

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4242452号
(P4242452)

(45) 発行日 平成21年3月25日 (2009. 3. 25)

(24) 登録日 平成21年1月9日 (2009. 1. 9)

(51) Int. Cl.

F I

H04N 7/26 (2006.01)

H04N 7/13

Z

請求項の数 20 (全 43 頁)

(21) 出願番号	特願平9-514434	(73) 特許権者	マイクロソフト コーポレーション
(86) (22) 出願日	平成8年10月4日 (1996. 10. 4)		アメリカ合衆国 ワシントン州 9805
(65) 公表番号	特表平11-512911		2-6399 レッドモンド ワン マイ
(43) 公表日	平成11年11月2日 (1999. 11. 2)		クロソフト ウェイ
(86) 国際出願番号	PCT/US1996/015892	(74) 代理人	弁理士 杉村 憲司
(87) 国際公開番号	W01997/013372	(74) 代理人	弁理士 藤原 英治
(87) 国際公開日	平成9年4月10日 (1997. 4. 10)	(74) 代理人	弁理士 杉村 興作
審査請求日	平成15年9月29日 (2003. 9. 29)	(74) 代理人	弁理士 徳永 博
(31) 優先権主張番号	60/005, 031	(74) 代理人	弁理士 高見 和明
(32) 優先日	平成7年10月5日 (1995. 10. 5)		
(33) 優先権主張国	米国 (US)		

最終頁に続く

(54) 【発明の名称】 ビデオ圧縮方法及び装置

(57) 【特許請求の範囲】

【請求項 1】

ビデオ画像オブジェクトの画素値を外挿する外挿方法であって、ビデオ画像オブジェクトの辺縁の外側に位置する少なくとも1つの画素の値を規定し、前記辺縁の外側に位置する画素は、前記画像オブジェクトの存するブロックの境界内に位置するものであり、ブロック境界内の画素のラインを走査するステップであって、画像オブジェクトの辺縁の外側で少なくとも1つの画素を識別し、これらの識別された各画素は2つの端部の画素を有するセグメントの一部であり、2つの端部の画素のうちの少なくとも1つの画素は辺縁画素の値を有する、ステップと、各識別された画素に対し、識別された画素を含むセグメントの両端の画素が辺縁画素の値を有する場合は、識別された画素に辺縁画素の値の平均値を割当て、識別された画素を含むセグメントの端部の画素の1つのみが辺縁画素の値を有する場合は、識別された画素に辺縁画素の値を割当てるステップと、を含むことを特徴とする外挿方法。

【請求項 2】

ブロック境界を水平に亘って前記走査を行うことを特徴とする、請求項 1 記載の外挿方法。

【請求項 3】

ブロック境界内で、複数の画素のラインに対して繰り返すステップを更に含むことを特徴とする、請求項 1 記載の外挿方法。

【請求項 4】

前記走査するステップは、

ブロック境界内の水平ラインを走査するステップと、

水平ライン内の各識別画素の決定に続き、ブロック境界内で垂直ラインの走査を行うステップとを含むことを特徴とする、請求項 3 記載の外挿方法。

【請求項 5】

画像オブジェクトの辺縁の外側で、かつ水平ライン走査で識別されなかったか、又は垂直ライン走査で識別されず残っている少なくとも 1 つの画素に対し、以前に付与した辺縁画素の値を割当てするステップを更に含むことを特徴とする、請求項 3 記載の外挿方法。

10

【請求項 6】

画像オブジェクトの辺縁において残留している少なくとも 1 つの画素で、走査において識別されていない画素に対し、これ迄に割当ててきた辺縁画素の値を割当てするステップを含むことを特徴とする、請求項 1 記載の外挿方法。

【請求項 7】

ブロック境界は、画素の行及び列を含み、画像オブジェクトの辺縁外の少なくとも 1 つの残りの画素は、ビデオの画像オブジェクトの画素を含んでいる行又は列の画素以外のものであることを特徴とする、請求項 6 記載の外挿方法。

【請求項 8】

20

コンピュータによって、ビデオ画像オブジェクトの画素値を規定するため、該ビデオ画像オブジェクトの辺縁の外側の少なくとも 1 つの画素値を外挿する方法であって、前記ビデオ画像オブジェクトの辺縁の外側の少なくとも 1 つの画素は、該オブジェクトの周のブロック境界の内側であるコンピュータによるビデオ外挿方法において、

