

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2005-234687

(P2005-234687A)

(43) 公開日 平成17年9月2日(2005.9.2)

(51) Int.CI.⁷

G06T 1/60
G06F 12/02
H04N 5/907
H04N 5/92
H04N 7/24

F 1

GO6T 1/60
 GO6F 12/02
 HO4N 5/907
 HO4N 5/92
 HO4N 7/13

450F
 540
 B
 H
 Z

テーマコード(参考)

5B047
 5B060
 5C052
 5C053
 5C059

審査請求 未請求 請求項の数 19 O L (全 35 頁)

(21) 出願番号
 (22) 出願日

特願2004-40177(P2004-40177)
 平成16年2月17日(2004.2.17)

(71) 出願人 000002185
 ソニー株式会社
 東京都品川区北品川6丁目7番35号
 (74) 代理人 100092152
 弁理士 服部 毅巖
 (72) 発明者 岡田 徹也
 東京都品川区北品川6丁目7番35号 ソ
 ニー株式会社内
 F ターム(参考) 5B047 EA01 EA05 EB02 EB15
 5B060 AA07 AC13 AC18
 5C052 AA17 CC11 GA07 GB06 GC09
 GE08 GF04
 5C053 FA27 GB06 GB37 HA33 LA01
 5C059 KK38 MA00 SS11 SS15 UA02
 UA05 UA36 UA38

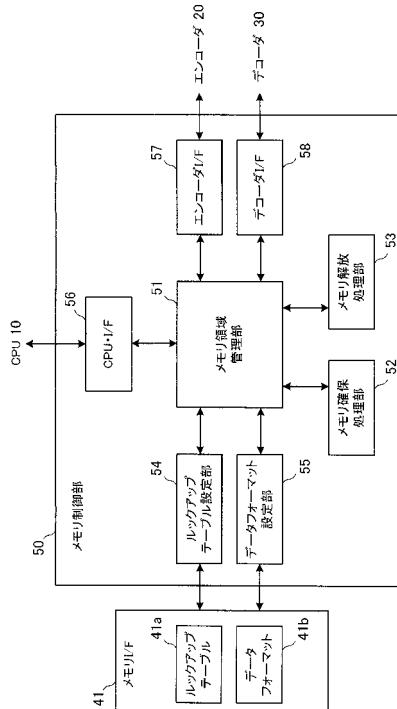
(54) 【発明の名称】メモリ管理方法、画像処理装置およびメモリ管理プログラム

(57) 【要約】

【課題】 1つの画像メモリを共用してエンコードおよびデコードを並行して実行する場合に、処理対象の画像サイズの変更に対して柔軟に対応することを可能にする。

【解決手段】 メモリ確保処理部52は、エンコードおよびデコードのそれぞれの処理のために個別に指定された画像サイズおよび画像枚数の入力を受けて、指定された画像サイズに対応するメモリ領域を画像メモリ上の空き領域に画像枚数分だけ個別に割り当てて確保し、確保したメモリ領域を示すアドレスを出力する。メモリ領域管理部51は、確保された各メモリ領域の識別情報と、メモリ領域にアクセスするためのアクセス情報との対応関係をメモリ領域ごとに記憶したメモリ領域情報を保持するとともに、各メモリ領域に対するエンコーダおよびデコーダからの使用要求および返却要求を受け付けて、各メモリ領域の使用状態をメモリ領域情報を用いて管理する。

【選択図】 図2



【特許請求の範囲】**【請求項 1】**

複数の画像処理手段により1つの画像メモリを作業領域として共用して、前記各画像処理手段による処理を並行して実行するためのメモリ管理方法において、

メモリ領域確保手段が、複数の前記画像処理手段のそれぞれの処理のために個別に指定される画像サイズおよび画像枚数の入力を受けて、前記画像サイズに対応するメモリ領域を前記画像メモリ上の空き領域に前記画像枚数分だけ個別に割り当てて確保し、確保した前記メモリ領域を示すアドレスを出力するメモリ領域確保ステップと、

メモリ領域管理手段が、前記メモリ領域確保ステップで確保された前記各メモリ領域の識別情報と、少なくとも前記アドレスを含む、前記メモリ領域にアクセスするためのアクセス情報との対応関係を、前記メモリ領域ごとに記憶したメモリ領域情報を生成するメモリ領域情報生成ステップと、

前記メモリ領域管理手段が、前記各メモリ領域に対する前記各画像処理手段からの使用要求および返却要求を受け付けて、前記各メモリ領域の使用状態を前記メモリ領域情報を用いて管理するメモリ領域管理ステップと、

を含むことを特徴とするメモリ管理方法。

【請求項 2】

前記画像処理手段の処理が停止されると、前記メモリ領域管理手段が、当該画像処理手段の処理のために確保された前記メモリ領域を再び空き領域に設定するとともに、対応する前記メモリ領域情報を消去するメモリ領域解放ステップをさらに含むことを特徴とする請求項1記載のメモリ管理方法。

【請求項 3】

前記メモリ領域管理ステップでは、対応する前記画像処理手段ごとに、当該画像処理手段により使用中の前記メモリ領域に対応する前記メモリ領域情報を含む第1の線形リストと、未使用の前記メモリ領域に対応する前記メモリ領域情報を含む第2の線形リストとを用いて、前記各メモリ領域の使用状態を管理することを特徴とする請求項1記載のメモリ管理方法。

【請求項 4】

前記メモリ領域情報生成ステップは、前記メモリ領域確保ステップで確保された前記メモリ領域に対応する前記メモリ領域情報を前記第2の線形リストに順次接続するステップを含み、

前記メモリ領域管理ステップは、前記画像処理手段からの前記使用要求を受けると、前記第2の線形リストの先頭の前記メモリ領域情報を取り出して、前記第1の線形リストの最後尾に接続するとともに、当該メモリ領域情報を識別するための識別番号を要求元の前記画像処理手段に出力するステップを含む、

ことを特徴とする請求項3記載のメモリ管理方法。

【請求項 5】

前記メモリ領域管理ステップは、前記画像処理手段から前記識別番号が指定された前記返却要求を受けると、指定された前記識別番号に該当する前記メモリ領域情報を前記第1の線形リストから取り出して、前記第2の線形リストの最後尾に接続するステップをさらに含むことを特徴とする請求項4記載のメモリ管理方法。

【請求項 6】

前記画像処理手段の処理が停止されると、前記メモリ領域管理手段が、対応する前記第2の線形リスト上の前記メモリ領域情報を消去するとともに、当該メモリ領域情報に対応する前記メモリ領域を空き領域に設定するメモリ領域解放ステップをさらに含むことを特徴とする請求項5記載のメモリ管理方法。

【請求項 7】

前記メモリ領域管理ステップは、前記画像処理手段から前記メモリ領域を指定した前記画像メモリに対するアクセス要求を受けると、指定された前記メモリ領域に対応する前記メモリ領域情報をにより対応付けられた前記アクセス情報を用いて前記画像メモリにアクセ

スし、指定された前記メモリ領域と要求元の前記画像処理装置との間で画像データを転送するステップを含むことを特徴とする請求項1記載のメモリ管理方法。

【請求項8】

前記メモリ領域確保ステップでは、前記画像メモリ上の単位メモリ領域を二分木構造の最下層のノードに対応付けて管理し、前記画像サイズが指定されると、指定された前記画像サイズ分の画像データを格納可能な1つ以上の前記単位メモリ領域が最下層の子ノードに対応付けられている前記二分木構造内のノードを抽出し、抽出したノードを示すノード番号を前記メモリ領域管理手段に出力することで、指定された前記画像サイズに対応する前記メモリ領域を確保し、

前記メモリ領域管理ステップでは、前記メモリ領域確保ステップで出力された前記ノード番号を前記メモリ領域情報内に記憶する、
10

ことを特徴とする請求項1記載のメモリ管理方法。

【請求項9】

前記メモリ領域確保ステップでは、確保した前記メモリ領域を示す前記アドレスとして当該メモリ領域の先頭アドレスを出力し、

前記アクセス情報は、前記先頭アドレスと、当該先頭アドレスにより示される前記メモリ領域に対応する画像サイズとを含むことを特徴とする請求項1記載のメモリ管理方法。

【請求項10】

複数の画像処理手段により1つの画像メモリを作業領域として共用して、前記各画像処理手段による処理を並行して実行することが可能な画像処理装置において、
20

複数の前記画像処理手段のそれぞれの処理のために個別に指定される画像サイズおよび画像枚数の入力を受けて、前記画像サイズに対応するメモリ領域を前記画像メモリ上の空き領域に前記画像枚数分だけ個別に割り当てて確保し、確保した前記メモリ領域を示すアドレスを出力するメモリ領域確保手段と、

前記メモリ領域確保手段により確保された前記各メモリ領域の識別情報と、少なくとも前記アドレスを含む、前記メモリ領域にアクセスするためのアクセス情報との対応関係を、前記メモリ領域ごとに記憶したメモリ領域情報を保持するとともに、前記各メモリ領域に対する前記各画像処理手段からの使用要求および返却要求を受け付けて、前記各メモリ領域の使用状態を前記メモリ領域情報を用いて管理するメモリ領域管理手段と、

を有することを特徴とする画像処理装置。
30

【請求項11】

前記メモリ領域管理手段は、前記画像処理手段の処理が停止されると、当該画像処理手段の処理のために確保された前記メモリ領域の解放要求を前記メモリ領域確保手段に出力して、対応する前記メモリ領域情報を消去し、

前記メモリ領域確保手段は、前記メモリ領域管理手段からの前記解放要求を受けると、対応する前記メモリ領域を再び空き領域に設定する、

ことを特徴とする請求項10記載の画像処理装置。

【請求項12】

前記メモリ領域管理手段は、対応する前記画像処理手段ごとに、当該画像処理手段により使用中の前記メモリ領域に対応する前記メモリ領域情報を含む第1の線形リストと、未使用的前記メモリ領域に対応する前記メモリ領域情報を含む第2の線形リストとを用いて、前記各メモリ領域の使用状態を管理することを特徴とする請求項10記載の画像処理装置。
40

【請求項13】

前記メモリ領域管理手段は、前記メモリ領域確保手段により確保された前記メモリ領域に対応する前記メモリ領域情報を前記第2の線形リストに順次接続し、前記画像処理手段からの前記使用要求を受けると、前記第2の線形リストの先頭の前記メモリ領域情報を取り出して、前記第1の線形リストの最後尾に接続するとともに、当該メモリ領域情報を識別するための識別番号を要求元の前記画像処理手段に出力することを特徴とする請求項12記載の画像処理装置。
50

【請求項 14】

前記メモリ領域管理手段は、前記画像処理手段から前記識別番号が指定された前記返却要求を受けると、指定された前記識別番号に該当する前記メモリ領域情報を前記第1の線形リストから取り出して、前記第2の線形リストの最後尾に接続することを特徴とする請求項13記載の画像処理装置。

【請求項 15】

前記メモリ領域管理手段は、前記画像処理手段の処理が停止されると、対応する前記第2の線形リスト上の前記メモリ領域情報に対応する前記メモリ領域の解放要求を前記メモリ領域確保手段に出力して、当該メモリ領域情報を消去し、

前記メモリ領域確保手段は、前記メモリ領域管理手段から前記解放要求に応じて、対応する前記メモリ領域を空き領域に設定する、

ことを特徴とする請求項14記載の画像処理装置。

【請求項 16】

前記メモリ領域管理手段は、前記画像処理手段から前記メモリ領域を指定した前記画像メモリに対するアクセス要求を受けると、指定された前記メモリ領域に対応する前記メモリ領域情報により対応付けられた前記アクセス情報を用いて前記画像メモリにアクセスし、指定された前記メモリ領域と要求元の前記画像処理装置との間で画像データを転送することを特徴とする請求項10記載の画像処理装置。

【請求項 17】

前記メモリ領域確保手段は、前記画像メモリ上の単位メモリ領域を二分木構造の最下層のノードに対応付けて管理し、前記画像サイズが指定されると、指定された前記画像サイズ分の画像データを格納可能な1つ以上の前記単位メモリ領域が最下層の子ノードに対応付けられている前記二分木構造内のノードを抽出し、抽出したノードを示すノード番号を前記メモリ領域管理手段に出力することで、指定された前記画像サイズに対応する前記メモリ領域を確保し、

前記メモリ領域管理手段は、前記メモリ領域確保手段からの前記ノード番号を前記メモリ領域情報内に記憶する、

ことを特徴とする請求項10記載の画像処理装置。

【請求項 18】

前記メモリ領域確保手段は、確保した前記メモリ領域を示す前記アドレスとして当該メモリ領域の先頭アドレスを出力し、

前記アクセス情報は、前記先頭アドレスと、当該先頭アドレスにより示される前記メモリ領域に対応する画像サイズとを含むことを特徴とする請求項10記載の画像処理装置。

【請求項 19】

複数の画像処理手段により1つの画像メモリを作業領域として共用して、前記各画像処理手段による処理を並行して実行するためのメモリ管理処理をコンピュータに実行させるメモリ管理プログラムにおいて、

複数の前記画像処理手段のそれぞれの処理のために個別に指定される画像サイズおよび画像枚数の入力を受けて、前記画像サイズに対応するメモリ領域を前記画像メモリ上の空き領域に前記画像枚数分だけ個別に割り当てて確保し、確保した前記メモリ領域を示すアドレスを出力するメモリ領域確保ステップと、

前記メモリ領域確保ステップで確保された前記各メモリ領域の識別情報と、少なくとも前記アドレスを含む、前記メモリ領域にアクセスするためのアクセス情報との対応関係を、前記メモリ領域ごとに記憶したメモリ領域情報を生成するメモリ領域情報生成ステップと、

前記各メモリ領域に対する前記各画像処理手段からの使用要求および返却要求を受け付けて、前記各メモリ領域の使用状態を前記メモリ領域情報を用いて管理するメモリ領域管理ステップと、

を含む処理を前記コンピュータに実行させることを特徴とするメモリ管理プログラム。

【発明の詳細な説明】

10

20

30

40

50

【技術分野】**【0001】**

本発明は、複数の画像処理手段により1つの画像メモリを作業領域として共用して、各画像処理手段による処理を並行して実行するためのメモリ管理方法、このメモリ管理方法を用いた画像処理装置、およびメモリ管理プログラムに関する。

【背景技術】**【0002】**

近年、MPEG (Moving Picture Experts Group) 方式などの画像圧縮技術の進歩などに伴い、画像信号をデジタル化して取り扱うことが一般的になっている。例えば、動画像を撮像してデジタルデータとして記録するデジタルビデオカメラなどが広く普及している。
10

【0003】

また、最近のデジタルビデオカメラの中には、磁気テープと、可搬型の半導体メモリからなるメモリカードのように、2種類の記録媒体を具備するものがある。このようなデジタルビデオカメラでは、動画像のエンコードとデコードや、2系統のエンコードあるいはデコードを並行して実行する機能を具備するものが多い。

【0004】

例えば、撮像中の動画像を、異なる画像サイズや圧縮率、フォーマットでそれぞれの記録媒体に同時に記録する機能を具備するものがある。この場合、画像サイズの異なる2系統の動画像のエンコードが並行して実行される。また、磁気テープに記録された動画像を一旦デコードし、画像サイズを小さくして再度エンコードしてメモリカードに記録する機能を具備するものがある。この場合、デコードとエンコードとが並行して実行される。
20

