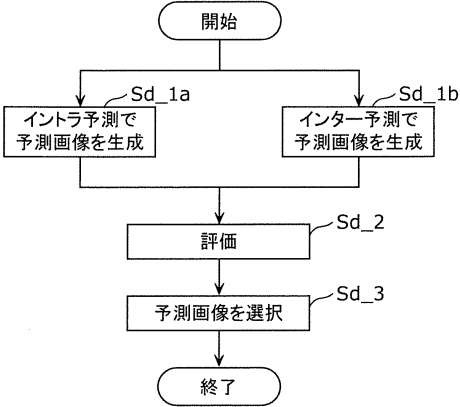


【図 1 2】

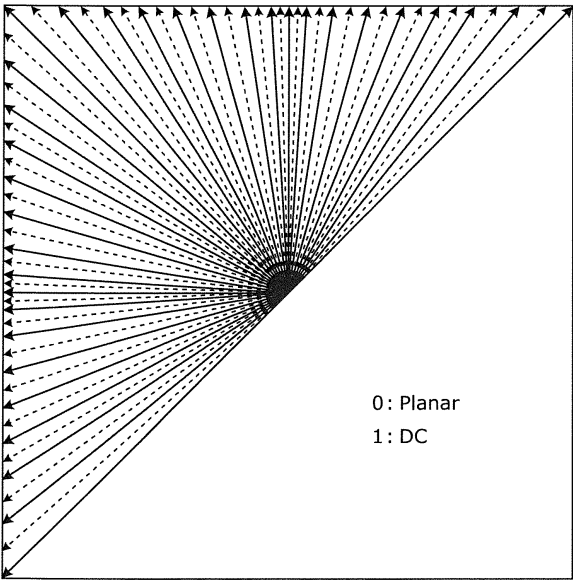


10

【図 1 3】



【図 1 4】



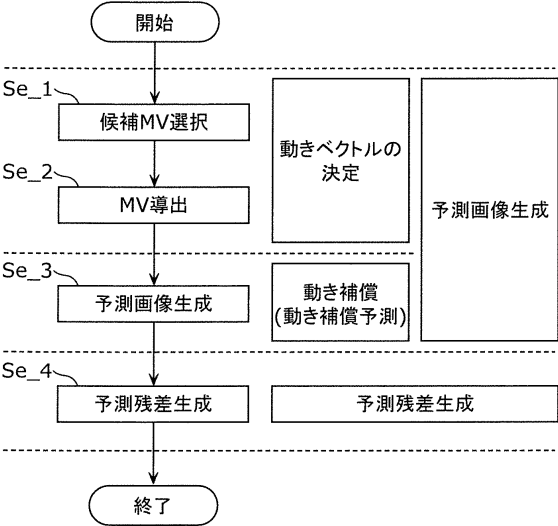
20

30

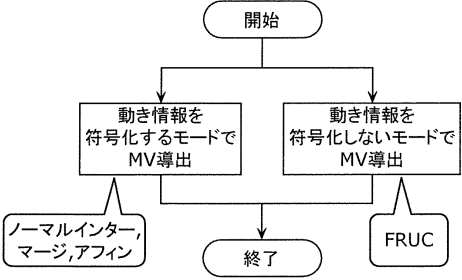
40

50

【図 15】

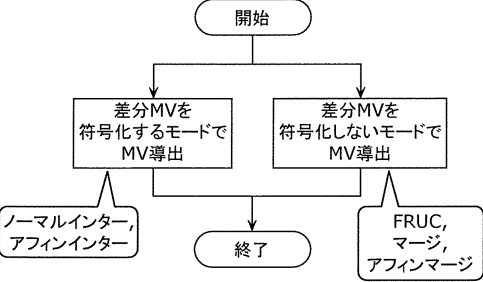


【図 16】

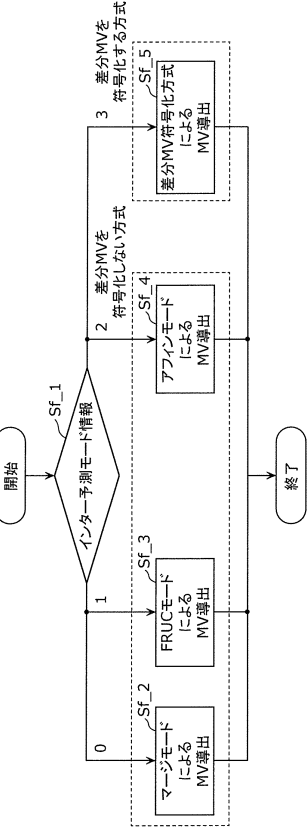


10

【図 17】



【図 18】



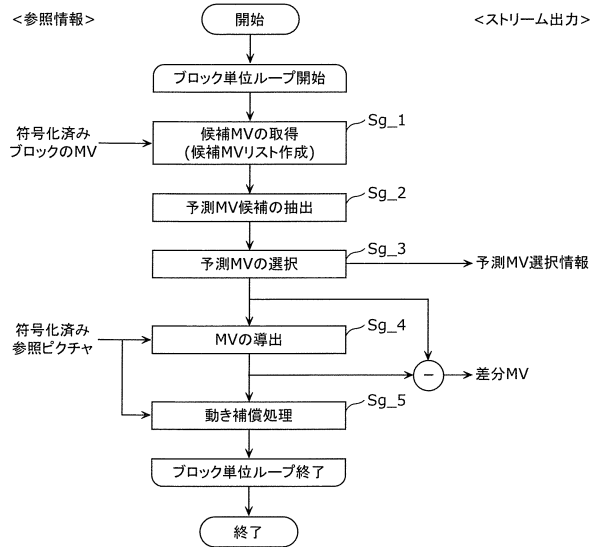
20

30

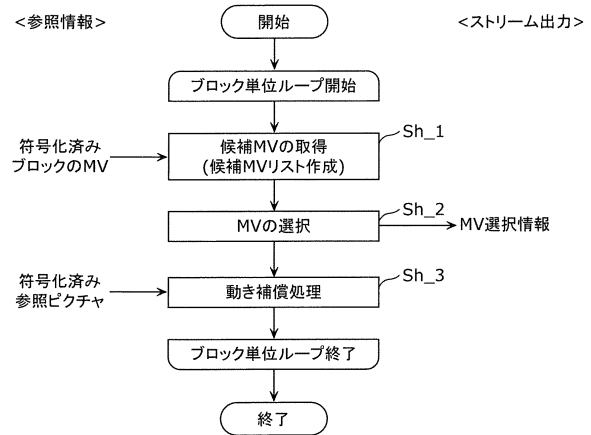
40

50

【図 19】

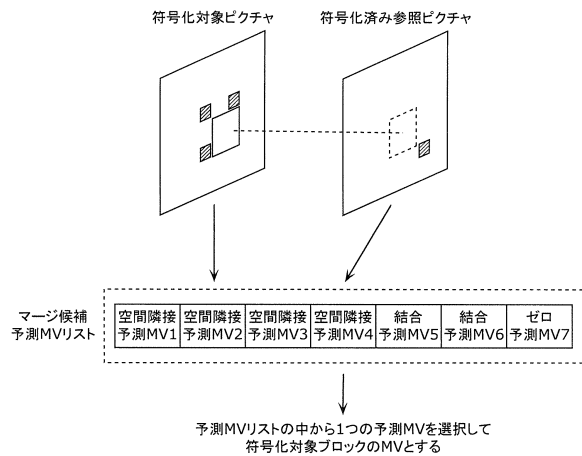


【図 20】

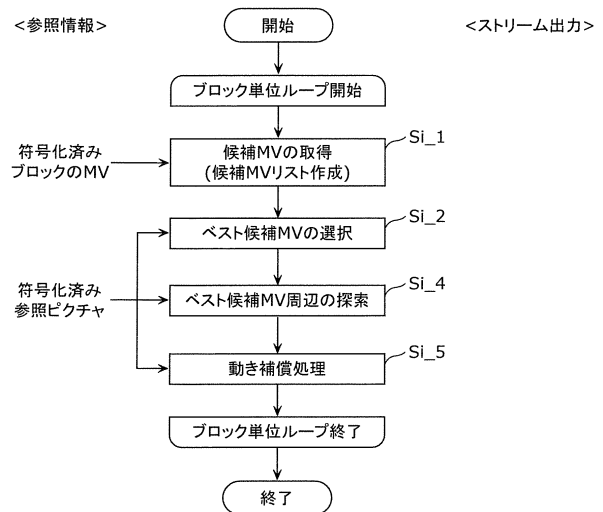