ブロック境界内において、1 つ以上の画素の水平セグメントにおける複数の水平ラインを走査するステップであって、各水平セグメントは画像オブジェクトの辺縁外の 1 つ以上の画素を有し、かつ各水平セグメントは 2 つの端部の画素を有し、少なくとも 1 つの端部の画素は辺縁画素の値を有する、ステップと、

1 つ以上の水平セグメントにおけるオブジェクトの辺縁外の各画素に対し、

水平セグメントの両端の画素が辺縁画素の値を有する場合には、辺縁外の画素に、前記両端の画素の辺縁画素値の平均値を割当てするステップと、

30

その他の場合には、辺縁外の画素に、辺縁画素の値を割当てするステップとを有し、かつ

ブロック境界内において、1 つ以上の画素の垂直セグメントにおける複数の垂直ラインを走査するステップであって、各垂直セグメントはオブジェクトの辺縁外の 1 つ以上の画素を有し、かつ各垂直セグメントは 2 つの端部の画素を有し、少なくとも 1 つの端部の画素が辺縁画素の値を有する、ステップと、

1 つ以上の垂直セグメントにおけるオブジェクトの辺縁外の各画素に対し、

垂直セグメントの両端の画素がオブジェクト辺縁画素の値を有する場合には、辺縁外の画素に、両端の画素の辺縁画素値の平均値を割当てするステップと、

その他の場合には、辺縁外の画素に、辺縁画素の値を割当てするステップとを含むことを特徴とする、コンピュータによるビデオ外挿方法。

40

【請求項 9】

画像オブジェクトの辺縁外の少なくとも 1 つの残留画素に、従前に割当てた辺縁画素の値を割当てすることを特徴とする、請求項 8 記載の方法。

【請求項 10】

ブロック境界は、画素の行及び列を含み、画像オブジェクトの辺縁外の少なくとも 1 つの残りの画素は、ビデオの画像オブジェクトの画素を含んでいる行又は列の画素以外のものであることを特徴とする、請求項 9 記載の方法。

【請求項 11】

ビデオ画像オブジェクトの画素値を外挿する外挿装置であって、

50

ビデオ画像オブジェクトの辺縁の外側に位置する少なくとも1つの画素の値を規定し、前記辺縁の外側に位置する画素は、前記画像オブジェクトの存するブロックの境界内に位置するものであり、

ブロック境界内の画素のラインを走査する手段であって、画像オブジェクトの辺縁の外側で少なくとも1つの画素を識別し、これらの識別された各画素は2つの端部の画素を有するセグメントの一部であり、2つの端部の画素のうちの少なくとも1つの画素は辺縁画素の値を有する、手段と、

各識別された画素に対し、

識別された画素を含むセグメントの両端の画素が辺縁画素の値を有する場合は、識別された画素に辺縁画素の値の平均値を割当て、

識別された画素を含むセグメントの端部の画素の1つのみが辺縁画素の値を有する場合は、識別された画素に辺縁画素の値を割当てる手段と、

を有することを特徴とする外挿装置。

【請求項12】

ブロック境界を水平に亘って前記走査を行うことを特徴とする、請求項11記載の外挿装置。

【請求項13】

ブロック境界内で、複数の画素のラインに対して繰り返す手段を更に有することを特徴とする、請求項11記載の外挿装置。

【請求項14】

前記走査する手段は、

ブロック境界内の水平ラインを走査する手段と、

水平ライン内の各識別画素の決定に続き、ブロック境界内で垂直ラインの走査を行う手段とを有することを特徴とする、請求項13記載の外挿装置。

【請求項15】

画像オブジェクトの辺縁の外側で、かつ水平ライン走査で識別されなかったか、又は垂直ライン走査で識別されず残っている少なくとも1つの画素に対し、以前に付与した辺縁画素の値を割当てる手段を更に有することを特徴とする、請求項13記載の外挿装置。

【請求項16】

画像オブジェクトの辺縁において残留している少なくとも1つの画素で、走査において識別されていない画素に対し、これ迄に割当ててきた辺縁画素の値を割当てる手段を有することを特徴とする、請求項11記載の外挿装置。