【0005】

図25は、一例として、動画像のエンコードとデコードを並行して実行することが可能な画像処理装置の要部構成を示す図である。

図25に示す画像処理装置は、例えば、動画像データおよび音声データの入力を受けてこれらをMPEGなどの圧縮符号化方式でエンコードし、各種の記録媒体に記録する機能と、記録媒体の記録データをデコードし、動画像および音声を再生出力する機能とを具備する画像記録再生装置や撮像装置などに設けられる。この画像処理装置では、動画像および音声のエンコードとデコードとを並行して実行することが可能となっており、これらの処理が、OS (Operating System) を搭載しない、いわゆる組み込み型システムにより実行されるようになっている。
30

【0006】

この画像処理装置は、図25に示すように、エンコード/デコードの制御のためのCPU11、エンコーダ21、デコーダ31、および処理中の画像データの一時保存のためにエンコーダ21およびデコーダ31により共用される画像メモリ40を具備している。

【0007】

CPU11は、外部のホストコントローラと通信し、このホストコントローラからの情報に応じてエンコーダ21およびデコーダ31の処理を制御する。具体的には、ホストコントローラから、エンコード/デコードの命令、生成する画像のサイズ、確保すべきメモリ領域の数などの情報を受け取って、対応するエンコーダ21またはデコーダ31にこれらの情報を出力し、処理を開始させる。
40

【0008】

エンコーダ21は、入力された動画像および音声のデジタルデータに対して、MPEG方式などの所定の圧縮符号化方式に従って圧縮符号化処理を施し、動画像および音声の符号化ストリームを生成する処理を実行する。また、デコーダ31は、入力された動画像および音声の符号化ストリームに対して伸張復号化処理を施す。エンコーダ21およびデコーダ31は、動画像データのエンコード、デコードをそれぞれ実行する際に、画像メモリ40にアクセスして、処理中の画像データを画像メモリ40に一時的に格納する。

【0009】

画像メモリ40は、例えばS D R A M (Synchronous Dynamic Random Access Memory)などの半導体メモリからなり、その内部に、外部からのアクセスを受けるメモリI / F 41を具備している。メモリI / F 41は、エンコーダ21およびデコーダ31から供給された画像データを、画像メモリ40の指定された領域に格納する。また、エンコーダ21およびデコーダ31から指定された読み出しあドレスに応じて、画像メモリ40から画像データを読み出して出力する。

【 0 0 1 0 】

このような画像処理装置において、動画像のエンコードが行われる場合は、C P U 1 1は、ホストコントローラからの情報に基づいて、エンコーダ21に対してエンコードの開始を命令するとともに、画像サイズや、確保すべきメモリ領域に対応する画像の枚数を指定する。エンコーダ21は、動画像データの入力を受けると、C P U 1 1からの情報に従って、画像メモリ40に対して画像を格納するメモリ領域を確保し、対応する物理アドレスをメモリI / F 41に指定して、処理中の画像データをそのメモリ領域に順次格納しながらエンコードを実行する。

【 0 0 1 1 】

また、動画像のデコードが行われる場合は、同様に、C P U 1 1からデコーダ31に対して画像サイズや画像の枚数がデコーダ31に対して指定され、符号化ストリームの入力を受けたデコーダ31は、C P U 1 1により指定された枚数に対応するメモリ領域を画像メモリ40に確保し、対応する物理アドレスをメモリI / F 41に指定して、処理中の画像データをそのメモリ領域に順次格納しながらデコードを実行する。

【 0 0 1 2 】

また、音声のエンコードおよびデコードが行われる際には、エンコーダ21およびデコーダ31は、ホストコントローラからC P U 1 1を通じて供給される情報に従って、入力される音声データや符号化ストリームを順次処理していく。この際に、処理中の音声データを画像メモリ40に一時的に格納するようにしてもよい。

【 0 0 1 3 】

なお、1系統のデータの符号化処理と、他の1系統のデータの復号処理とを並行して実行することが可能な従来の装置として、以下のような画像の復号装置があった。この復号装置では、データの符号化および復号が共用して行われる逆変換回路が設けられている。そして、高能率符号化された画像信号として入力されて可変長復号回路および逆量子化回路を経由した信号と、入力された別系統の画像信号とが、多重化回路により選択されて逆変換回路に入力され、前者の信号は逆変換回路において復号されて出力され、後者の画像信号は、逆変換回路において符号化された後、さらに量子化回路および可変長符号化回路を介して出力される。ここで、逆変換回路を実時間速度の2倍またはそれ以上の速度で動作させることにより、逆変換回路における復号機能と符号化機能とを切り替えて使用し、復号処理と符号化処理とを並行して実行することが可能となる（例えば、特許文献1参照）。

【特許文献1】特開平9-322121号公報（段落番号〔0029〕～〔0032〕、図5）

【発明の開示】

【発明が解決しようとする課題】

【 0 0 1 4 】

ところで、複数の記録媒体を具備する画像記録再生装置では、各記録媒体を用いた動画像データの記録または再生を行っている途中で、エンコーダまたはデコーダの処理対象となる画像サイズを変更可能とする要求がある。例えば、磁気テープとメモリカードとを具備するデジタルビデオカメラにおいて、撮像画像を各記録媒体に記録している途中で、メモリカードへの記録動作のみを中断し、記録画像の画像サイズを変更して記録動作を再開させる場合などが想定される。

【 0 0 1 5 】

しかし、図25に示した従来の画像処理装置では、エンコーダ21およびデコーダ31

により画像メモリ40に確保されるメモリ領域の大きさや数は、ホストコントローラからCPU11を介して指定される、処理対象とする画像サイズに応じて、あらかじめ決められている。このため、一旦エンコードおよびデコードが開始されると、画像メモリ40内にメモリ領域が固定的に確保され、すべてのエンコードおよびデコードの動作を停止させない限り、画像サイズの設定を変更することができなかった。また、エンコーダ21やデコーダ31は、確保したメモリ領域を、その物理アドレスを常に認識して管理する必要があるため、画像サイズなどを変更した際にメモリ領域の管理が煩雑となり、画像サイズの変更に対して柔軟に対応できないことが問題であった。

【0016】

本発明はこのような点に鑑みてなされたものであり、1つの画像メモリを共用してエンコードおよびデコードを並行して実行する場合に、処理対象の画像サイズの変更に対して柔軟に対応することが可能なメモリ管理方法を提供することを目的とする。10

【0017】

また、本発明の他の目的は、1つの画像メモリを共用してエンコードおよびデコードを並行して実行する場合に、処理対象の画像サイズの変更に対して柔軟に対応することが可能な画像処理装置を提供することである。

【0018】

さらに、本発明の他の目的は、1つの画像メモリを共用してエンコードおよびデコードを並行して実行する場合に、処理対象の画像サイズの変更に対して柔軟に対応することが可能なメモリ管理プログラムを提供することである。20

【課題を解決するための手段】

【0019】

本発明では上記課題を解決するために、複数の画像処理手段により1つの画像メモリを作業領域として共用して、前記各画像処理手段による処理を並行して実行するためのメモリ管理方法において、メモリ領域確保手段が、複数の前記画像処理手段のそれぞれの処理のために個別に指定される画像サイズおよび画像枚数の入力を受けて、前記画像サイズに対応するメモリ領域を前記画像メモリ上の空き領域に前記画像枚数分だけ個別に割り当てて確保し、確保した前記メモリ領域を示すアドレスを出力するメモリ領域確保ステップと、メモリ領域管理手段が、前記メモリ領域確保ステップで確保された前記各メモリ領域の識別情報と、少なくとも前記アドレスを含む、前記メモリ領域にアクセスするためのアクセス情報との対応関係を、前記メモリ領域ごとに記憶したメモリ領域情報を生成するメモリ領域情報生成ステップと、前記メモリ領域管理手段が、前記各メモリ領域に対する前記各画像処理手段からの使用要求および返却要求を受け付けて、前記各メモリ領域の使用状態を前記メモリ領域情報を用いて管理するメモリ領域管理ステップとを含むことを特徴とするメモリ管理方法が提供される。30

【0020】

このようなメモリ管理方法では、メモリ領域確保手段によるメモリ領域確保ステップにおいて、指定された画像サイズに対応するメモリ領域が、指定された画像枚数分だけ画像メモリ上の空き領域内に個別に確保されるので、各メモリ領域は、画像メモリ上に画像枚数分だけ固定的に割り当てられず、それぞれが個別に分散して割り当てられる。また、メモリ領域管理手段によるメモリ領域情報生成ステップにおいて、確保された各メモリ領域の識別情報と、メモリ領域にアクセスするためのアクセス情報との対応関係をメモリ領域ごとに記憶したメモリ領域情報を生成する。これにより、メモリ領域情報を識別する番号などを各画像処理手段が指定することで、対応するメモリ領域にアクセスし、画像データの書き込みおよび読み出しを行うことが可能となる。さらに、メモリ領域管理手段によるメモリ領域管理ステップにおいて、各メモリ領域に対する各画像処理手段からの使用要求および返却要求を受け付けて、各メモリ領域の使用状態をメモリ領域情報を用いて管理することにより、各画像処理手段は、メモリ領域を使用する際に、対応する画像メモリ内のアドレスなどを認識して使用状態を管理する必要がなくなる。40

【0021】

10

20

30

40

50

また、本発明では、複数の画像処理手段により1つの画像メモリを作業領域として共用して、前記各画像処理手段による処理を並行して実行することが可能な画像処理装置において、複数の前記画像処理手段のそれぞれの処理のために個別に指定される画像サイズおよび画像枚数の入力を受けて、前記画像サイズに対応するメモリ領域を前記画像メモリ上の空き領域に前記画像枚数分だけ個別に割り当てて確保し、確保した前記メモリ領域を示すアドレスを出力するメモリ領域確保手段と、前記メモリ領域確保手段により確保された前記各メモリ領域の識別情報と、少なくとも前記アドレスを含む、前記メモリ領域にアクセスするためのアクセス情報との対応関係を、前記メモリ領域ごとに記憶したメモリ領域情報を保持するとともに、前記各メモリ領域に対する前記各画像処理手段からの使用要求および返却要求を受け付けて、前記各メモリ領域の使用状態を前記メモリ領域情報を用いて管理するメモリ領域管理手段とを有することを特徴とする画像処理装置が提供される。
10

【0022】

ここで、メモリ領域確保手段は、指定された画像サイズに対応するメモリ領域を、指定された画像枚数分だけ画像メモリ上の空き領域内に個別に確保するので、各メモリ領域は、画像メモリ上に画像枚数分だけ固定的に割り当てられず、それぞれが個別に分散して割り当てられる。また、メモリ領域管理手段は、確保された各メモリ領域の識別情報と、メモリ領域にアクセスするためのアクセス情報との対応関係をメモリ領域ごとに記憶したメモリ領域情報を生成する。これにより、メモリ領域情報を識別する番号などを各画像処理手段が指定することで、対応するメモリ領域にアクセスし、画像データの書き込みおよび読み出しを行うことが可能となる。さらに、各メモリ領域に対する各画像処理手段からの使用要求および返却要求を受け付けて、各メモリ領域の使用状態をメモリ領域情報を用いて管理することにより、各画像処理手段は、メモリ領域を使用する際に、対応する画像メモリ内のアドレスなどを認識して使用状態を管理する必要がなくなる。
20

【発明の効果】

【0023】

本発明によれば、複数の画像処理手段のそれぞれが、画像メモリ内のメモリ領域のアドレスなどを認識して使用状態を管理することなく、単にメモリ領域情報を指定することでき、対応するメモリ領域にアクセスし、画像データの書き込みおよび読み出しを行うことが可能となる。また、画像サイズに対応するメモリ領域が画像メモリ内に個別に割り当てられるので、画像サイズを変更した場合にも、割り当てた領域を一旦解放して空き領域に設定し、その空き領域を異なる画像サイズに対応するメモリ領域として容易に再利用することが可能となる。従って、各画像処理手段の処理負担を高めることなく、ある画像処理手段において画像サイズが変更された場合にも、他の画像処理手段の処理を中断せずに、必要なメモリ領域を確保し直して利用させることが可能となる。
30

【発明を実施するための最良の形態】

【0024】

以下、本発明の実施の形態を図面を参照して詳細に説明する。

[第1の実施の形態の構成]

以下の第1の実施の形態では、動画像および音声の信号を、所定の圧縮符号化方式に従ってエンコード／デコードする画像処理装置に、本発明を適用した場合を想定する。図1は、本発明の第1の実施の形態に係る画像処理装置の要部構成を示すブロック図である。
40

【0025】

図1に示す画像処理装置は、例えば、動画像データおよび音声データの入力を受けてこれらをMPEGなどの圧縮符号化方式でエンコードし、各種の記録媒体に記録する機能と、記録媒体の記録データをデコードし、動画像および音声を再生出力する機能とを具備する画像記録再生装置や撮像装置などに設けられる。この画像処理装置では、動画像および音声のエンコードとデコードとを並行して実行することが可能となっており、これらの処理が、OSを搭載しない、いわゆる組み込み型システムにより実行されるようになっている。

【0026】

この画像処理装置は、図1に示すように、エンコード/デコードの制御のためのCPU10、エンコーダ20、デコーダ30、処理中の画像データの一時保存のためにエンコーダ20およびデコーダ30により共用される画像メモリ40、および画像メモリ40の書き込み/読み出しを制御するメモリ制御部50を具備している。

【0027】

CPU10は、外部のホストコントローラと通信し、このホストコントローラからの情報に応じて、エンコーダ20、デコーダ30およびメモリ制御部50の動作を制御する。具体的には、例えば、ホストコントローラに指示された画像サイズや、確保すべきメモリ領域の数をメモリ制御部50に通知するとともに、この画像サイズでのエンコード/デコードの実行をエンコーダ20およびデコーダ30に指示する。また、エンコーダ20およびデコーダ30からの画像メモリ40に対する書き込み/読み出し要求をメモリ制御部50に通知し、画像データが書き込まれたメモリ領域を示す識別IDを、エンコーダ20およびデコーダ30とメモリ制御部50との間で転送する。10

【0028】

エンコーダ20は、入力された動画像および音声のデジタルデータに対して、MPEG方式などの所定の圧縮符号化方式に従って圧縮符号化処理を施し、動画像および音声の符号化ストリームを生成する。また、デコーダ30は、入力された動画像および音声の符号化ストリームに対して伸張復号化処理を施す。エンコーダ20およびデコーダ30は、動画像のエンコード、デコードをそれぞれ実行する際に、メモリ制御部50を通じて画像メモリ40にアクセスし、処理中の画像データを画像メモリ40に一時的に格納する。20

【0029】

画像メモリ40は、例えばSDRAMなどの半導体メモリからなり、その内部に、外部からのアクセスを受け付けるメモリI/F41を具備している。メモリI/F41は、画像メモリ40のハードウェアI/Fをメモリ制御部50に対して提供し、メモリ制御部50からアドレスおよび画像サイズの設定を受けて、エンコーダ20およびデコーダ30からメモリ制御部50を通じて供給された画像データを、画像メモリ40内の対応する領域に格納する。また、メモリ制御部50から指定された領域の画像データを読み出して、メモリ制御部50を通じてエンコーダ20およびデコーダ30に出力する。なお、メモリI/F41は、後述するように、ルックアップテーブルおよびデータフォーマットを用いて、画像メモリ40へのデータの書き込み/読み出しを制御する。30