10

【図 21】



【図 22】



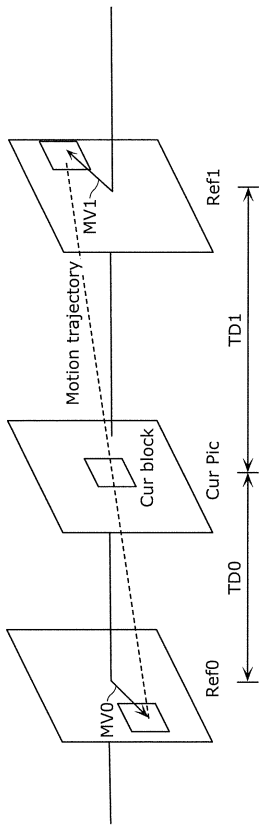
20

30

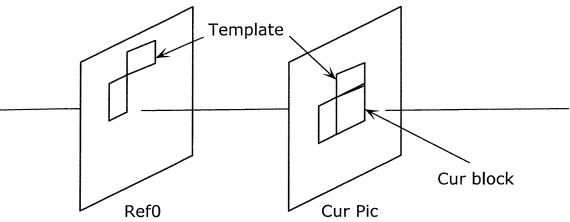
40

50

【図 2 3】



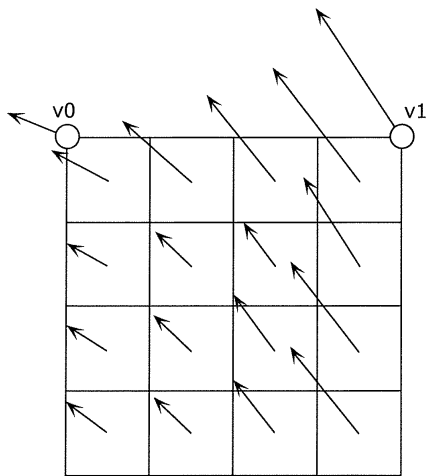
【図 2 4】



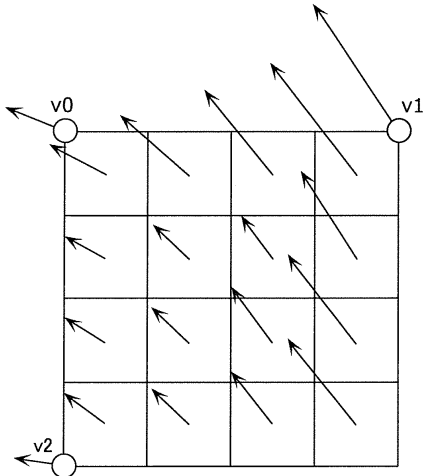
10

20

【図 2 5 A】



【図 2 5 B】

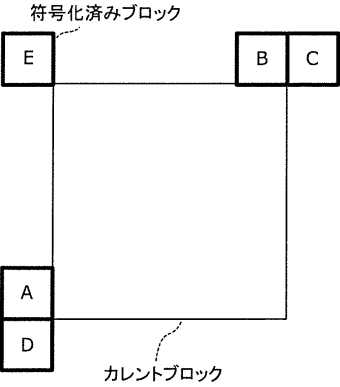


30

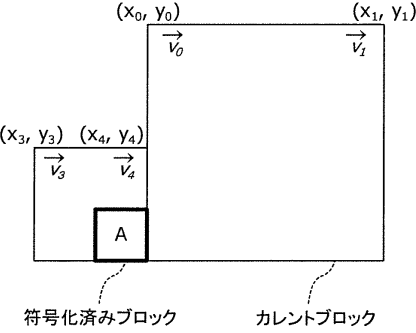
40

50

【図 2 6 A】

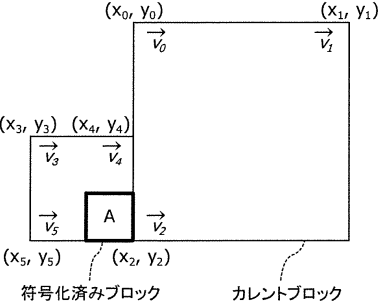


【図 2 6 B】

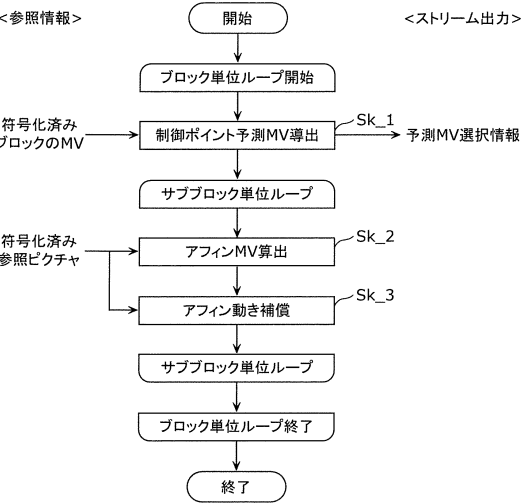


10

【図 2 6 C】



【図 2 7】



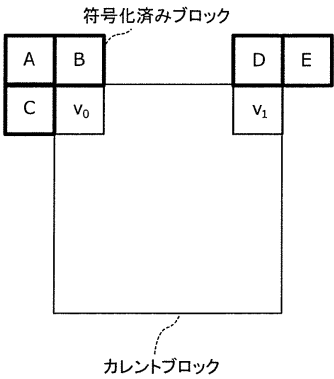