【請求項17】

ブロック境界は、画素の行及び列を含み、画像オブジェクトの辺縁外の少なくとも1つの残りの画素は、ビデオの画像オブジェクトの画素を含んでいる行又は列の画素以外のものであることを特徴とする、請求項16記載の外挿装置。

【請求項18】

ビデオ画像オブジェクトの画素値を規定するため、該ビデオ画像オブジェクトの辺縁の外側の少なくとも1つの画素値を外挿するコンピュータ装置であって、前記ビデオ画像オブジェクトの辺縁の外側の少なくとも1つの画素は、該オブジェクトの周のブロック境界の内側である、コンピュータ装置において、

ブロック境界内において、1つ以上の画素の水平セグメントにおける複数の水平ラインを走査する手段であって、各水平セグメントは画像オブジェクトの辺縁外の1つ以上の画素を有し、かつ各水平セグメントは2つの端部の画素を有し、少なくとも1つの端部の画素は辺縁画素の値を有する、手段と、

1つ以上の水平セグメントにおけるオブジェクトの辺縁外の各画素に対し、

水平セグメントの両端の画素が辺縁画素の値を有する場合には、辺縁外の画素に、前記両端の画素の辺縁画素値の平均値を割当てる手段と、

その他の場合には、辺縁外の画素に、辺縁画素の値を割当てる手段とを有し、かつ

ブロック境界内において、1つ以上の画素の垂直セグメントにおける複数の垂直ラインを

10

20

30

40

50

走査する手段であって、各垂直セグメントはオブジェクトの辺縁外の1つ以上の画素を有し、かつ各垂直セグメントは2つの端部の画素を有し、少なくとも1つの端部の画素が辺縁画素の値を有する、手段と、

1つ以上の垂直セグメントにおけるオブジェクトの辺縁外の各画素に対し、垂直セグメントの両端の画素がオブジェクト辺縁画素の値を有する場合には、辺縁外の画素に、両端の画素の辺縁画素値の平均値を割当てて手段と、

その他の場合には、辺縁外の画素に、辺縁画素の値を割当てて手段とを有することを特徴とする、コンピュータ装置。

【請求項19】

画像オブジェクトの辺縁外の少なくとも1つの残留画素に、従前に割当てた辺縁画素の値を割当ててことを特徴とする、請求項18記載のコンピュータ装置。

10

【請求項20】

ブロック境界は、画素の行及び列を含み、画像オブジェクトの辺縁外の少なくとも1つの残りの画素は、ビデオの画像オブジェクトの画素を含んでいる行又は列の画素以外のものであることを特徴とする、請求項19記載のコンピュータ装置。

【発明の詳細な説明】

技術分野

本発明は、デジタル・ビデオ信号を圧縮する方法に関わる。特に、精度を増すための誤差・フィードバックを備えたオブジェクトベースのデジタル・ビデオ符号化方法に関わる。

20

背景技術

アナログ・ビデオ信号によるフルモーションのビデオ表示は、テレビジョンという形式で長く一般に利用されている。最近の、コンピュータの処理機能や、コンピュータ入手性の増大に伴って、デジタル・ビデオ信号によるフルモーションのビデオ表示がさらに広く一般に利用されるようになってきている。デジタル・ビデオ装置は、従来のアナログ・ビデオ装置に比べて、フルモーションのビデオ画面を生成し、修正し、伝送し、保存し、再生することにおいて目立った改善を加えている。

デジタル・ビデオディスプレイは、多数の画像フレームからなり、このフレームは、30~75Hzの間の周波数で連続的に表示され、即ちレンダリングされる。各画像フレームは、その特定の装置のディスプレイ解像度による、ある配列の画素から形成される静止画像である。例を挙げるならば、VHS装置は、320x480画素のディスプレイ解像度を持ち、NTSC装置は、720x486画素の解像度を持ち、現在開発中のハイ・ディフィニッション・テレビ(high-definition television, HDTV)装置は、1360x1024画素の解像度を持つ。