【0030】

メモリ制御部50は、エンコーダ20およびデコーダ30からの画像メモリ40に対するアクセスを一元的に管理する機能ブロックである。メモリ制御部50は、エンコードやデコードが実行される際に、画像サイズや確保すべきメモリ領域の数の情報をCPU10から受け取り、これらの情報に基づいて画像メモリ40内のメモリ領域を確保する。そして、エンコーダ20およびデコーダ30から供給された画像メモリ40に格納し、その格納領域を示す識別IDを、CPU10を通じてエンコーダ20およびデコーダ30に通知する。また、エンコーダ20およびデコーダ30からCPU10を通じて識別IDが指定されると、識別IDに応じた画像メモリ40上のメモリ領域から画像データを読み出し、エンコーダ20およびデコーダ30に出力する。40

【0031】

このような画像処理装置において、動画像のエンコードが行われる場合は、CPU10は、ホストコントローラからの情報に基づいて、エンコーダ20に対してエンコードの開始を命令するとともに、画像サイズや、確保すべきメモリ領域に対応する画像の枚数をメモリ制御部50に指定する。メモリ制御部50は、CPU10からの情報に応じて、エンコードに必要なメモリ領域を画像メモリ40に確保する。

【0032】

エンコーダ20は、動画像データの入力を受けると、処理中の画像データをメモリ制御部50を通じて画像メモリ40に一時的に格納し、また適宜読み出しながらエンコードを行う。画像データを格納する際には、エンコーダ20からの要求がCPU10を通じてメ50

モリ制御部 50 に伝達され、メモリ制御部 50 は、確保したメモリ領域のうちの 1 つにエンコーダ 20 からの画像データを書き込み、当該メモリ領域を示す識別 ID を、CPU 10 を通じてエンコーダ 20 に通知する。

【0033】

その後、画像メモリ 40 から画像データを読み出す際には、エンコーダ 20 は、読み出す画像データに対応する識別 ID を、CPU 10 を通じてメモリ制御部 50 に通知する。メモリ制御部 50 は、識別 ID に対応する画像メモリ 40 内のメモリ領域から画像データを読み出し、エンコーダ 20 に出力する。そして、ホストコントローラからエンコードの終了が要求されると、メモリ制御部 50 は、エンコードのために確保していたメモリ領域を解放し、他のデータの書き込みが可能な状態に設定する。

10

【0034】

動画像のデコードが行われる場合も、上記と同様の動作が行われる。すなわち、デコードが開始されると、メモリ制御部 50 により必要なメモリ領域が確保される。そして、デコーダ 30 からの画像データがメモリ制御部 50 の制御により 1 つのメモリ領域に格納され、そのメモリ領域の識別 ID がデコーダ 30 に通知される。画像データの読み出しの際は、デコーダ 30 から識別 ID がメモリ制御部 50 に通知され、メモリ制御部 50 は、その識別 ID に対応するメモリ領域から画像データを読み出し、デコーダ 30 に出力する。

【0035】

このように、上記の画像処理装置では、エンコーダ 20 およびデコーダ 30 が、画像メモリ 40 に対してメモリ制御部 50 を介して間接的にアクセスする構成となっている。エンコーダ 20 およびデコーダ 30 は、画像メモリ 40 の物理アドレスの代わりに、メモリ制御部 50 により確保されるメモリ領域の識別 ID を用いて、画像データの書き込み / 読み出しを要求することができる。

20

【0036】

一方、メモリ制御部 50 は、エンコーダ 20 およびデコーダ 30 の一方の処理が開始されるたびに、その処理のために必要な画像メモリ 40 内のメモリ領域をその都度確保する。このとき、メモリ制御部 50 は、画像メモリ 40 内の単位メモリ領域をノードとした二分木構造を用いて、画像メモリ 40 内の空き領域にメモリ領域を効率的に割り当て、また、画像メモリ 40 のハードウェア I/F に合わせて、割り当てたメモリ領域をメモリ I/F 41 に設定する。

30

【0037】

このため、エンコード / デコードの少なくとも一方の処理対象となる画像サイズがホストコントローラからの指示により変更された場合にも、変更された処理のために確保していたメモリ領域を一旦解放して空き領域に設定し、その空き領域を変更後の画像サイズに対応するメモリ領域として容易に再利用することが可能となる。これとともに、画像サイズに応じたメモリ領域の確保がメモリ制御部 50 に応じて一元的に制御され、エンコーダ 20 およびデコーダ 30 は、画像メモリ 40 の仕様や物理アドレスを常に認識することなく、処理中の画像データの受け渡しを行うことが可能となる。

【0038】

ここで、エンコード / デコードの処理対象となる画像サイズが変更される場合とは、エンコーダ 20 およびデコーダ 30 による処理中に画像メモリ 40 に一時的に格納される画像データの画像サイズに変更が生じる場合を意味する。例えば、エンコード時には、入力された動画像データに対してエンコーダ 20 が画像サイズを変化させた後、エンコードを実行する、あるいは、画像サイズの変更された動画像データの入力を受けてエンコードを実行することにより、生成される符号化ストリームの画像サイズが変更される場合などが該当する。また、デコード時には、デコーダ 30 に入力される符号化ストリームの画像サイズが変更される場合などが該当する。

40

【0039】

次に、図 2 は、メモリ制御部 50 の内部構成を示すブロック図である。

図 2 に示すように、メモリ制御部 50 は、メモリ領域管理部 51 、メモリ確保処理部 5

50

2、メモリ解放処理部53、ルックアップテーブル設定部54、データフォーマット設定部55、CPU・I/F56、エンコーダI/F57およびデコーダI/F58を具備している。なお、図2では、画像メモリ40のメモリI/F41の具備するルックアップテーブル41aおよびデータフォーマット41bも示している。

【0040】

メモリ領域管理部51は、CPU・I/F56を通じてCPU10と通信し、メモリ制御部50内の全体の動作を制御する。具体的には、CPU10から指定された画像サイズおよび枚数に対応するメモリ領域の確保を、メモリ確保処理部52、ルックアップテーブル設定部54およびデータフォーマット設定部55に要求し、確保されたメモリ領域を識別IDを用いてエンコーダ20およびデコーダ30に提供して、その使用状態を管理する。また、エンコーダ20およびデコーダ30と画像メモリ40との間のデータ転送を制御する。そして、エンコード/デコードの処理が終了したときに、メモリ解放処理部53、ルックアップテーブル設定部54およびデータフォーマット設定部55を制御して、確保していたメモリ領域を解放させる。

【0041】

メモリ確保処理部52は、メモリ領域管理部51から指定された画像サイズに対応するメモリ領域を画像メモリ40内の空き領域に割り当て、その領域のアドレスを算出する。具体的には、メモリ確保処理部52は、画像メモリ40内の単位メモリ領域をノードとした二分木構造を用いて、各単位メモリ領域の割り当て状態を管理する。そして、メモリ領域管理部51に指定された画像サイズのデータを格納可能なメモリ領域を空き領域から抽出し、その領域の先頭アドレスをメモリ領域管理部51に対して出力する。メモリ解放処理部53は、メモリ領域管理部51から指定されたメモリ領域を解放する処理を行う。

【0042】

ルックアップテーブル設定部54およびデータフォーマット設定部55は、確保されたメモリ領域が使用可能となるように、画像メモリ40のアクセス仕様に合わせて情報の設定を行う機能ブロックである。本実施の形態では、メモリ領域の先頭アドレスと、そのメモリ領域のデータサイズを1次元または2次元のデータとして表した画像サイズとを指定することで、画像メモリ40の書き込み/読み出しを行うことが可能となっている。そのため、メモリI/F41は、メモリ領域の大きさに対応する画像サイズの設定を保持するルックアップテーブル41aと、メモリ領域の先頭アドレスとこれに対応するルックアップテーブル41a内の要素とが対応付けられたデータフォーマット41bを具備している。

【0043】

ルックアップテーブル設定部54は、メモリ領域を確保する際に、メモリ領域管理部51から指定された画像サイズを、メモリI/F41のルックアップテーブル41aに対して設定する。

【0044】

データフォーマット設定部55は、メモリ確保処理部52により算出されたアドレスを、メモリI/F41のデータフォーマット41bに対して設定するとともに、そのアドレスをルックアップテーブル41aに対応付ける。そして、画像メモリ40に対する画像データの書き込み/読み出し時に、メモリ領域管理部51から指定される情報に応じて、データフォーマット41b内の対応する要素を指定し、書き込み/読み出しをメモリI/F41に対して要求する。

【0045】

CPU・I/F56は、CPU10とメモリ領域管理部51との間で情報を受け渡すためのI/F回路である。また、エンコーダI/F57およびデコーダI/F58は、それぞれエンコーダ20およびデコーダ30とメモリ領域管理部51との間で画像データを受け渡すためのI/F回路である。

【0046】

[メモリ領域管理部の動作]

10

20

30

40

50

次に、メモリ制御部 50 内の各部の動作について詳述する。まず、メモリ領域管理部 51 によるメモリ領域の管理方法について説明する。

【0047】

図 3 は、確保したメモリ領域のメモリ領域管理部 51 による管理方法を説明するための図である。

メモリ領域管理部 51 は、図 3 に示すように、画像サイズ要素および画像メモリ要素という 2 つの要素がポインタにより接続された線形リストを用いて、メモリ領域を管理する。

【0048】

画像サイズ要素は、1 つのエンコーダ 20 またはデコーダ 30 による処理の実行に対応して生成される要素であり、後に生成された他の画像サイズ要素をポインタにより指示することで構成されるリスト構造を有している。1 つの画像サイズ要素は、エンコードまたはデコードの処理の開始時に、CPU10 に指定された画像サイズを設定した画像メモリ要素を、このとき CPU10 に指定された枚数分だけ具備する。すなわち、各画像メモリ要素は、エンコーダ 20 またはデコーダ 30 が画像メモリ 40 に格納する 1 ピクチャ分の画像データのために確保されるメモリ領域に対応している。

【0049】

また、1 つの画像サイズ要素において、画像データが格納済みである画像メモリ要素と未格納である画像メモリ要素とが、ポインタを用いたリスト構造を持つアクティベートリスト (Activate List) およびネゲートリスト (Negate List) にそれぞれ接続される。そして、エンコーダ 20 から画像メモリ 40 に対する 1 ピクチャ分の画像データの書き込みが要求されると、ネゲートリストの先頭の画像メモリ要素が、アクティベートリストの最後尾に接続され、画像メモリ 40 に書き込んだ 1 ピクチャ分の画像データが不要となると、当該画像メモリ要素がネゲートリストの最後尾に接続される。

【0050】

また、画像サイズ要素は、アクティベートリストの先頭および最後尾の画像メモリ要素をそれぞれ示すアクティベートトップ (Activate_top) およびアクティベートテール (Activate_tail) と、ネゲートリストの先頭および最後尾の画像メモリ要素をそれぞれ示すネゲートトップ (Negate_top) およびネゲートテール (Negate_tail) とを常に認識することで、各リストを管理している。

【0051】

図 3 の例では、2 つの画像サイズ要素 101a および 101b が生成されている。本実施の形態では、エンコーダ 20 およびデコーダ 30 がそれぞれ 1 つずつ設けられていることから、画像サイズ要素は最大 2 つ生成される。

【0052】

例えば、エンコードのために画像サイズ要素 101a が生成されたものとすると、エンコーダ 20 による処理のために 5 つの画像メモリ要素 102a ~ 102c, 103a および 103b に対応するメモリ領域が確保され、それらのうち、アクティベートリスト 102 に接続された画像メモリ要素 102a ~ 102c に対応するメモリ領域にはすでにエンコーダ 20 からの画像データが格納され、ネゲートリスト 103 に接続された画像メモリ要素 103a および 103b に対応するメモリ領域は使用されていない状態となっている。

【0053】

また、同様に、デコードのために画像サイズ要素 101b が生成されたものとすると、デコーダ 30 による処理のために 4 つの画像メモリ要素 105a ~ 105d に対応するメモリ領域が確保されており、アクティベートリスト 104 には要素が存在しておらず、ネゲートリスト 105 に接続されたすべての画像メモリ要素 105a ~ 105d に対応するメモリ領域が未使用状態となっている。

【0054】

この図 3 の状態から、例えばデコーダ 30 によるデコードが終了されると、画像サイズ

10

20

30

40

50

要素 101b とこれが具備するすべての画像メモリ要素 105a ~ 105d が消去される。これにより、デコーダ 30 の処理のために確保されていたメモリ領域が解放される。また、図 3 の状態から、例えばデコーダ 30 の処理対象の画像サイズをデコードの途中で変更する場合には、一旦画像サイズ要素 101b を消去した後、新たな画像サイズが設定された画像メモリ要素を具備する画像サイズ要素を生成する。このような設定変更の間、エンコーダ 20 は、画像サイズ要素 101a の具備する画像メモリ要素に対応するメモリ領域を用いてエンコードを続行することができる。

【0055】

図 4 は、画像サイズ要素および画像メモリ要素のデータ構造を示す図である。

各画像サイズ要素には、メモリ領域管理部 51 により図 4 (A) に示すようなデータが設定される。識別 ID 111 は、各画像サイズ要素を識別するユニークな番号である。画像幅 112 および画像高さ 113 は、当該画像サイズ要素が具備する画像メモリ要素に設定される画像サイズを示し、画像枚数 114 は、その画像メモリ要素の数を示す。

【0056】

また、当該画像サイズの具備する画像メモリ要素を管理するために、当該画像サイズ要素内のアクティベートトップ、アクティベートテール、ネゲートトップおよびネゲートテールとなっている画像メモリ要素を指し示すポインタ 115 ~ 118 が設定される。これらのポインタ 115 ~ 118 としては、例えば後述する画像メモリ要素の識別 ID などが用いられる。さらに、当該画像サイズ要素より後に生成された他の画像サイズ要素へのポインタ 119 が、例えばその画像サイズ要素の識別 ID により設定される。

【0057】

一方、各画像メモリ要素には、メモリ領域管理部 51 により図 4 (B) に示すようなデータが設定される。識別 ID 121 は、各画像メモリ要素を識別するユニークな番号であり、対応するメモリ領域に画像データを格納した際にエンコーダ 20 またはデコーダ 30 に提供されるものである。ノード番号 122 は、メモリ確保処理部 52 により確保されたメモリ領域に対応する番号である。

【0058】

また、画像メモリ 40 上に確保される実際のメモリ領域と対応付けるために、ルックアップテーブル設定部 54 が保持するルックアップテーブル要素へのポインタ 123 と、データフォーマット設定部 55 が保持するデータフォーマット要素へのポインタ 124 が設定される。これらのポインタ 123 および 124 としては、例えばその要素の識別 ID が用いられる。なお、ルックアップテーブル要素およびデータフォーマット要素については後述する。

【0059】

さらに、同じアクティベートリストまたはネゲートリストにおいて次に接続された画像メモリ要素を指し示すポインタ 125 が、例えばその画像メモリ要素の識別 ID により設定される。