20

30

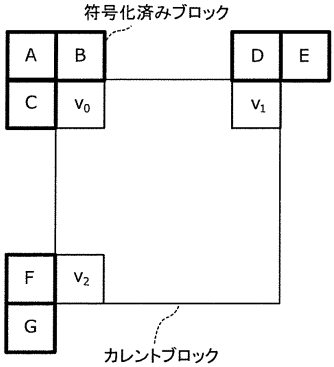
40

50

【図 2 8 A】

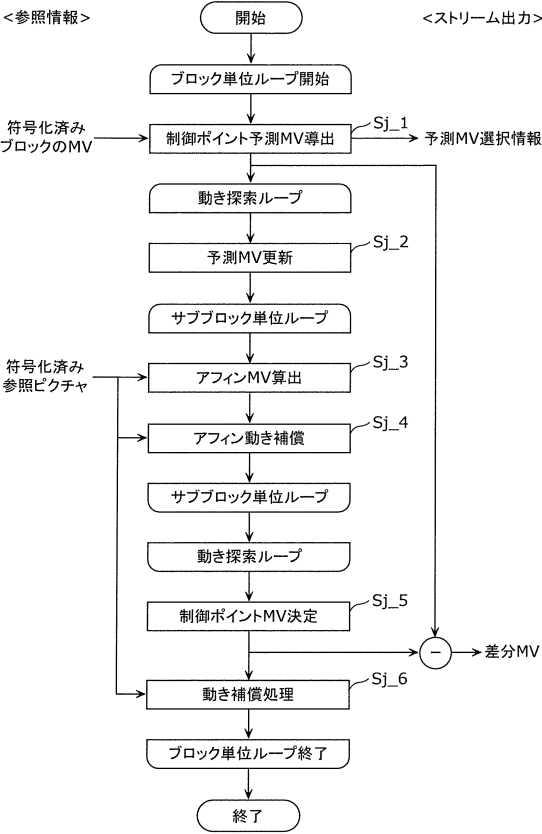


【図 2 8 B】

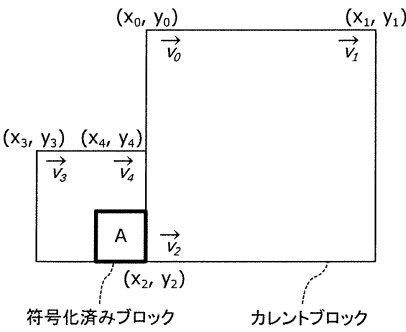


10

【図 2 9】



【図 3 0 A】



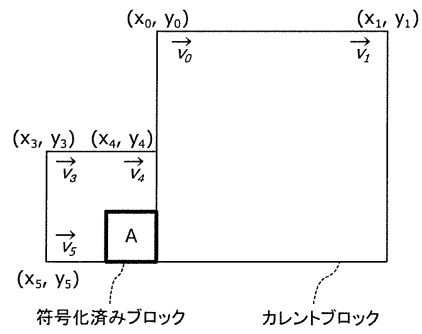
20

30

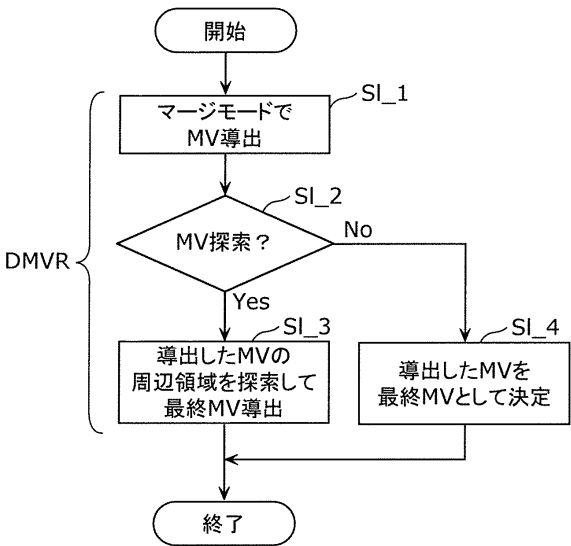
40

50

【図 3 0 B】

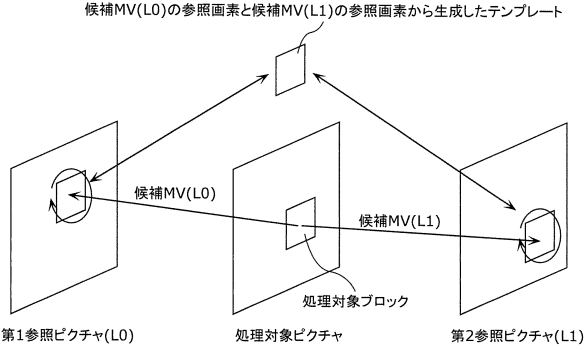


【図 3 1 A】

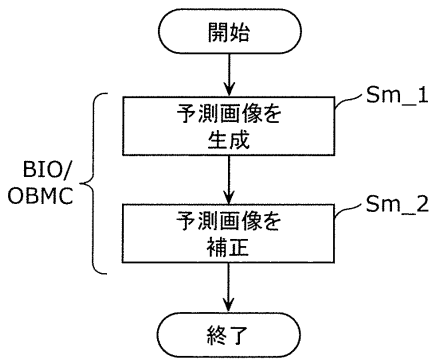


10

【図 3 1 B】



【図 3 2】



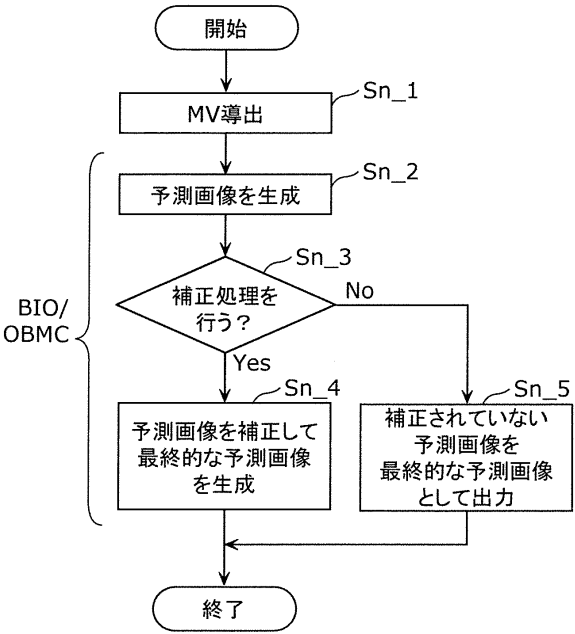
20

30

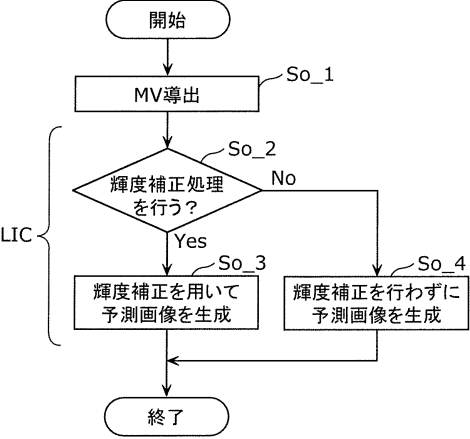
40