30

ビデオ配列に含まれる生のデジタル情報の量は莫大である。この莫大量のビデオ情報の保存・伝送は、従来のパソコン装置では不可能である。320x480画素解像度を持つ比較的
低解像度のVHS画像方式におけるデジタル形式に関しては、持続2時間の映画1巻は、100ギガバイトのデジタル・ビデオ信号に相当する。比較として、従来のコンパクト・ディスクは、約0.6ギガバイトの容量を持ち、磁気ハード・ディスクは、1-2ギガバイトの容量を持ち、また、現在開発中の光学的コンパクト・ディスクは、最大8ギガバイトまでの容量を持つ。

40

このような莫大量のデジタル・ビデオ信号を保存・伝送する際に認められる限界を処理するために、各種のビデオ圧縮規格ないし方法が導入されている。その中にはMPEG-1、MPEG-2、および、H.26Xがある。これら従来のビデオ圧縮法は、時間的ないしフレーム間相関と呼ばれる、連続する画像フレーム間の類似性を利用して、フレーム間圧縮を行なう。この圧縮においては、画像フレームの画素表現値が、動き表現値に変換される。さらに、従来のビデオ圧縮法は、空間ないしフレーム内相関と呼ばれる画像フレーム内部の類似性を利用し、フレーム内圧縮を実行する。この圧縮においては、ある画像フレーム内の動き表現値がさらに圧縮される。フレーム内圧縮は、従来の静止画像圧縮方法、例えば、離散コサイン変換(DCT)符号化法に基づく。

特定の導入方法は異なるものの、MPEG-1、MPEG-2、および、H.26Xのビデオ圧縮基準はい

50

くつかの点で類似する。下記のMPEG-2ビデオ圧縮基準に関する記載は、一般に他の二者にも当てはまる。

MPEG-2は、ビデオ画像における方形ブロック、すなわち、画素配列に基づいてフレーム間圧縮とフレーム内圧縮とを提供する。一つのビデオ画像は、16x16画素のサイズを持つ変換ブロックに分割される。ある画像フレームNにおける各変換ブロック T_N について、直前の画像フレームN-1の画像か、又は直後の画像フレームN+1（すなわち両方向性）の画像にも亘って探索を行ない、それぞれにおいてもっとも類似性の高い変換ブロック T_{N+1} 、または、 T_{N+1} を特定する。

尚、直後の画像フレームの探索に関して、変換ブロック T_N および T_{N+1} の画素値は、例えその変換ブロックが、それぞれの画像フレームにおいて異なる位置にある場合でも、同一であるのが理想的である。このような状況においては、変換ブロック T_{N+1} における画素情報は、変換ブロック T_N の画素情報に対して冗長となる。圧縮は、変換ブロック T_{N+1} の画素情報の代わりに、変換ブロック T_N 及び T_{N+1} 間の位置的な線形移動を用いることによって実行する。この単純化した例においては、変換ブロック T_{N+1} における256の画素値で関連付けられたビデオ情報に対して、ただ一つの線形移動ベクトル（X、Y）が指定される。

しばしばあることであるが、対応する変換ブロック T_N および T_{N+1} におけるビデオ情報（すなわち、画素値）は同一ではない。両者の差は変換ブロック誤差Eと呼ばれるが、無視できない場合がしばしばある。この誤差は、離散コサイン変換(DCT)符号化法のような従来の圧縮方法によって圧縮されるのではあるが、この変換ブロック誤差Eは厄介なものであり、ビデオ信号の圧縮の程度(比)及び精度を制限する。

ブロックによるビデオ圧縮法においては、いくつかの理由で、大きな変換ブロック誤差Eを生ずる。ブロックによる動き推定は、ただ連続する画像フレーム間の線形移動の動きを表すにすぎない。対応する変換ブロック T_N 及び T_{N+1} 間の表現可能な唯一の変化は、それら変換ブロックの相対的位置の変化だけである。このような表現法の欠点は、フルモーションのビデオ配列が、線形移動以外の複雑な動き、例えば、回転、拡大、および、斜行をしばしば含むことである。このような複雑な動きを単純な線形移動の近似で表現することはかなりの誤差を生むことになる。