【0060】

次に、メモリ領域管理部 51 の処理について説明する。まず、図 5 は、メモリ領域を確保する際のメモリ領域管理部 51 の処理の流れを示すフローチャートである。

[ステップ S101] ホストコントローラよりエンコードまたはデコードの開始が CPU 10 に対して要求されたとき、CPU 10 は、ホストコントローラから指定された確保すべき画像サイズ、および必要なメモリ領域に対応する画像の枚数を、CPU・I/F 56 を介してメモリ領域管理部 51 に通知する。メモリ領域管理部 51 は、CPU 10 からの画像サイズおよび枚数を受け取り、以下の処理の実行を開始する。

【0061】

[ステップ S102] 画像サイズ要素のリストに、新たな画像サイズ要素を加える。このとき、新たな画像サイズ要素の識別 ID 111 の欄に値を設定し、また、すでに生成されていた画像サイズ要素がある場合には、その要素のポインタ 119 の欄に、新たな画像サイズ要素の識別 ID を格納する。

10

20

30

40

50

【0062】

[ステップS103] 新たな画像サイズ要素について、CPU10からの画像サイズを画像幅112および画像高さ113の欄に格納し、枚数を画像枚数114の欄に格納する。なお、画像サイズが1次元のデータとして表されている場合は、そのデータは例えば画像幅112の欄にのみ格納されればよい。

【0063】

[ステップS104] 新たに生成した画像サイズ要素のネゲートリストの最後尾に、画像メモリ要素を1つ加える。このとき、画像サイズ要素のネゲートテールを示すポインタ118を更新する。また、すでにネゲートリストに画像サイズ要素が存在する場合には、ネゲートトップを示すポインタ117を更新するとともに、最後尾の画像サイズ要素に設定されたポインタ119に、新たな画像サイズ要素の識別IDを設定する。

【0064】

[ステップS105] ステップS101で受け取った画像サイズをメモリ確保処理部52に渡し、その画像サイズに対応するメモリ領域の確保を要求する。これにより、メモリ確保処理部52は、図21で説明するように、受け取った画像サイズに対応するメモリ領域を画像メモリ40内に確保し、そのメモリ領域を示す二分木構造のノード番号と、メモリ領域の先頭アドレスとをメモリ領域管理部51に返す。

【0065】

[ステップS106] メモリ確保処理部52から先頭アドレスおよびノード番号を受け取る。

[ステップS107] 当該画像メモリ要素のノード番号122の欄に、メモリ確保処理部52から受け取ったノード番号を格納する。

【0066】

[ステップS108] ルックアップテーブル設定部54に対して、CPU10から指定された画像サイズを渡す。これにより、ルックアップテーブル設定部54は、図12で説明するように、画像サイズをルックアップテーブル41aに設定し、また、対応するルックアップテーブル要素を更新して、その要素の識別IDをメモリ領域管理部51に返す。

【0067】

[ステップS109] ルックアップテーブル設定部54からの識別IDを受け取り、ルックアップテーブル要素へのポインタ123としてその識別IDを設定する。

[ステップS110] メモリ確保処理部52から受け取った先頭アドレスおよびノード番号、ルックアップテーブル設定部54から受け取ったルックアップテーブル要素の識別ID、画像サイズに基づくメモリマッピングタイプを、データフォーマット設定部55に渡す。これにより、データフォーマット設定部55は、図17で説明するように、使用中であることを示すリストに1つのデータフォーマット要素を加え、その要素に対応付けたデータフォーマット41bの要素にデータを設定する。そして、当該データフォーマット要素の識別IDを、メモリ領域管理部51に返す。

【0068】

[ステップS111] データフォーマット設定部55からの識別IDを受け取り、データフォーマット要素へのポインタ124としてその識別IDを設定する。以上のステップS104～S111の処理により、1つの画像メモリ要素に対応するメモリ領域が確保される。

【0069】

[ステップS112] 必要なメモリ領域に対応する画像の枚数分だけメモリ領域が確保されたか否かを判別し、確保された場合は処理を終了する。また、確保されていない場合は、ステップS104に進み、新たな画像メモリ要素をネゲートリストの最後尾に加えて、この要素に対応するメモリ領域を確保する処理を行う。

【0070】

以上の処理により、CPU10から指定された枚数分だけの画像メモリ要素がネゲートリストに生成される。すなわち、エンコードまたはデコードに必要な画像枚数分の未使用

10

20

30

40

50

状態（使用可能状態）のメモリ領域が確保される。

【0071】

次に、図6は、確保されたメモリ領域をエンコーダ20またはデコーダ30に提供する際のメモリ領域管理部51の処理の流れを示すフローチャートである。

〔ステップS201〕エンコーダ20およびデコーダ30の処理が開始され、画像データを一時的に格納するためのメモリ領域が必要になると、エンコーダ20またはデコーダ30は、CPU10を通じてメモリ制御部50に対してメモリ領域の提供を要求する。メモリ領域管理部51は、CPU10から上記の要求を受け、以下の処理の実行を開始する。

【0072】

〔ステップS202〕画像サイズ要素のリストを先頭から順に参照し、参照した画像サイズ要素が、提供すべきもの、すなわちメモリ領域を要求したエンコーダ20またはデコーダ30に対応するものであるか否かを判定する。提供すべき要素でない場合はステップS203に進み、提供すべき要素である場合はステップS204に進む。

【0073】

〔ステップS203〕リスト中の次の画像サイズ要素を参照し、当該要素についてステップS202の判定を行う。

〔ステップS204〕参照している画像サイズ要素に設定されているネゲートトップへのポインタ117に基づき、ネゲートリストの先頭の画像メモリ要素を取り出す。具体的には、ネゲートトップへのポインタ117を、ネゲートリスト上の次の画像メモリ要素を指し示すように更新する。

【0074】

〔ステップS205〕取り出した画像メモリ要素の識別IDを、メモリ領域の要求元のエンコーダ20またはデコーダ30に対して、CPU10を通じて提供する。

〔ステップS206〕取り出した画像メモリ要素を、アクティベートリストの最後尾に接続する。具体的には、アクティベートテールへのポインタ116を、新たな画像メモリ要素を指し示すように更新する。また、すでにアクティベートリストに画像メモリ要素が接続されていた場合は、そのうち最後尾に接続されていた要素について、次の画像メモリ要素へのポインタ125を、新たな画像メモリ要素を指し示すように設定する。また、必要に応じてアクティベートトップへのポインタ115を更新する。

【0075】

〔ステップS207〕メモリ領域の要求元のエンコーダ20またはデコーダ30から画像データを受け取る。そして、当該画像メモリ要素に設定されたルックアップテーブル要素およびデータフォーマット要素へのポインタ123および124に基づいて各要素を指定し、ルックアップテーブル設定部54およびデータフォーマット設定部55に対して画像データの書き込みを指示するとともに、受け取った画像データを画像メモリ40に転送する。これにより、アクティベートリストに接続した新たな画像メモリ要素に対応するメモリ領域に、画像データが書き込まれる。

【0076】

以上の処理により、エンコーダ20またはデコーダ30からの1ピクチャ分の画像データが、画像メモリ40に書き込まれる。このとき、画像データが書き込まれたメモリ領域に対応する画像メモリ要素が、アクティベートリストに接続され、使用状態とされる。

【0077】

また、当該画像メモリ要素の識別IDを受け取ったエンコーダ20またはデコーダ30は、この後、その識別IDをCPU10を通じてメモリ制御部50に通知することで、必要な画像データを画像メモリ40から読み出すことが可能となる。このとき、メモリ領域管理部51は、CPU10を通じて識別IDを受け取ると、該当する画像メモリ要素に設定されたポインタに基づいて、対応するルックアップテーブル要素およびデータフォーマット要素をそれぞれルックアップテーブル設定部54およびデータフォーマット設定部55に指定し、画像データの読み出しを要求する。これにより、該当する画像データが画像

10

20

30

40

50

メモリ 40 から読み出されてメモリ領域管理部 51 に供給され、さらにエンコーダ I / F 57 またはデコーダ I / F 58 を通じて、エンコーダ 20 またはデコーダ 30 に転送される。

【0078】

次に、図 7 は、エンコーダ 20 またはデコーダ 30 からメモリ領域が返却される際のメモリ領域管理部 51 の処理の流れを示すフロー チャートである。

[ステップ S 301] エンコーダ 20 またはデコーダ 30 は、所定のピクチャのデータの画像メモリ 40 からの読み出しを、処理に必要な回数だけ行った後、対応するメモリ領域を返却するための情報を、CPU 10 を通じてメモリ制御部 50 に通知する。このとき、返却するメモリ領域を示す識別 ID が、メモリ制御部 50 に通知される。メモリ領域管理部 51 は、CPU 10 からの返却通知情報を読み取る。
10

【0079】

[ステップ S 302] 画像サイズ要素のリストを先頭から順に参照し、参照した画像サイズ要素が、返却が要求された画像サイズ要素を含むものか否かを判定する。含まない場合はステップ S 303 に進み、含む場合はステップ S 304 に進む。

【0080】

[ステップ S 303] リスト中の次の画像サイズ要素を参照し、当該要素についてステップ S 302 の判定を行う。

[ステップ S 304] 参照している画像サイズ要素のアクティベートリストを先頭から順に参照し、参照した画像メモリ要素が、返却が要求されたものであるか否かを判定する。該当しない場合はステップ S 305 に進み、該当する場合はステップ S 306 に進む。
20

【0081】

[ステップ S 305] アクティベートリスト中の次の画像メモリ要素を参照し、当該要素についてステップ S 304 の判定を行う。

[ステップ S 306] 参照している画像メモリ要素を、アクティベートリストから取り出す。ここで、メモリ領域の返却の際には、アクティベート内のどの位置からでも画像メモリ要素を取り出すことができる。従ってこのとき、取り出された画像メモリ要素の直前の要素について、次の画像メモリ要素へのポインタ 125 を更新する。また、必要に応じて、当該画像サイズ要素に設定された、アクティベートトップへのポインタ 115 またはアクティベートテールへのポインタ 116 を更新する。
30

【0082】

[ステップ S 307] 取り出した画像メモリ要素を、ネゲートリストの最後尾に接続する。具体的には、ネゲートテールへのポインタ 118 を、新たに接続した画像メモリ要素を指し示すように更新する。また、すでにネゲートリストに画像メモリ要素が接続されていた場合は、そのうち最後尾に接続されていた要素について、次の画像メモリ要素へのポインタ 125 を、新たな画像メモリ要素を指し示すように設定する。また、必要に応じてネゲートトップへのポインタ 117 を更新する。

【0083】

以上の処理により、読み出しの必要がなくなったメモリ領域に対応する画像メモリ要素がネゲートリストに接続され、新たな画像データを格納可能な状態とされる。ここで、解放するメモリ領域はアクティベートリスト中のどの位置からも選択可能で、選択されたメモリ領域はネゲートリストの最後尾に接続される。このような線形リストの管理を行うことで、例えば MPEG 方式のように、画像メモリに対して画像データを書き込む順番と、読み出す順番とが一致しない場合にも容易に対応することが可能となる。
40

【0084】

次に、図 8 は、メモリ領域を解放する際のメモリ領域管理部 51 の処理の流れを示すフロー チャートである。

[ステップ S 401] ホストコントローラからエンコーダ 20 またはデコーダ 30 に対して、エンコードまたはデコードの処理終了、または処理対象の画像サイズの変更が指示されると、メモリ制御部 50 には、当該処理で使用していたメモリ領域の解放を指示する
50

情報が、C P U 1 0 を通じて供給される。このとき、当該処理で設定されていた画像サイズおよび枚数がメモリ制御部50に供給される。メモリ領域管理部51は、C P U 1 0 からの画像サイズおよび枚数を受け取ると、以下の処理の実行を開始する。

【0085】

[ステップS402] 画像サイズ要素のリストを先頭から順に参照し、参照した画像サイズ要素が、解放をすべきもの、すなわちメモリ領域の解放を要求したエンコーダ20またはデコーダ30に対応するものであるか否かを判定する。解放すべき要素でない場合はステップS403に進み、解放すべき要素である場合はステップS404に進む。

【0086】

[ステップS403] リスト中の次の画像サイズ要素を参照し、当該要素についてステップS402の判定を行う。

[ステップS404] 参照している画像サイズ要素のネゲートリストから、先頭の画像メモリ要素を取り出す。具体的には、当該画像サイズ要素に設定されたネゲートトップへのポインタ117を、次の画像メモリ要素を指し示すように更新する。

【0087】

[ステップS405] 取り出した画像メモリ要素に設定されたノード番号を、メモリ解放処理部53に渡し、対応するメモリ領域の解放を要求する。これにより、メモリ解放処理部53は、図22で説明するように、受け取ったノード番号に対応する画像メモリ40内の領域を解放する処理を実行する。

【0088】

[ステップS406] 取り出した画像メモリ要素に設定されたポインタ124により指示されたデータフォーマット要素を解放するように、データフォーマット設定部55に要求する。これにより、データフォーマット設定部55は、図18で説明するように、指定されたデータフォーマット要素を解放する処理を実行し、さらに、対応するルックアップテーブル要素を解放するようにメモリ領域管理部51に要求する

[ステップS407] データフォーマット設定部55からの要求を受けた後、ポインタ123により指示されたルックアップテーブル要素を解放するように、ルックアップテーブル設定部54に要求する。これにより、ルックアップテーブル設定部54は、図13で説明するように、指定されたルックアップテーブル要素を解放する処理を実行する。

【0089】

以上のステップS404～S407の処理により、ネゲートリスト内の1つの画像メモリ要素に対応するメモリ領域が解放される。

[ステップS408] 当該画像サイズ要素のネゲートリストに、画像メモリ要素が存在するか否かを判定する。存在する場合はステップS404に戻り、ネゲートリストの先頭の画像メモリ要素について、解放処理を行う。そして、すべての画像メモリ要素が解放され、ネゲートリストに要素が存在しなくなったとき、ステップS409に進む。

【0090】

[ステップS409] 当該画像サイズ要素を解放する。

以上の処理により、エンコーダ20またはデコーダ30の処理のために確保されていたメモリ領域が解放され、新たに使用可能な状態とされる。

【0091】

[ルックアップテーブル設定部の動作]

次に、ルックアップテーブル設定部54の動作について説明する。まず、図9は、ルックアップテーブル41aの構成例を示す図である。

【0092】

図9に示すように、ルックアップテーブル41aは、画像メモリ40のハードウェアI/Fにマッピングされた領域に対応する32個の要素を持つ。各要素には、一次元または二次元で表される画像サイズのデータが格納される。例えば、一次元で表される場合、画像サイズは18ビットのデータとして格納され、二次元で表される場合、垂直方向、水平方向の画像サイズがそれぞれ9ビットのデータとして格納される。

10

20

30

40

50

【0093】

図10は、ルックアップテーブル設定部54によるメモリ領域の管理方法を説明するための図である。

ルックアップテーブル設定部54は、メモリ領域のデータサイズの情報を、ルックアップテーブル要素を用いて管理する。ルックアップテーブル要素は、図10に示すように、アクティベートリスト131およびネゲートリスト132の2つの線形リストにより管理される。アクティベートリスト131には、メモリ領域として現在使用されている、すなわち、確保されたメモリ領域に割り当てられているルックアップテーブル要素が接続され、ネゲートリスト132には、未使用のルックアップテーブル要素が接続される。