50

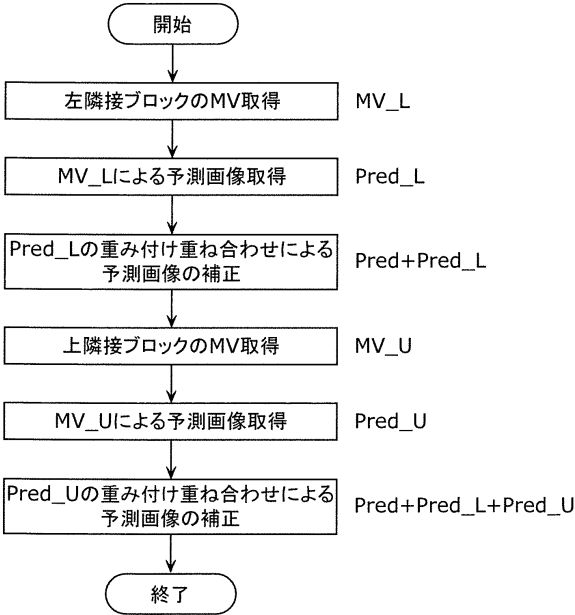
【図 3 3】



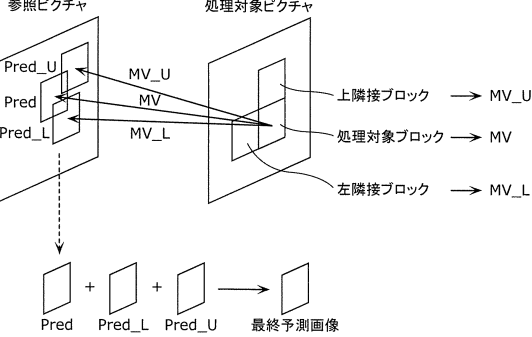
【図 3 4】



【図 3 5】



【図 3 6】



10

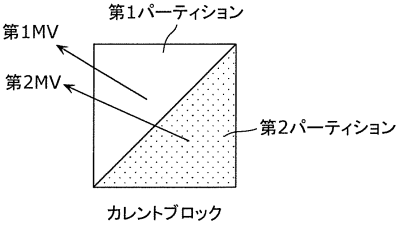
20

30

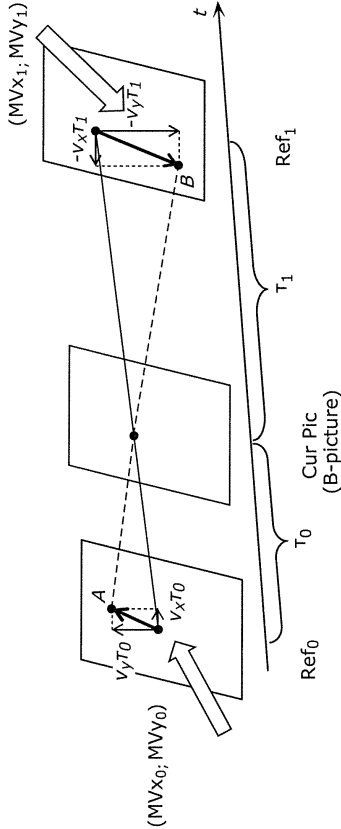
40

50

【図 3 7】



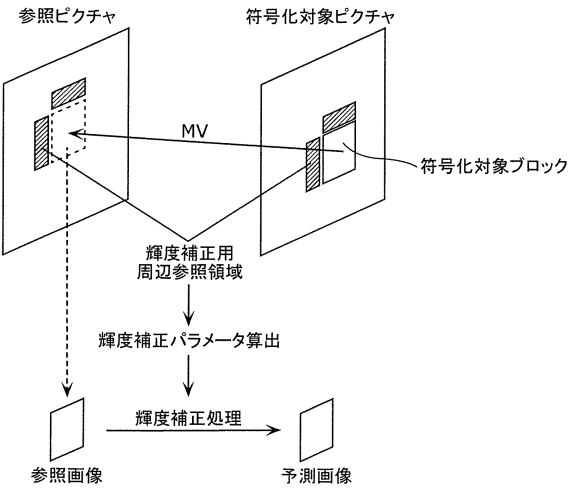
【図 3 8】



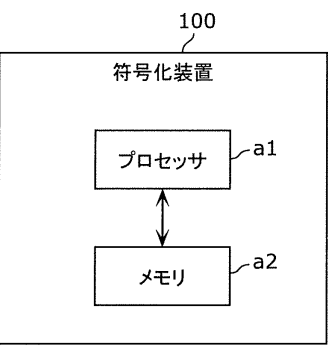
10

20

【図 3 9】



【図 4 0】

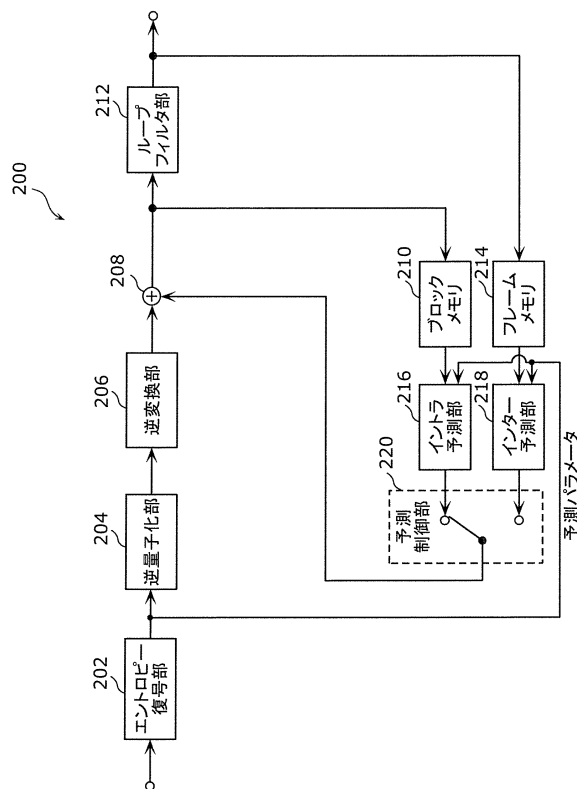


30

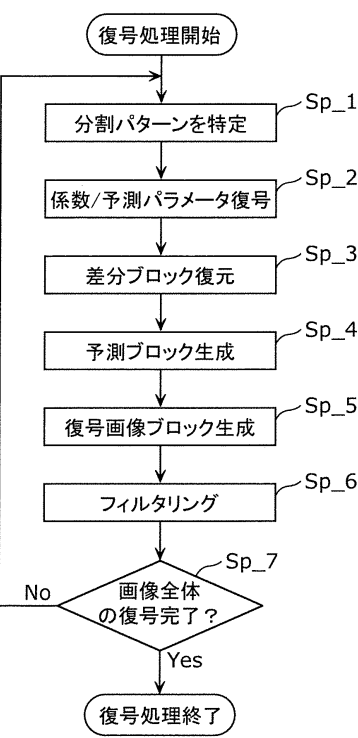
40

50

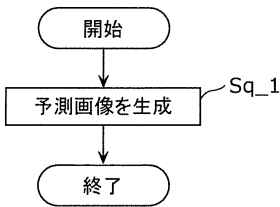
【図 4 1】



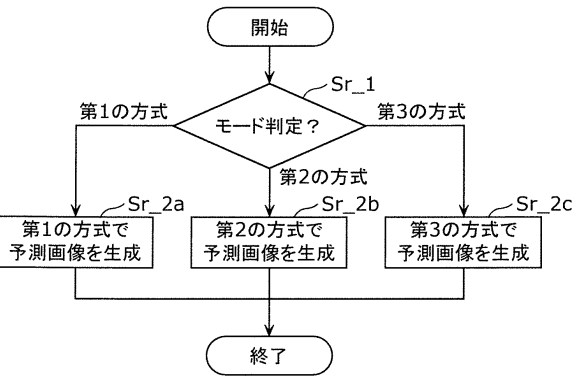