ビデオ表示の別の態様として、このビデオ表示が、典型的には、互いに相対的に変化、即ち動く複数の画像特徴または画像オブジェクト（以下単にオブジェクトと呼称する）を含むことがある。オブジェクトは、あるビデオ表示内における個別の人物、事物、または、情景とすることができる。映画における情景に関しては、例えば、その情景の中の人物（すなわち、俳優）や事物（すなわち、点景）の各々が、個別のオブジェクトとなり得る。あるビデオ配列におけるオブジェクト間の相対的な動きが、従来のビデオ圧縮方法における、無視できない別の変換ブロック誤差要因となる。変換ブロックが規則的な形・大きさを持つために、その多くは、種々のオブジェクトの部分を含む。連続する画像フレーム中におけるオブジェクト間の相対的な動きが、対応する変換ブロック間に極端に低い相関を生じさせることがある（すなわち、変換誤差Eが高くなる）。同様に、連続する画像フレームにおけるオブジェクトの部分の出現（例えば、人物が振り向く）も高い変換誤差Eを招く。

従来のビデオ圧縮法は、変換誤差Eの大きさによって、その本来的な性質によって制限されているようである。デジタル・ビデオ表示能力に対する要求が高まるにつれて、デジタル・ビデオ圧縮の改良法が望まれている。

発明の概要

本発明は、多数の画像フレームから成るビデオ配列における表示の動きを表わすデジタル化ビデオ信号を圧縮するビデオ圧縮符号化方法を含む。この符号化方法は、オブジェクトベースのビデオ圧縮を利用し、それによって、フレーム内の動き、および、フレーム間の画像特徴の精度、および、融通性を改善するものである。ビデオ情報は、従来のビデオ圧縮法に見られるような固定の規則的な画素配列に対してではなく、任意の形を持つオブジェクトに対して圧縮される。これによって、誤差成分を低下させ、それによって、圧縮効率、および、精度を改善するようにする。別の利点は、本発明のオブジェクトベースの

10

20

30

40

50

ビデオ圧縮法が、圧縮したビデオ情報を処理するインタラクティブビデオ編集能力をもた
らすことである。

一つの好ましい実施態様においては、本発明の方法は、第1ビデオ画像フレームにおける任意の形をした画像特徴を特定し、その画像特徴の内部に、複数の異なる特徴点を定義することである。第1ビデオ画像フレームにおける画像特徴の特徴点は、直後の第2ビデオ画像フレームにおける画像特徴の対応する特徴点と相関する。これによって、第2のビデオ画像フレームにおける画像特徴の推定を実行する。第2のビデオ画像フレームにおける推定した画像特徴と、実際の画像特徴との差を求め、これを圧縮形式に符号化する。

本発明の符号化方法は、従来のブロックベースのビデオ圧縮法の欠点を克服するものである。この符号化方法はできれば多次元変換法を用いることが好ましい。すなわち、それによって連続する画像フレームにおける対応するオブジェクト間のマッピングを表わすためである。変換の多次元とは、一般的な形の座標の数を指す。多次元変換は、線形移動、回転、拡大、および、斜行のいずれかを、または、それらのすべてを含む複雑な動きを表わすことができる。そのため、連続する画像フレーム間のオブジェクトの複雑な動きを、比較的変換誤差を低く抑えて、表現することができる。

従来のブロックベースのビデオ圧縮法のもう一つの誤差要因は、一つの変換ブロック内部に含まれる複数のオブジェクト間の動きにある。本発明のオブジェクトベースのビデオ圧縮または符号化においては、変換ブロック内部における複数のオブジェクト間の相対的な動きを実質的に取り除く。そのため、オブジェクト間の動きによって生ずる変換誤差も実質的に減少する。本発明の符号化方法による変換誤差はきわめて低いために、本方法では、従来の符号化方法、例えば、MPEG-2で得られるものよりも、最大300%に達する大きな圧縮率を得ることができる。

本発明の好ましい実施態様の前記した利点及び他の利点は、下記の詳細な説明によりさらに明瞭になるであろう。添付図面を参照して説明する。

実施例の説明

第1図を参照すると、本発明の好ましい実施態様における動作環境は、汎用ないし特殊型いずれかのコンピュータ装置20であって、少なくとも1個の高速演算処理装置(Central processing unit, CPU)22を、メモリ装置24、入力装置26、および、出力装置28と結合状態において含むものである。これらの要素は、バス構造30によって相互に結合される。