【0094】

また、ルックアップテーブル要素は、ルックアップテーブル41aの要素数だけあらかじめ用意され、初期状態では、すべてのルックアップテーブル要素がネゲートリスト132に接続される。そして、エンコーダ20またはデコーダ30による処理の開始時に、必要なメモリ領域の確保が要求されると、ネゲートリスト132の先頭のルックアップテーブル要素が取り出され、アクティベートリスト131の最後尾に接続される。そして、エンコーダ20またはデコーダ30の処理が終了して、確保されていたメモリ領域の解放が要求されると、対応するルックアップテーブル要素がアクティベートリスト131から取り出され、ネゲートリスト132の最後尾に接続される。

【0095】

図11は、ルックアップテーブル要素のデータ構造を示す図である。

各ルックアップテーブル要素には、ルックアップテーブル設定部54により図11に示すようなデータが設定される。識別ID141は、各ルックアップテーブル要素を識別するユニークな番号である。使用要素数142は、当該ルックアップテーブル要素を使用する画像メモリ要素のカウント値を示す。垂直方向サイズ143および水平方向サイズ144は、2次元のデータで表された場合の画像サイズを示し、一次元サイズ145は、1次元のデータで表された場合の画像サイズを示す。

【0096】

また、ルックアップテーブル41aとの対応付けを行うために、ルックアップテーブル41a内の対応する要素を指し示すポインタ146が設定される。さらに、線形リストを構成するために、同一リストにおいて次に接続されるルックアップテーブル要素を指し示すポインタ147も設定される。

【0097】

なお、上記の各データのうち、識別ID141およびルックアップテーブル41a内の要素へのポインタ146については、あらかじめ設定される。また、初期状態では、使用要素数142、垂直方向サイズ143、水平方向サイズ144および一次元サイズ145として、0が設定される。

【0098】

また、ルックアップテーブル設定部54は、この他に、アクティベートリスト131の先頭（アクティベートトップ）および最後尾（アクティベートテール）のルックアップテーブル要素をそれぞれ指し示すポインタと、ネゲートリスト132の先頭（ネゲートトップ）および最後尾（ネゲートテール）のルックアップテーブル要素をそれぞれ示すポインタとを図示しないテーブルに保持することで、各リストを管理している。

【0099】

次に、ルックアップテーブル設定部54の処理について説明する。まず、図12は、メモリ領域が確保される際のルックアップテーブル設定部54の処理の流れを示すフローチャートである。

【0100】

[ステップS501] エンコードまたはデコードのためのメモリ領域を確保する処理の実行中に、メモリ領域管理部51から、確保すべきメモリ領域の画像サイズがルックアップテーブル設定部54に渡される（図5のステップS108に対応）。ルックアップテー

10

20

30

40

50

ブル設定部 5 4 は、この画像サイズを受け取って、以下の処理を開始する。

【 0 1 0 1 】

[ステップ S 5 0 2] アクティベートリスト 1 3 1 の先頭（アクティベートトップ）のルックアップテーブル要素を参照する。なお、図示しないが、アクティベートリスト 1 3 1 にルックアップテーブル要素が接続されていない場合は、ステップ S 5 0 6 に進む。

【 0 1 0 2 】

[ステップ S 5 0 3] 参照しているルックアップテーブル要素が、メモリ領域管理部 5 1 からの画像サイズに対応するものであるか否かを判定し、対応する場合はステップ S 5 1 0 に進み、対応しない場合はステップ S 5 0 4 に進む。

【 0 1 0 3 】

[ステップ S 5 0 4] アクティベートリスト 1 3 1 に次のルックアップテーブル要素がある場合はステップ S 5 0 5 に進み、ない場合はステップ S 5 0 6 に進む。

[ステップ S 5 0 5] アクティベートリスト 1 3 1 上の次のルックアップテーブル要素を参照して、ステップ S 5 0 3 に戻る。以上のステップ S 5 0 3 ~ S 5 0 5 の処理により、アクティベートリスト 1 3 1 の最後尾のルックアップテーブル要素が、メモリ領域管理部 5 1 からの画像サイズに対応するものである場合にステップ S 5 1 0 に進み、対応するルックアップテーブル要素がアクティベートリスト 1 3 1 に存在しない場合にステップ S 5 0 6 に進む。

【 0 1 0 4 】

[ステップ S 5 0 6] ネゲートリスト 1 3 2 の先頭（ネゲートトップ）のルックアップテーブル要素を取り出し、ネゲートトップの要素を指し示すポインタを、次の要素を指し示すように更新する。

【 0 1 0 5 】

[ステップ S 5 0 7] 取り出したルックアップテーブル要素を、アクティベートリスト 1 3 1 の最後尾に接続する。具体的には、それまでアクティベートリスト 1 3 1 の最後尾に接続されていたルックアップテーブル要素に設定されていた、次の要素へのポインタ 1 4 7 と、アクティベートテールの要素を指し示すポインタとを、新たに接続した要素を指し示すように更新する。

【 0 1 0 6 】

[ステップ S 5 0 8] 当該ルックアップテーブル要素の垂直方向サイズ 1 4 3 および水平方向サイズ 1 4 4 の各欄、または一次元サイズ 1 4 5 の欄に、メモリ領域管理部 5 1 からの画像サイズを設定する。

【 0 1 0 7 】

[ステップ S 5 0 9] 当該ルックアップテーブル要素に設定されたポインタ 1 4 7 が指し示すルックアップテーブル 4 1 a 上の要素に、ステップ S 5 0 8 で設定した画像サイズを設定する。

【 0 1 0 8 】

[ステップ S 5 1 0] 当該ルックアップテーブル要素の使用要素数 1 4 2 のカウント値に 1 を加算する。

[ステップ S 5 1 1] 当該ルックアップテーブル要素の識別 ID 1 4 1 を、メモリ領域管理部 5 1 に渡す。メモリ領域管理部 5 1 は、ルックアップテーブル設定部 5 4 からの識別 ID を受け取ることで、画像メモリ要素とルックアップテーブル要素とを対応付ける（図 5 のステップ S 1 0 9 に対応）。

【 0 1 0 9 】

以上の処理により、エンコーダ 2 0 またはデコーダ 3 0 の処理の開始時に、確保されたメモリ領域に対応するルックアップテーブル要素が、アクティベートリスト 1 3 1 の最後尾に生成される。ここで、本実施の形態では、使用要素数 1 4 2 のカウント値を用いることで、1 つのルックアップテーブル要素に対して、同じ画像サイズが設定された複数のメモリ領域（画像メモリ要素）を対応付けている。これにより、3 2 個という限られた要素数を有するルックアップテーブル 4 1 a に対して、画像サイズを効率よく割り当てるこ

10

20

30

40

50

が可能となっている。

【0110】

次に、図13は、メモリ領域が解放される場合のルックアップテーブル設定部54の処理の流れを示すフローチャートである。

[ステップS601] エンコーダ20またはデコーダ30の処理が一旦終了する場合には、メモリ領域管理部51は、1つの画像メモリ要素を解放する処理中に、その画像メモリ要素に対応するルックアップテーブル要素を解放するように、ルックアップテーブル設定部54に対して要求する(図8のステップS407に対応)。この要求では、例えば、解放するメモリ領域の画像サイズが指定される。ルックアップテーブル設定部54は、メモリ領域管理部51からの要求を受けると、以下の処理を開始する。10

【0111】

[ステップS602] アクティベートリスト131の先頭(アクティベートトップ)のルックアップテーブル要素を参照する。

[ステップS603] 参照したルックアップテーブル要素が、メモリ領域管理部51からの要求に対応する要素であるか否かを、例えば画像サイズの設定から判定し、対応しない場合はステップS604に進み、対応する場合はステップS605に進む。

【0112】

[ステップS604] アクティベートリスト131上の次のルックアップテーブル要素を参照し、ステップS603に進む。

[ステップS605] 当該ルックアップテーブル要素に設定された使用要素数142のカウント値を、1だけ減算する。20

【0113】

[ステップS606] 使用要素数142のカウント値が0である場合はステップS607に進み、0でない場合は処理を終了する。

[ステップS607] 当該ルックアップテーブル要素を、アクティベートリスト131から取り出す。このとき、必要に応じて、アクティベートトップを指し示すポインタ、アクティベートテールを指し示すポインタ、取り出した要素の直前要素に設定された、次の要素へのポインタ147などを更新する。

【0114】

[ステップS608] 取り出したルックアップテーブル要素を、ネゲートリスト132の最後尾に接続し、それまでネゲートリスト132の最後尾に接続されていた要素の、次の要素を指し示すポインタ147と、ネゲートテールを指し示すポインタとを、新たに接続した要素を指し示すように更新する。30

【0115】

以上の処理が、エンコーダ20またはデコーダ30の処理のために確保されていたメモリ領域(画像メモリ要素)を1つずつ解放するたびに実行され、対応するルックアップテーブル要素の使用要素数142のカウント値が減算されていく。そして、カウント値が0となると、1つのルックアップテーブル要素が、アクティベートリスト131からネゲートリスト132に移行されて、ルックアップテーブル41a上の1つの要素に対する画像サイズの設定が解除される。40

【0116】

[データフォーマット設定部の動作]

次に、データフォーマット設定部55の動作について説明する。まず、図14は、データフォーマット41bの構成例を示す図である。

【0117】

図14に示すように、データフォーマット41bは、画像メモリ40のハードウェアI/Fにマッピングされた領域に対応する256個の要素を持つ。各要素には、画像サイズが1次元または2次元のいずれにより表されるかを示すメモリマッピングタイプ(Memory Mapping Type)151と、ルックアップテーブル41aの対応する要素へのポインタ152と、画像メモリ40内の対応領域の先頭アドレスを示すベースアドレス(Base Addre50

ss) 153 が格納される。なお、ルックアップテーブル 41a の対応要素へのポインタ 152 としては、例えば、この要素に対応するルックアップテーブル要素の識別 ID が設定される。

【0118】

図 15 は、データフォーマット設定部 55 によるメモリ領域の管理方法を説明するための図である。

データフォーマット設定部 55 は、メモリ領域のアドレスおよびルックアップテーブル 41a との対応情報を、データフォーマット要素を用いて管理する。データフォーマット要素は、図 15 に示すように、アクティベートリスト 161 およびネゲートリスト 162 の 2 つの線形リストにより管理される。アクティベートリスト 161 には、メモリ領域として現在使用されている、すなわち、確保されたメモリ領域に割り当てられているデータフォーマット要素が接続され、ネゲートリスト 162 には、未使用のデータフォーマット要素が接続される。

【0119】

また、データフォーマット要素は、データフォーマット 41b の要素数だけあらかじめ用意され、初期状態では、すべてのデータフォーマット要素がネゲートリスト 162 に接続される。そして、エンコーダ 20 またはデコーダ 30 による処理の開始時に、必要なメモリ領域の確保が要求されると、ネゲートリスト 162 の先頭から確保されたメモリ領域の数だけのデータフォーマット要素が取り出され、アクティベートリスト 161 の最後尾に接続される。そして、エンコーダ 20 またはデコーダ 30 の処理が終了して、確保されていたメモリ領域の解放が要求されると、対応するすべての要素がアクティベートリスト 161 から順次取り出され、ネゲートリスト 162 の最後尾に接続される。

【0120】

図 16 は、データフォーマット要素のデータ構造を示す図である。

各データフォーマット要素には、データフォーマット設定部 55 により図 16 に示すようなデータが設定される。識別 ID 171 は、各データフォーマット要素を識別するユニークな番号であり、初期設定時に設定される。ノード番号 172 は、メモリ領域の確保時にメモリ確保処理部 52 から出力される二分木構造のノード番号を示す。

【0121】

また、データフォーマット 41b との対応付けを行うために、データフォーマット 41b 内の対応する要素を指示するポインタ 173 が設定される。このポインタ 173 は、初期設定時にあらかじめ設定される。さらに、線形リストを構成するために、同一リストにおいて次に接続されるデータフォーマット要素を指示するポインタ 174 も設定される。

【0122】

また、データフォーマット設定部 55 は、この他に、アクティベートリスト 161 の先頭（アクティベートトップ）および最後尾（アクティベートテール）のデータフォーマット要素をそれぞれ指示するポインタと、ネゲートリスト 162 の先頭（ネゲートトップ）および最後尾（ネゲートテール）のデータフォーマット要素をそれぞれ示すポインタとを図示しないテーブルに保持することで、各リストを管理している。

【0123】

次に、データフォーマット設定部 55 の処理について説明する。まず、図 17 は、メモリ領域が確保される際のデータフォーマット設定部 55 の処理の流れを示すフローチャートである。

【0124】

[ステップ S701] エンコードまたはデコードのためのメモリ領域の確保処理の実行中に、データフォーマット設定部 55 は、メモリ領域管理部 51 から、解放すべきメモリ領域に対応するデータフォーマット要素の解放要求を受け（図 5 のステップ S110 に対応）、以下の処理を開始する。なお、このとき、メモリ領域管理部 51 からは、確保すべきメモリ領域の先頭アドレス、ノード番号、対応するルックアップテーブル要素の識別 ID、画像サイズに基づくメモリマッピングタイプの各情報がデータフォーマット設定部 5

10

20

20

30

40

50

5に供給される。

【0125】

[ステップS702]ネゲートリスト162の先頭(ネゲートトップ)のデータフォーマット要素を取り出し、ネゲートトップの要素を指示するポインタを、次の要素を指示するように更新する。

【0126】

[ステップS703]取り出したデータフォーマット要素を、アクティベートリスト161の最後尾(アクティベートテール)に接続する。具体的には、それまでアクティベートリスト161の最後尾に接続されていたデータフォーマット要素に設定されていた、次の要素へのポインタ174と、アクティベートテールの要素を指示するポインタとを、新たに接続した要素を指示するように更新する。10

【0127】

[ステップS704]当該データフォーマット要素に設定されたポインタ173の指示す、データフォーマット41bの要素に、メモリ領域管理部51から受け取ったメモリマッピングタイプを設定する。

【0128】

[ステップS705]ポインタ173の指示すデータフォーマット41bの対応要素に、ルックアップテーブル41a上の対応要素へのポインタを設定する。この値としては、メモリ領域管理部51から受け取ったルックアップテーブル要素の識別IDが設定される。なお、この識別IDは、このフローチャートの開始時以外に、例えばステップS705の処理時に、メモリ領域管理部51を通じてルックアップテーブル設定部54に問い合わせることにより取得してもよい。20

【0129】

[ステップS706]ポインタ173の指示すデータフォーマット41bの対応要素に、メモリ領域管理部51から受け取った先頭アドレスを設定する。

[ステップS707]メモリ領域管理部51から受け取ったノード番号を、当該データフォーマット要素のノード番号172の欄に設定する。

【0130】

[ステップS708]当該データフォーマット要素の識別ID171を、メモリ領域管理部51に渡す。メモリ領域管理部51は、データフォーマット設定部55からの識別IDを受け取ることで、画像メモリ要素とデータフォーマット要素とを対応付ける(図5のステップS111に対応)。30

【0131】

以上の処理により、エンコーダ20またはデコーダ30の処理の開始時に、確保された各メモリ領域に対応するデータフォーマット要素が、アクティベートリスト161の最後尾に順次生成される。