【図 4 2】



【図 4 3】



【図 4 4】



10

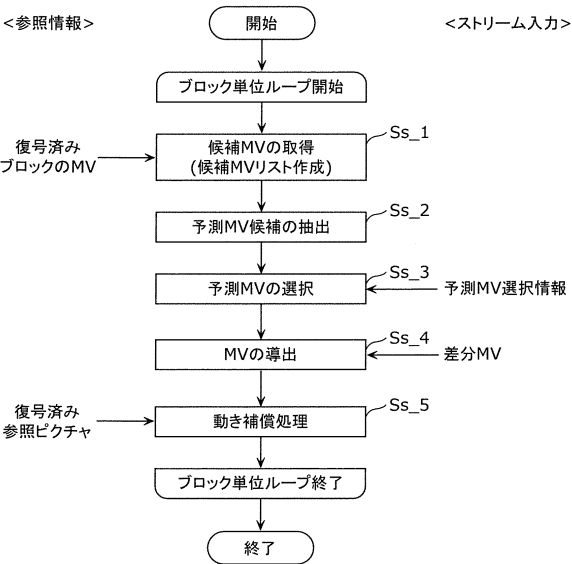
20

30

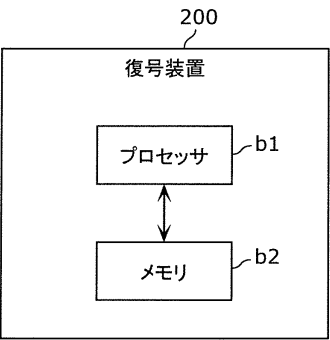
40

50

【図 4 5】

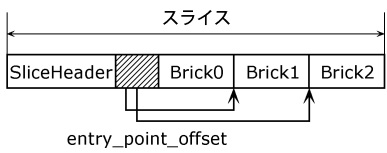


【図 4 6】



10

【図 4 7】



【図 4 8】

シンタックス	説明
slice_header(){ if(entry_point_offsetを符号化する条件を満たしている){ num_entry_point_offsets; offset_length; for(i=0;i<entry_point_offsetの個数;i++){ entry_point_offset; } } byte_alignment(); }	符号化するentry_point_offsetのビット長 entry_point_offsetの値 Byte境界に達するまでbitを埋める

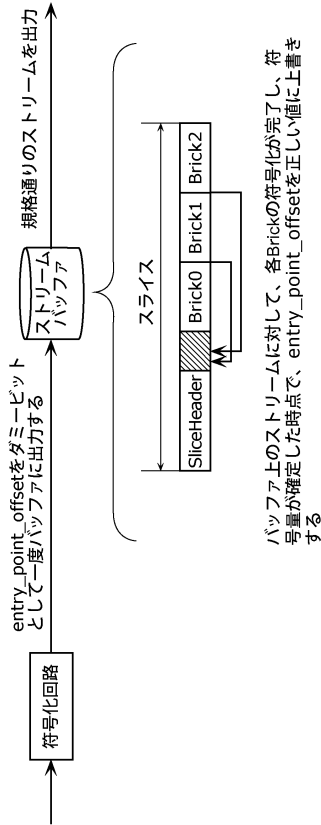
20

30

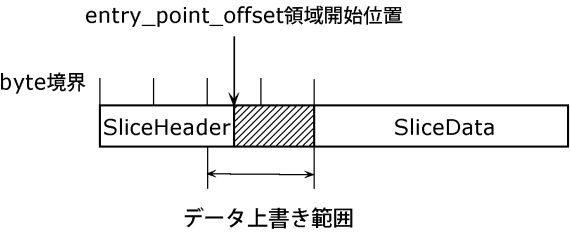
40

50

【図 49】



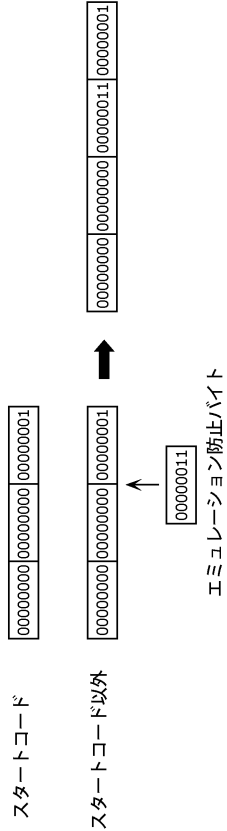
【図 50】



10

20

【図 51】



【図 52】

シンタックス	説明
slice_header(){	
if(entry_point_offsetを符号化する条件を満たしている){	
num_entry_point_offsets;	符号化するentry_point_offsetの個数
offset_length;	entry_point_offsetのビット長
byte_alignment();	Byte境界に達するまでbitを埋める
for(i=0;i<entry_point_offsetの個数;i++){	
entry_point_offset;	entry_point_offsetの値
entry_point_separate_one_bit;	entry_point_offset毎に値1の1bitを挿入
}	
byte_alignment();	Byte境界に達するまでbitを埋める
}	