【0132】

次に、図18は、メモリ領域が解放される場合のデータフォーマット設定部55の処理の流れを示す図である。

[ステップS801]エンコーダ20またはデコーダ30の処理が一旦終了する場合には、メモリ領域管理部51は、1つの画像メモリ要素を解放する処理中に、その画像メモリ要素に対応するデータフォーマット要素を解放するように、データフォーマット設定部55に対して要求する(図8のステップS406に対応)。この要求では、例えば、解放するメモリ領域(画像メモリ要素)の保持するポインタ124の設定値(識別ID)が指定される。データフォーマット設定部55は、メモリ領域管理部51からの要求を受けると、以下の処理を開始する。40

【0133】

[ステップS802]アクティベートリスト161の先頭(アクティベートトップ)のデータフォーマット要素を参照する。

[ステップS803]参照したデータフォーマット要素が、メモリ領域管理部51から50

の要求に対応する要素であるか否かを識別IDから判定し、対応しない場合はステップS804に進み、対応する場合はステップS805に進む。

【0134】

[ステップS804] アクティベートリスト161上の次のデータフォーマット要素を参照し、ステップS803に進む。

[ステップS805] 当該データフォーマット要素を、アクティベートリスト161から取り出す。このとき、必要に応じて、アクティベートトップを指し示すポインタ、アクティベートテールを指し示すポインタ、取り出した要素の直前要素に設定された、次の要素へのポインタ174などを更新する。

【0135】

[ステップS806] 取り出したデータフォーマット要素に対応するルックアップテーブル要素の解放を、メモリ領域管理部51に要求する。これにより、メモリ領域管理部51は、対応するルックアップテーブル要素の識別IDを指定して、この要素を解放するよう、ルックアップテーブル設定部54に対して要求する(図8のステップS407に対応)。そして、ルックアップテーブル設定部54が、この要求に応じて、図13に示したルックアップテーブル要素の解放処理を実行する。

【0136】

[ステップS807] 当該データフォーマット要素に対応付けられたデータフォーマット41bの要素の設定値をクリアし、この要素を対応する。

[ステップS808] 当該データフォーマット要素を、ネゲートリスト162の最後尾に接続し、それまでネゲートリスト162の最後尾に接続されていた要素の、次の要素を指し示すポインタ174と、ネゲートテールを指し示すポインタとを、新たに接続した要素を指し示すように更新する。

【0137】

以上の処理が、エンコーダ20またはデコーダ30の処理のために確保されていたメモリ領域(画像メモリ要素)を1つずつ解放するたびに実行され、対応するデータフォーマット要素が、アクティベートリスト161からネゲートリスト162に順次移行されて、データフォーマット41b上の1つの要素に対するアドレスなどの設定が解除される。

【0138】

[メモリ確保処理部およびメモリ解放処理部の動作]

図19は、メモリ確保処理部52およびメモリ解放処理部53によるメモリ領域の割り当て方法を説明するための図である。

【0139】

メモリ確保処理部52およびメモリ解放処理部53は、画像メモリ40の記憶領域を2のべき乗だけ分割した単位領域を、二分木構造の最も深いノードに割り当てることで管理する。最も深い位置以外のノードは、その子ノードに割り当てられたすべての単位領域を包含する。そして、メモリ確保処理部52およびメモリ解放処理部53は、各単位領域の先頭アドレスを保持し、ノードを指定することにより、そのノードに対応する単位領域の先頭アドレスを出力することが可能となっている。

【0140】

図19の例では、説明を簡単にするために、画像メモリ40を、単位領域40a～40dの4つに分割している。そして、7つのノードからなる二分木構造を利用し、その最も深い位置のノード“3”～“6”に、各単位領域40a～40dを割り当てている。

【0141】

また、メモリ領域の確保および解放の処理のために、二分木構造における階層の深さ(Depth)、同階層のノードの順番を示すオフセット(Offset)、同階層におけるノード数を示す階層幅(Width)の値が利用される。ここで、図19の例では、階層の深さ“1”的ノード“1”は単位領域40aおよび40bを包含し、ノード“2”は同様に単位領域40cおよび40dを包含する。また、深さ“0”的ノード“0”は、単位領域40a～40dを包含する。

10

20

30

40

50

【0142】

図20は、メモリ確保処理部52およびメモリ解放処理部53による二分木構造のノード管理方法を説明するための図である。

単位領域の使用状態は、二分木構造の各ノードの状態（ノードステータス）を用いて管理される。ノードステータスは、未使用状態（確保されていない状態）を“NONE”、そのノードに対応する単位領域が使用状態（確保されている状態）を“USED”、そのノードの子ノードが使用中である状態を“USING”、そのノードあるいは子ノードが解放処理中である状態を“FREE”として表す。

【0143】

各ノードのノードステータスは、メモリ確保処理部52およびメモリ解放処理部53により、共通のノードステータス管理テーブルを用いて管理される。例えば、図19に示した二分木構造において、各ノードが図20（A）に示すような状態である場合、ノードステータス管理テーブルは図20（B）のように設定される。ノードステータス管理テーブルは、例えば、メモリ領域管理部51が保持し、メモリ確保処理部52およびメモリ解放処理部53による以下の図21および図22の処理中に設定され、また参照される。

【0144】

図21は、メモリ領域が確保される場合のメモリ確保処理部52の処理の流れを示すフローチャートである。

[ステップS901] エンコードまたはデコードのためのメモリ領域を確保する処理の実行中に、メモリ領域管理部51から、確保すべきメモリ領域の画像サイズがメモリ確保処理部52に渡される（図5のステップS105に対応）。メモリ確保処理部52は、この画像サイズを受け取って、以下の処理を開始する。

【0145】

[ステップS902] 階層の深さ“0”的ノードを参照する。

[ステップS903] メモリ領域管理部51に指定された画像サイズに対応するデータサイズが、参照しているノードが包含するメモリ領域のデータサイズより小さいか否かを判定する。小さい場合はステップS904に進み、小さくない場合はステップS905に進む。

【0146】

[ステップS904] 階層の深さの値に1を加算し、参照するノードの階層を1つ下げる。

[ステップS905] 参照する階層の深さの値から1を減算する。これにより、確保すべき画像サイズに対して適切な階層が選択される。そして、参照するノードのオフセットの値を“0”として、現在の階層の階層幅を設定する。

【0147】

[ステップS906] オフセットの値が、階層幅の値より小さいか否かを判定し、小さい場合はステップS907に進み、小さくない場合はステップS909に進む。

[ステップS907] 参照しているノードのノードステータスが“NONE”であるか否かを判定し、“NONE”である場合はステップS910に進み、そうでない場合はステップS908に進む。

【0148】

[ステップS908] 参照しているノードが使用中または解放処理中であることから、オフセットの値に1を加算し、同階層の次のノードを参照する。そして、ステップS906に戻る。なお、この後のステップS906において、オフセットの値が階層幅以上になり、同階層のすべてのノードが使用中と判定された場合に、ステップS909に進むことになる。

【0149】

[ステップS909] 階層の深さの値から1を減算し、1つ上位の階層のノードを参照する。そして、オフセットの値を“0”とし、現在の階層の階層幅を設定して、ステップS906に戻る。

10

20

30

40

50

【0150】

[ステップS910] 参照しているノードのノードステータスを“USED”とし、さらにその親ノードのノードステータスを“USING”とする。

[ステップS911] 参照しているノードのノード番号を、メモリ領域管理部51に返す。

【0151】

[ステップS912] 当該ノードの包含するメモリ領域の先頭アドレスを、メモリ領域管理部51に返す。

以上の処理により、指定された画像サイズの画像データを格納するために必要最小限のメモリ領域が確保され、ステップS911およびS912の処理により、メモリ領域管理部51は、確保されたメモリ領域に対応するノード番号と、その領域の先頭アドレスを受け取る(図5のステップS106)。

10

【0152】

次に、図22は、メモリ領域が解放される場合のメモリ解放処理部53の処理の流れを示すフローチャートである。

[ステップS1001] エンコーダ20またはデコーダ30の処理が一旦終了する場合には、メモリ領域管理部51は、1つの画像メモリ要素に対応付けられたノード番号を、メモリ解放処理部53に渡し、そのメモリ領域の解放を要求する(図8のステップS405に対応)。メモリ解放処理部53は、メモリ領域管理部51からのノード番号を受け取り、以下の処理を開始する。

20

【0153】

[ステップS1002] 受け取ったノード番号に対応する階層の深さ、オフセットの値を求める。

[ステップS1003] 求められた階層の深さおよびオフセットに該当するノードのノードステータスを“FREE”に設定する。

【0154】

[ステップS1004] 同じ親ノードを持つ同階層の他のノードのノードステータスが“FREE”である場合はステップS1005に進み、そうでない場合は処理を終了する。

30

【0155】

[ステップS1005] 同じ親ノードを持つ同階層の他のノードのノードステータスを“NONE”とし、その親ノードのノードステータスを“FREE”とする。

以上の処理により、確保されていた1つの画像メモリ要素に対応するメモリ領域が解放される。そして、1つの画像サイズ要素中のすべての画像メモリ要素が解放された後、上記のステップS1004およびS1005で“FREE”とされたノードのノードステータスを“NONE”に変更し、解放したメモリ領域を使用可能な状態とする。

40

【0156】

[第1の実施の形態における効果]

以上説明した画像処理装置では、エンコーダ20またはデコーダ30の一方の処理における画像サイズの設定変更を、他方の処理を停止することなく行うことが可能となる。例えば、上記の画像処理装置がデジタルビデオカメラに搭載された場合を想定すると、撮像された動画像は、エンコーダ20によりエンコードされ、磁気テープなどの記録媒体に記録される。また、記録媒体から読み出された動画像の符号化ストリームは、デコーダ30によりデコードされて、ディスプレイに表示される。

40

【0157】

さらに、磁気テープに記録された動画像の符号化ストリームを、その画像サイズを小さくして再エンコードし、メモリカードに記録する、あるいはネットワークを通じて外部機器に送信する場合が想定される。この場合、磁気テープから読み出された符号化ストリームがデコーダ30によりデコードされた後、エンコーダ20により異なる画像サイズでエンコードされる。この処理において、本実施の形態では、磁気テープから符号化ストリー

50

ムを読み出し、デコーダ30によりデコードする処理を続けたまま、エンコーダ20によるエンコード時の画像サイズを途中で変更することが可能となる。

【0158】

また、画像サイズの異なる動画像の符号化ストリームをメモリカード内に用意しておき、これらを再エンコードしてネットワークを通じて外部機器に送信する場合を想定すると、一定の画像サイズでのエンコーダ20によるエンコードを続けたまま、メモリカード内の符号化ストリームを次々に読み出し、デコーダ30によりデコードしてエンコーダ20に供給することが可能となる。

【0159】

このような効果は、主に、二分木構造を用いてメモリ領域の確保および解放を行うメモリ確保処理部52およびメモリ解放処理部53による処理と、確保されたメモリ領域の使用状態を線形リストを用いて一元的に管理するメモリ領域管理部51の処理により得られるものである。すなわち、メモリ確保処理部52およびメモリ解放処理部53は、エンコードおよびデコードの際に1ピクチャ分の画像データを格納するために必要とされる最低限のメモリ領域を、二分木構造を用いることで画像メモリ40上の空き領域から効率的に確保することが可能となっている。

【0160】

また、メモリ領域管理部51は、エンコードおよびデコードのために必要なメモリ領域の確保を、1ピクチャ分ずつ個別にメモリ確保処理部52に要求し、それらの使用状態を管理することから、エンコーダ20またはデコーダ30がそれぞれの処理の際に使用するメモリ領域は、実際には画像メモリ40内の散らばった領域に割り当てられ、固定的には割り当てられない。従って、必要ななくなったメモリ領域のみ解放して、解放した領域を空き領域として容易に再利用することが可能となるので、エンコーダ20およびデコーダ30の処理中における画像サイズの変更に対して柔軟に対応することが可能となる。

【0161】

また、エンコーダ20およびデコーダ30は、画像メモリ40の仕様や物理アドレスを認識することなく、メモリ領域管理部51から受け取った画像メモリ要素の識別IDのみを用いて、画像メモリ40へのデータの書き込みおよび読み出しを容易に行うことが可能となる。従って、エンコーダ20およびデコーダ30の回路規模や製造コストが削減される。

【0162】

また、ルックアップテーブル設定部54およびデータフォーマット設定部55が、確保されたメモリ領域の使用が可能となるように、画像メモリ40のハードウェアI/Fに合わせて必要な設定を行うので、画像メモリ40の仕様が変更された際には、これらの機能ブロックの処理手順のみ変更することで、システムを柔軟に対応することが可能となる。

【0163】

〔他の実施の形態例〕

図23は、本発明の第2の実施の形態に係る画像処理装置の要部構成を示すブロック図である。

【0164】

図23に示す画像処理装置では、2つのエンコーダ20aおよび20bにより並行してエンコードを行うことが可能となっている。例えば、2つのエンコーダ20aおよび20bに対して、撮像画像などの同一の動画像データおよび音声データを供給して、それぞれに異なる画像サイズでのエンコードを実行させ、一方の符号化ストリームを磁気テープに記録し、他方をメモリカードなどの他の記録媒体に記録する、あるいはネットワークを通じて外部機器に送信することが可能となっている。

【0165】

この画像処理装置では、第1の実施の形態と同様に、エンコーダ20aおよび20bによる画像メモリ40へのアクセスは、メモリ制御部50を通じて行われる。これにより、一方のエンコードを続行させたままで、他方のエンコードを停止させて画像サイズを変更

10

20

30

40

50

し、エンコードを再開させることが可能となる。

【0166】

図24は、本発明の第3の実施の形態に係る画像処理装置の要部構成を示すブロック図である。

図24に示す画像処理装置では、2つのデコーダ30aおよび30bにより並行してデコードを行うことが可能となっている。例えば、2つのデコーダ30aおよび30bに対して、それぞれ異なる符号化ストリームを供給してデコードを並行して実行させ、双方から出力される動画像データを合成し、1つの画面に合成画像を表示させることができとなっている。

【0167】

この画像処理装置では、第1の実施の形態と同様に、デコーダ30aおよび30bによる画像メモリ40へのアクセスは、メモリ制御部50を通じて行われる。これにより、一方のデコードを続行させたままで、他方のデコードを停止させて画像サイズを変更し、デコードを再開させることができる。

【0168】

また、本発明は、動画像のエンコード・デコードに限らず、複数系統の画像処理を、同一の画像メモリを共用して実行する場合に適用することが可能である。このような画像処理としては、例えば画像の拡大・縮小、合成、画質補正などが適用可能である。さらに、1系統の画像処理を行う場合に本発明を適用した場合にも、画像メモリの仕様に関係なく、画像処理エンジンの構成が簡素化されるという効果が得られる。

【0169】

また、本発明は、画像処理や画像メモリへのアクセス管理などの処理が、OSを搭載しない組み込み型システムにより実行される場合に特に好適である。しかし、汎用OSを搭載した組み込み型システムに本発明を適用することも可能である。この場合、メモリ制御部50に相当する処理プログラムが汎用OS上で実行されるが、このプログラムは画像メモリ管理に特化しているため、汎用OSによる画像メモリの管理処理より性能が優れていると言える。