30

40

50

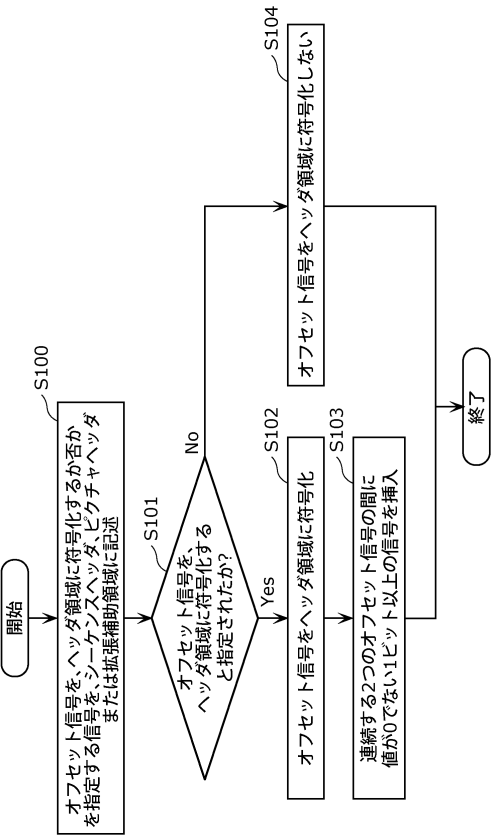
【図 5 3】

シンタックス	説明
slice_header(){ if(entry_point_offsetを符号化する条件を満たしている){ num_entry_point_offsets; offset_length; byte_alignment(); for(i=0;i<entry_point_offsetの個数;i++){ entry_point_offset_prefix; entry_point_separate_one_bit; entry_point_offset_suffix; entry_point_separate_one_bit; } } byte_alignment(); }	 符号化するentry_point_offsetの個数 entry_point_offsetのビット長 Byte境界に達するまでbitを埋める entry_point_offsetの上位bitのbit列 entry_point_offsetのprefix/suffix毎に値1の1bitを挿入 entry_point_offsetの低位bitのbit列 entry_point_offsetのprefix/suffix毎に値1の1bitを挿入 Byte境界に達するまでbitを埋める

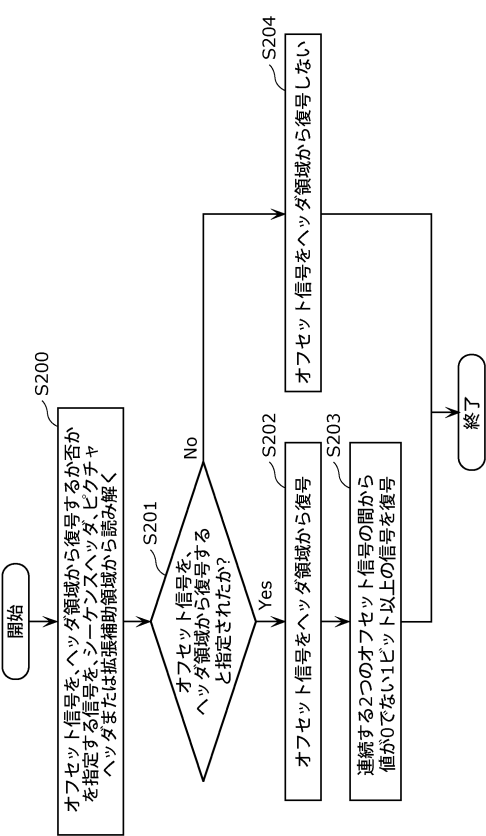
【図 5 4】

シンタックス	説明
slice_header(){ if(entry_point_offsetを符号化する条件を満たしている かつ、SPSで指定したpresent_flagが有効){ num_entry_point_offsets; offset_length; for(i=0;i<entry_point_offsetの個数;i++){ entry_point_offset; } } byte_alignment(); }	 符号化するentry_point_offsetの個数 entry_point_offsetのビット長 entry_point_offsetの値 Byte境界に達するまでbitを埋める