【0170】

なお、上記の各実施の形態で示したメモリ制御部50の処理機能は、コンピュータによって実現することができる。その場合、メモリ制御部50が有すべき機能の処理内容を記述したプログラムが提供される。そのプログラムをコンピュータで実行することにより、上記処理機能がコンピュータ上で実現される。処理内容を記述したプログラムは、コンピュータで読み取り可能な記録媒体に記録しておくことができる。このような記録媒体としては、磁気記録装置、光ディスク、光磁気記録媒体、半導体メモリなどがある。

【0171】

プログラムを流通させる場合には、例えば、半導体メモリなどからなる可搬型記録媒体にそのプログラムが記録され、この可搬型記録媒体が販売される。また、プログラムをサーバコンピュータの記憶装置に格納しておき、ネットワークを介して、サーバコンピュータから他のコンピュータにそのプログラムを転送することもできる。

【0172】

プログラムを実行するコンピュータは、例えば、可搬型記録媒体に記録されたプログラムまたはサーバコンピュータから転送されたプログラムを、自己の記憶装置に格納する。そして、コンピュータは、自己の記憶装置からプログラムを読み取り、プログラムに従った処理を実行する。なお、コンピュータは、可搬型記録媒体から直接プログラムを読み取り、そのプログラムに従った処理を実行することもできる。

【図面の簡単な説明】

【0173】

【図1】本発明の第1の実施の形態に係る画像処理装置の要部構成を示すブロック図である。

【図2】メモリ制御部の内部構成を示すブロック図である。

10

20

30

40

50

【図3】確保したメモリ領域のメモリ領域管理部による管理方法を説明するための図である。

【図4】画像サイズ要素および画像メモリ要素のデータ構造を示す図である。

【図5】メモリ領域を確保する際のメモリ領域管理部の処理の流れを示すフローチャートである。

【図6】確保されたメモリ領域をエンコーダまたはデコーダに提供する際のメモリ領域管理部の処理の流れを示すフローチャートである。

【図7】エンコーダまたはデコーダからメモリ領域が返却される際のメモリ領域管理部の処理の流れを示すフローチャートである。

【図8】メモリ領域を解放する際のメモリ領域管理部の処理の流れを示すフローチャート 10 である。

【図9】ロックアップテーブルの構成例を示す図である。

【図10】ロックアップテーブル設定部によるメモリ領域の管理方法を説明するための図である。

【図11】ロックアップテーブル要素のデータ構造を示す図である。

【図12】メモリ領域が確保される際のロックアップテーブル設定部の処理の流れを示すフローチャートである。

【図13】メモリ領域が解放される場合のロックアップテーブル設定部の処理の流れを示すフローチャートである。

【図14】データフォーマットの構成例を示す図である。

【図15】データフォーマット設定部によるメモリ領域の管理方法を説明するための図である。

【図16】データフォーマット要素のデータ構造を示す図である。

【図17】メモリ領域が確保される際のデータフォーマット設定部の処理の流れを示すフローチャートである。

【図18】メモリ領域が解放される場合のデータフォーマット設定部の処理の流れを示す図である。

【図19】メモリ確保処理部およびメモリ解放処理部によるメモリ領域の割り当て方法を説明するための図である。

【図20】メモリ確保処理部およびメモリ解放処理部による二分木構造のノード管理方法 30 を説明するための図である。

【図21】メモリ領域が確保される場合のメモリ確保処理部の処理の流れを示すフローチャートである。

【図22】メモリ領域が解放される場合のメモリ解放処理部の処理の流れを示すフローチャートである。

【図23】本発明の第2の実施の形態に係る画像処理装置の要部構成を示すブロック図である。

【図24】本発明の第3の実施の形態に係る画像処理装置の要部構成を示すブロック図である。

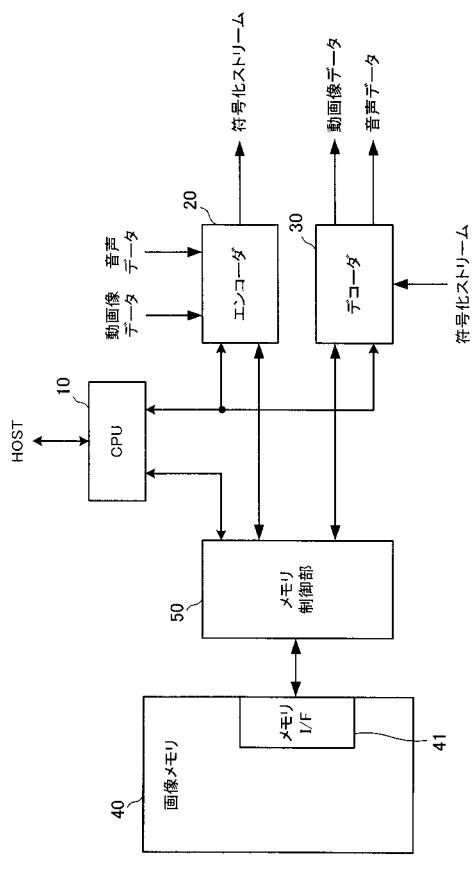
【図25】動画像のエンコードとデコードを並行して実行することが可能な画像処理装置 40 の要部構成を示す図である。

【符号の説明】

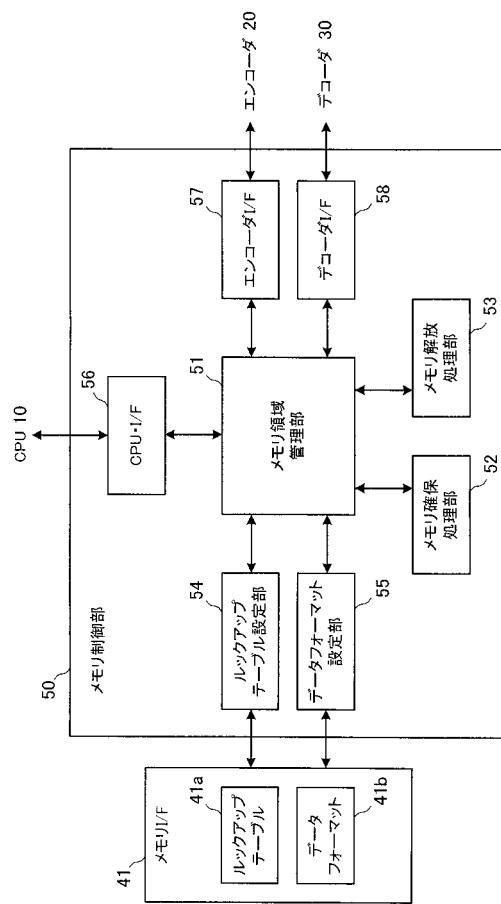
【0174】

10 …… CPU、 20 …… エンコーダ、 30 …… デコーダ、 40 …… 画像メモリ、 41 …… メモリI/F、 41a …… ロックアップテーブル、 41b …… データフォーマット、 50 …… メモリ制御部、 51 …… メモリ領域管理部、 52 …… メモリ確保処理部、 53 …… メモリ解放処理部、 54 …… ロックアップテーブル設定部、 55 …… データフォーマット設定部、 56 …… CPU・I/F、 57 …… エンコーダI/F、 58 …… デコーダI/F

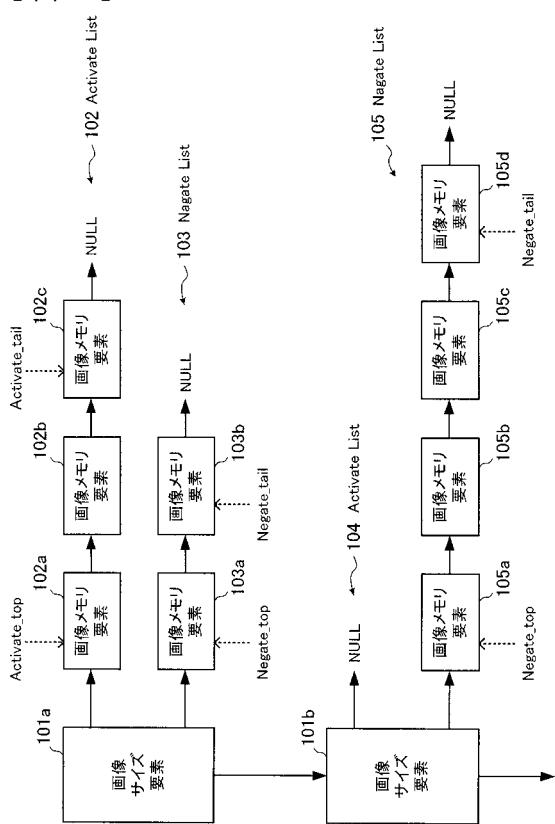
【図1】



【図2】

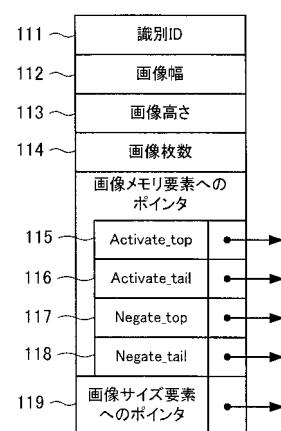


【図3】

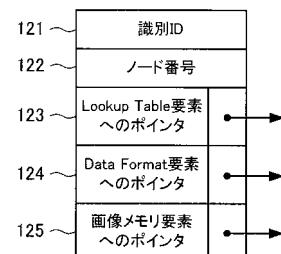


【図4】

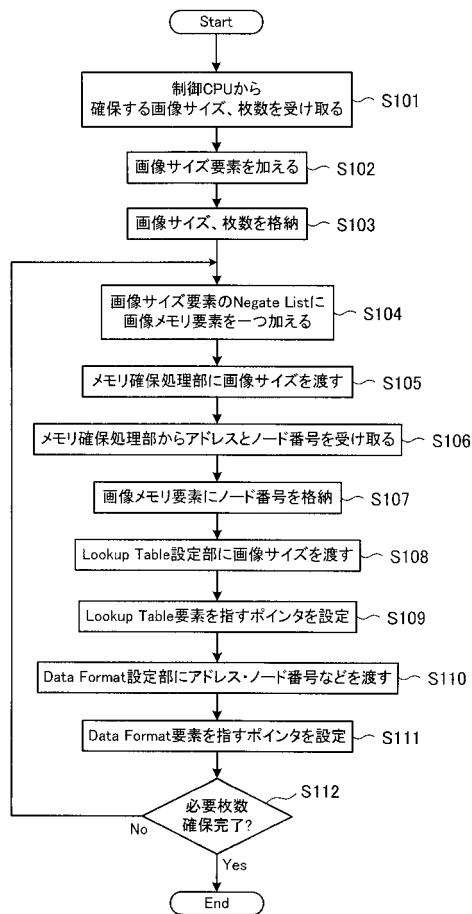
(A) 画像サイズ要素のデータ構造



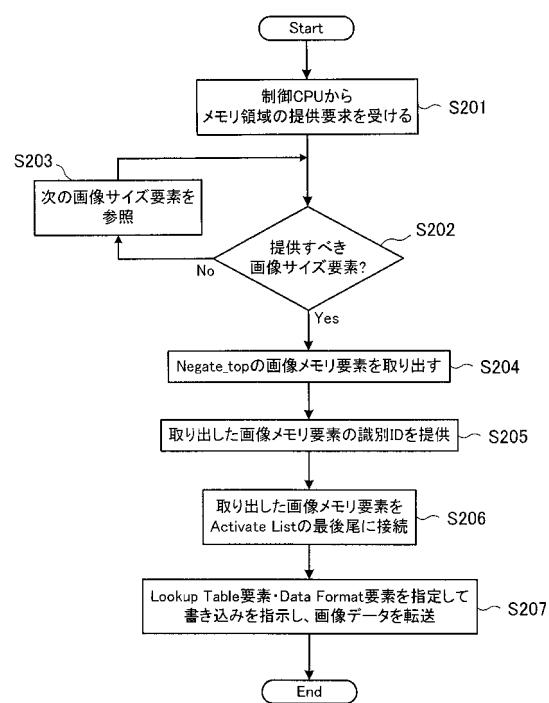
(B) 画像メモリ要素のデータ構造



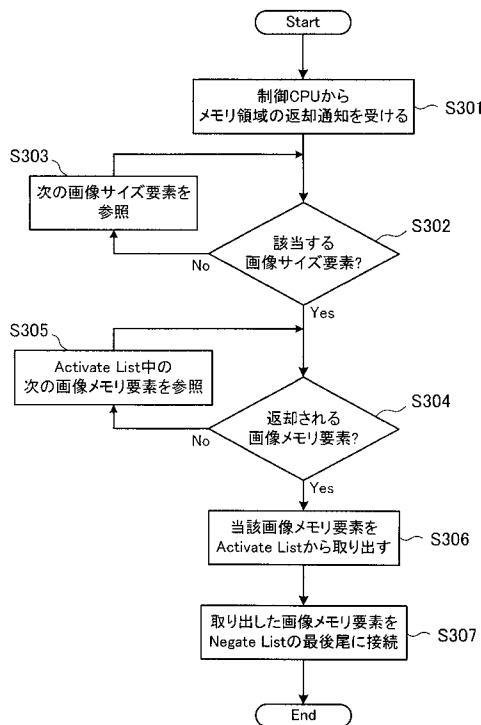
【図5】



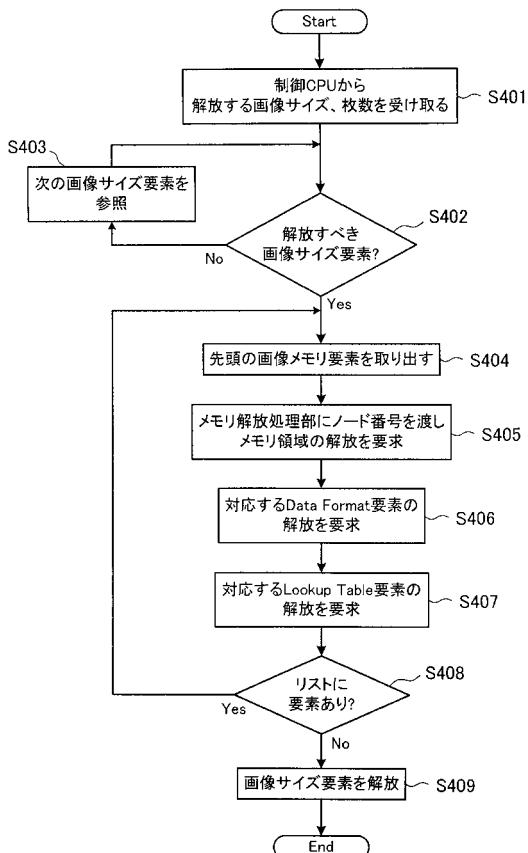
【図6】



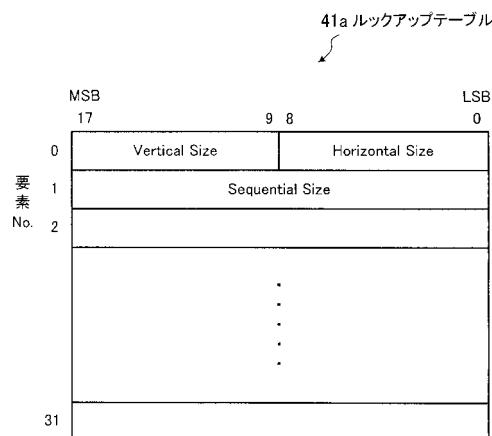
【図7】



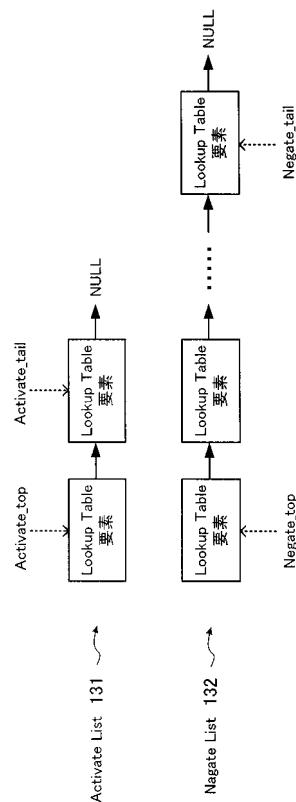
【図8】



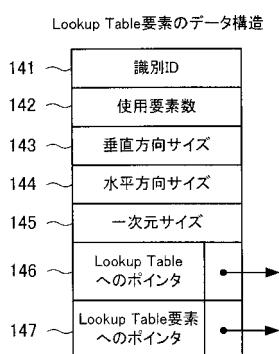
【図9】



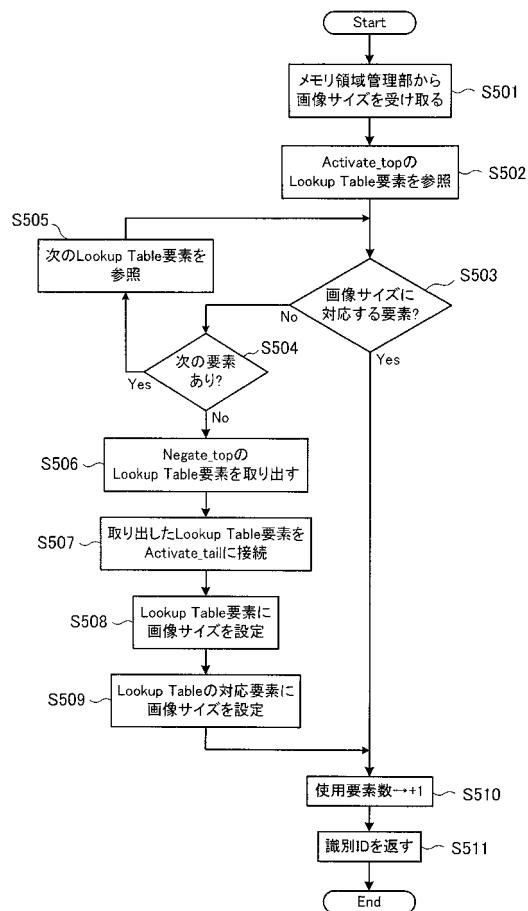
【図10】



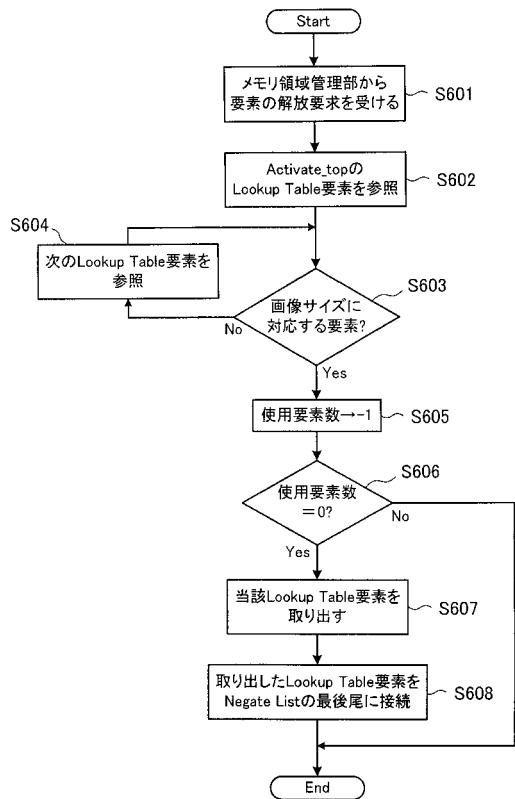
【図11】



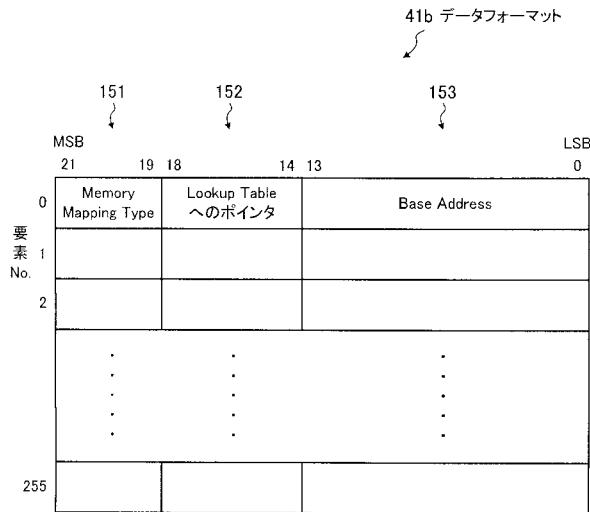
【図12】



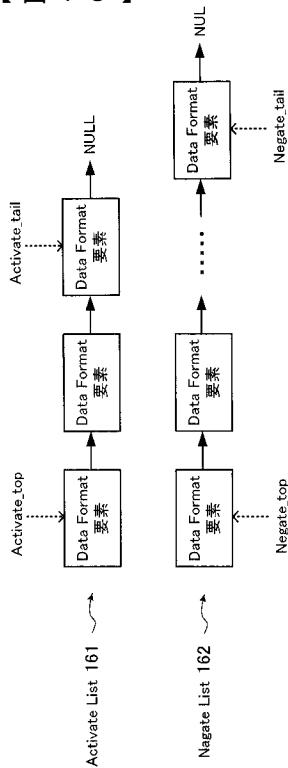
【図13】



【図14】

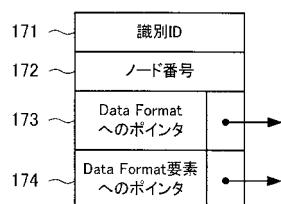


【図15】

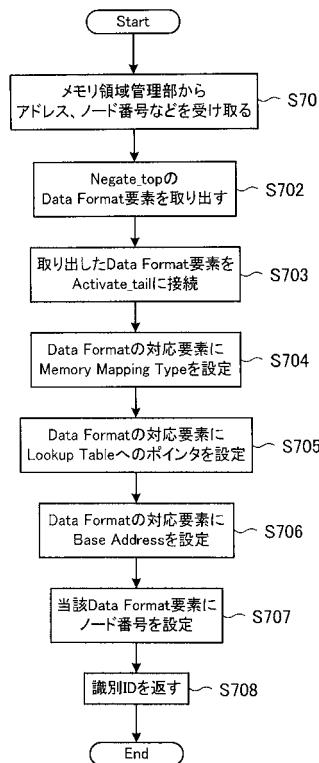


【図16】

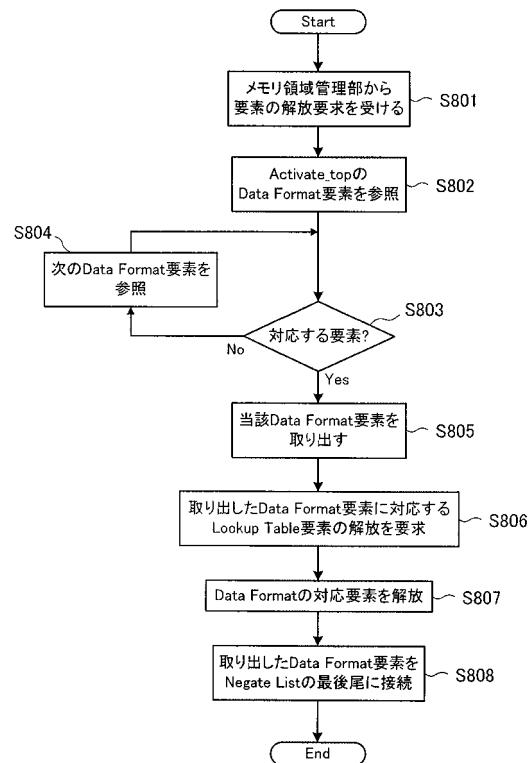
Data Format要素のデータ構造



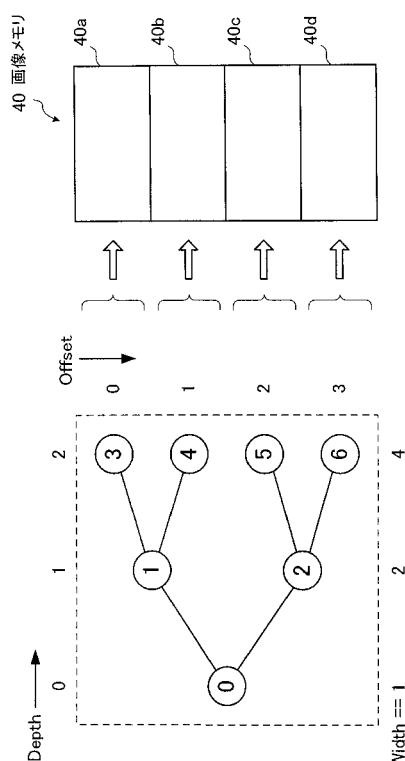
【図17】



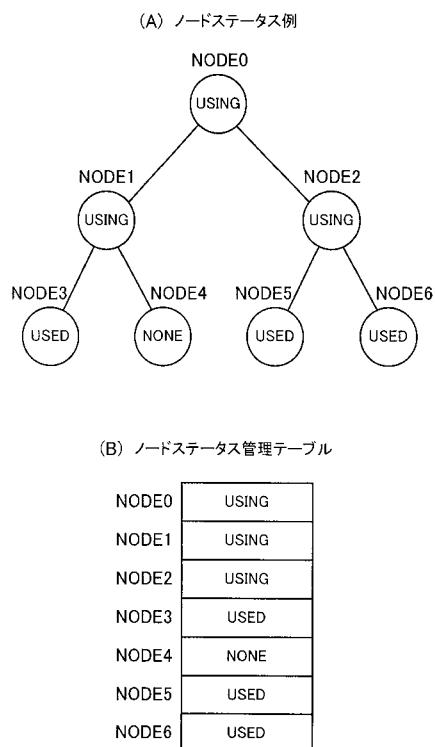
【図18】



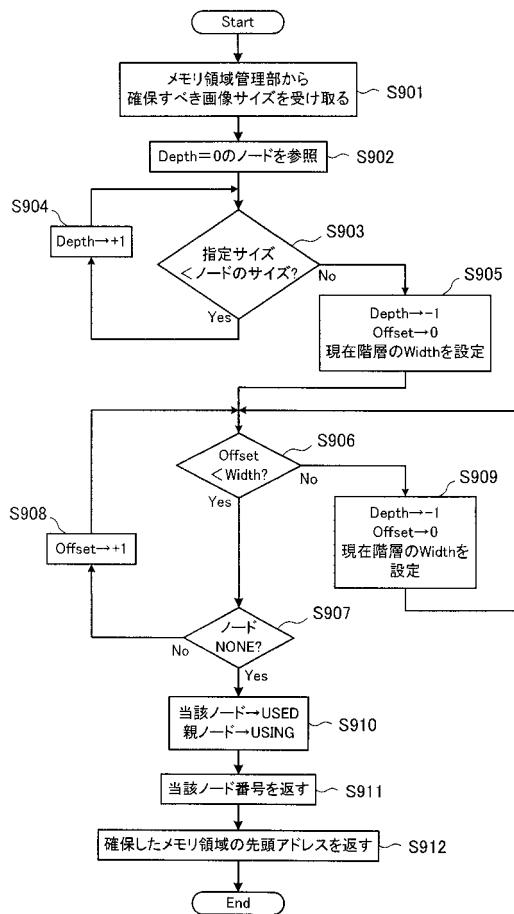
【図19】



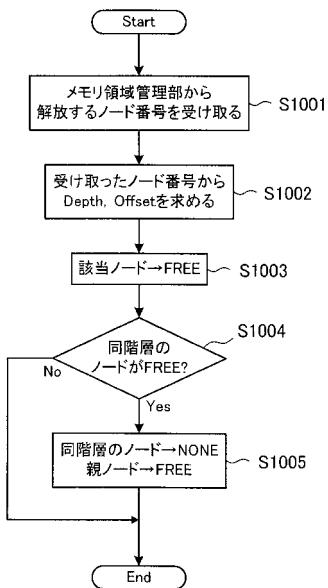
【図20】



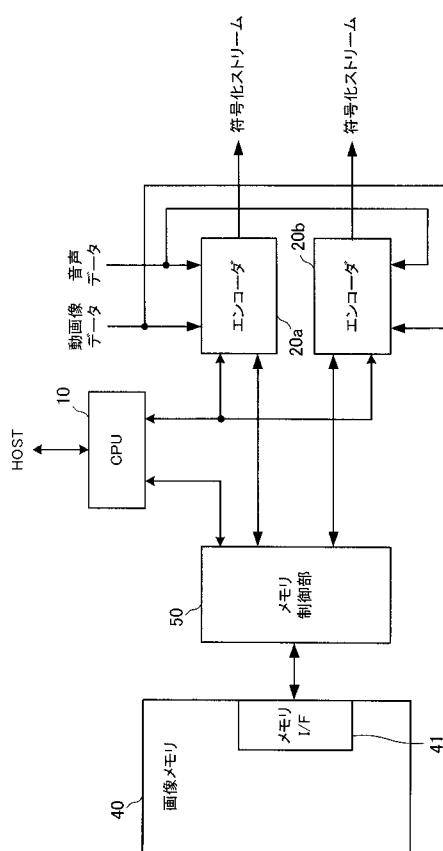
【図21】



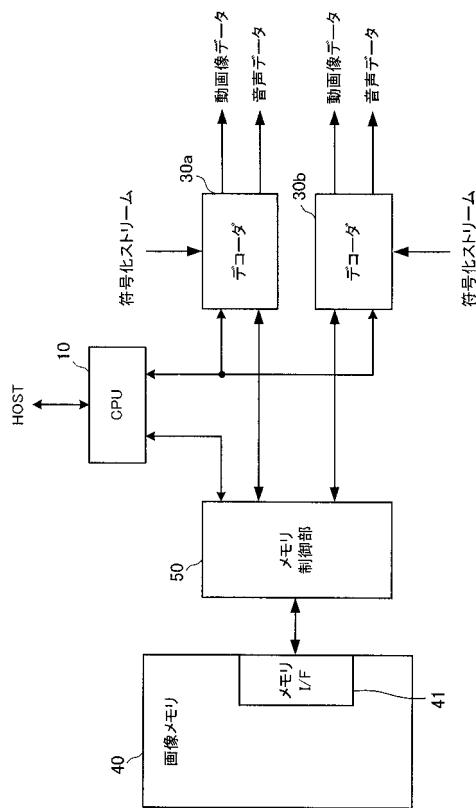
【図22】



【図23】



【図24】



【図25】