【図 5 5】



【図 5 6】



10

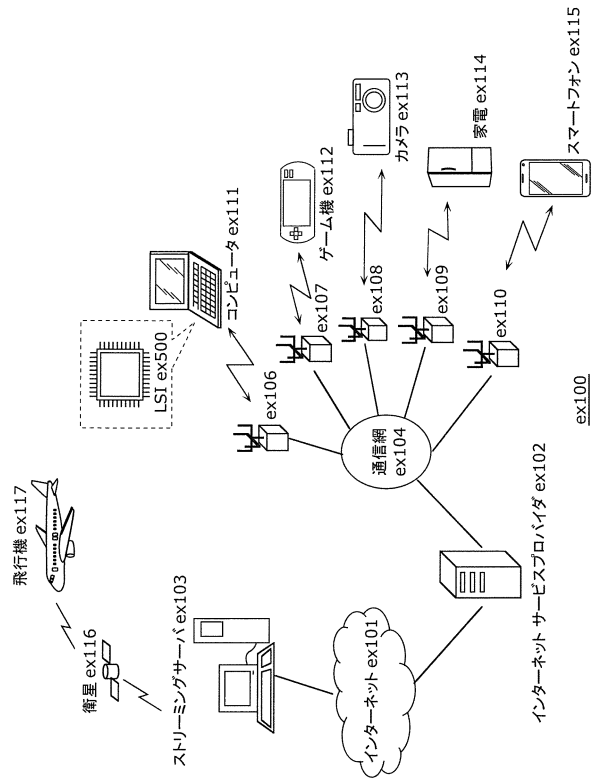
20

30

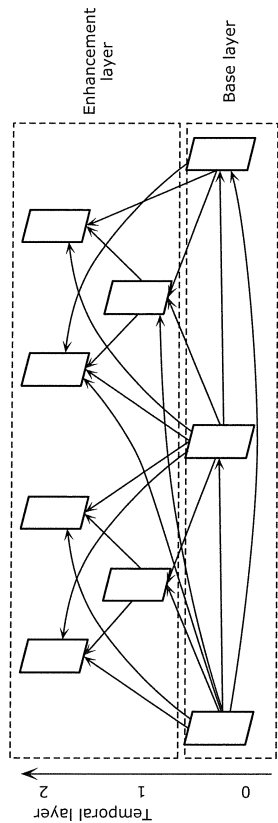
40

50

【図 5 7】



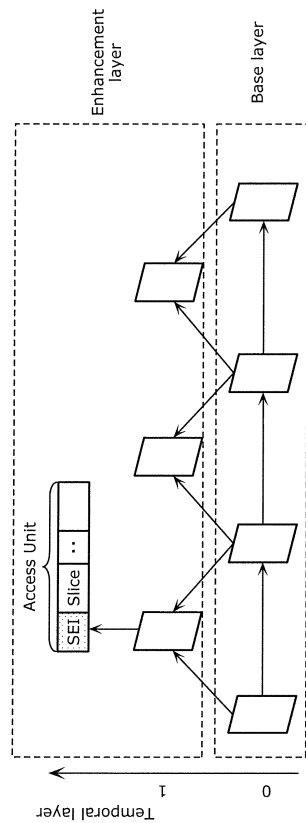
【図 5 8】



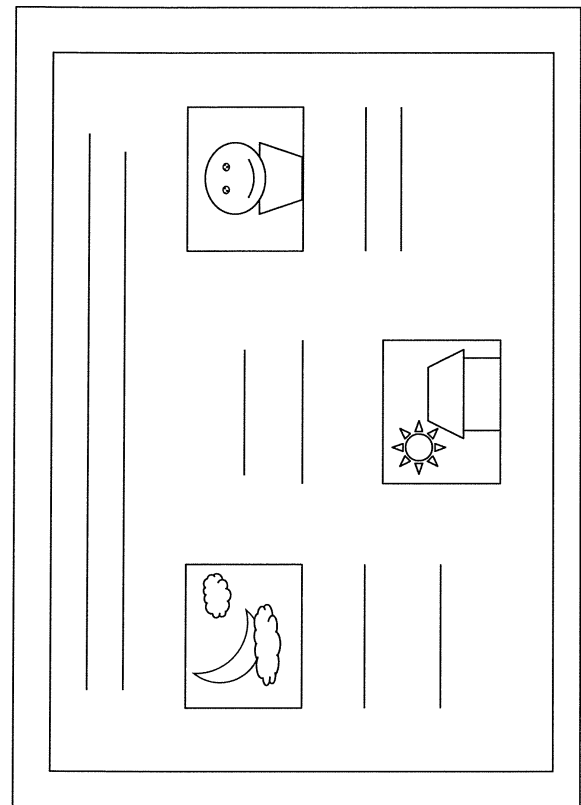
10

20

【図 5 9】



【図 6 0】

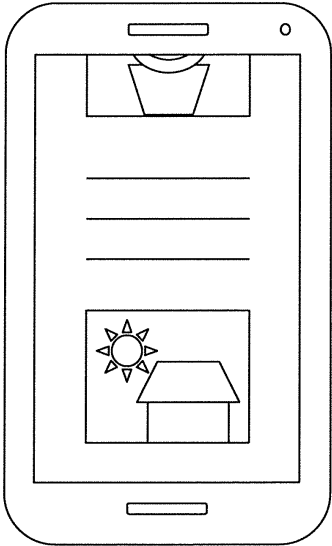


30

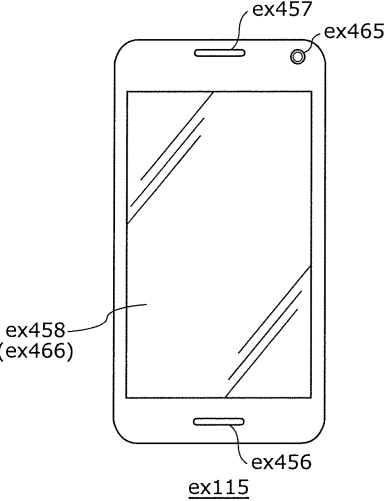
40

50

【図 6 1】

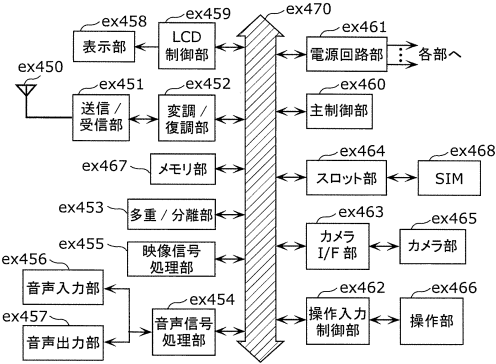


【図 6 2】



10

【図 6 3】



20

30

40

50

フロントページの続き

- (72)発明者 安倍 清史
日本国大阪府門真市大字門真１００６番地 パナソニック株式会社内
- (72)発明者 西 孝啓
日本国大阪府門真市大字門真１００６番地 パナソニック株式会社内
- (72)発明者 遠間 正真
日本国大阪府門真市大字門真１００６番地 パナソニック株式会社内
- (72)発明者 加藤 祐介
日本国大阪府門真市大字門真１００６番地 パナソニック株式会社内
- 審査官 坂東 大五郎
- (56)参考文献 国際公開第２０１３／１５８０１９（WO，A１）
Muhammed Coban et al., Tile entry point signalling, [JCTVC-I0357] (version 2), Joint Collaborative Team on Video Coding (JCT-VC), 2012年04月30日
Rickard Sjoberg et al., Tile groups for VVC, [JVET-L0415-v1] (version 1), Joint Video Experts Team (JVET), 2018年09月25日
- (58)調査した分野 (Int.Cl., DB名)
H 0 4 N １ 9 / 0 0 - 1 9 / 9 8