図示のCPU22は、普及型デザインのもので、計算を実行するためのALU32、データの一次記憶のためのレジスタ集合34、装置20の動作を制御するための制御装置36を含む。CPU22は、以下の各種アーキテクチャーのいずれかを持つプロセッサであってもよい。すなわち、デジタル社のアルファ、MIPSテクノロジー社、NEC、ジューメンズおよびその他のMIPS、インテル社および、サイリックス、AMD、および、ネクスジェンを含むその他のx86、および、IBMおよびモトローラ社のパワーPcである。

メモリ装置24は、メイン・メモリ38、および、二次メモリ40を含む。図示のメイン・メモリ38は、16メガバイトの半導体RAMメモリの形態を取る。二次メモリ40は、ROM、光学ディスクまたは磁気ディスク、フラッシュ・メモリ、または、テープのような長期メモリの形態を取る。当業者であれば、メモリ装置24は、他にも、たくさんの別様の要素を含むことができることを了解するであろう。

入力装置、および、出力装置26、28もよく知られたものである。入力装置26は、キーボード、マウス、物理的トランスデューサー(例えば、マイクロフォン)等を含んでいてよい。出力装置28は、ディスプレイ、プリンター、トランスデューサー(例えば、スピーカー)等を含んでいてよい。ネットワーク・インターフェイスまたはモデムのようないくつかの装置も、入力、および、または、出力装置として使用することができる。

当業者であれば熟知のように、コンピュータ装置20はさらにオペレーティング・システム、および、少なくとも1個のアプリケーション・プログラムを含む。オペレーティング・システムはコンピュータ装置の動作とリソースの割り当てを制御する、一組のソフトウェアである。アプリケーション・プログラムは、オペレーティング・システムを通じて利用可能になるコンピュータのリソースを利用しながら、ユーザーの望むタスクを実行する一

10

20

30

40

50

組のソフトウェアである。両方とも図示のメモリ装置24中に滞在する。

コンピュータ・プログラミングの分野における当業者の慣用に従って、以下に本発明を、別様に断らないかぎり、コンピュータ装置20によって実行される動作に関する記号的表記法に基づいて説明する。上記動作は、時に、コンピュータ実行性と呼ばれることがある。記号的に表記される動作は、CPU22による、データ・ビットを表わす電気信号の操作、および、メモリ装置24におけるメモリ位置におけるデータ・ビットの維持、その他の信号処理を含む。データ・ビットが維持されるメモリ位置は、そのデータ・ビットに対する特定の電氣的、磁氣的、ないし、光学的性質を持つ物理的位置である。

第2Aおよび2B図は、ビデオ・ディスプレイ装置52(例えば、テレビ、または、コンピュータのモニタ)のディスプレイ画面50の模式図であって、対応するビデオ信号によって電子的に表わされるビデオ画像配列の内の、2個の連続する画像フレーム54a、および、54bを示す。ビデオ信号は、アナログ・ビデオ形式、例えば、NTSC、PAL、および、SECAMを含むビデオ信号形式、および、コンピュータ・ディスプレイで通常用いられる、画素式、または、デジタル化ビデオ信号形式、例えば、VGA、CGA、および、EGAのビデオ信号形式など、各種ビデオ信号形式の内のいずれで表わされたものであってもよい。できれば、画像フレームに対応するビデオ信号は、デジタル化信号形式のものであることが好ましい。すなわち、当業者であれば熟知のように、もともとそのようなものとして生成されたものか、アナログ・ビデオ信号形式から変換されたものかのいずれかであることが好ましい。画像フレーム54aおよび54bは、それぞれ、背景60上に位置する、直方形の固相画像特徴56と、ピラミッド型の画像特徴58とを含む。画像フレーム54aと54bにおける画像特徴56と58は、種々の部分が隠れて示されているように異なる外見を持つ。下記の説明のために、ある画像フレームの中のある画像特徴の特定の形をオブジェクト、または、マスクと言うことにする。従って、直方体固相特徴56は、それぞれの画像フレーム54aと54bにおいて、直方体固相オブジェクト56aおよび56bとして示され、ピラミッド型画像特徴58は、それぞれの画像フレーム54aと54bにおいて、ピラミッド型オブジェクト58aおよび58bとして示される。

ピラミッド型画像特徴58は、画像フレーム54aと54bとにおいて、同じ位置と方向で示されるので、ビデオ配列の中で示された場合「外見的に」不動に見えるものである。直方形固形体56は、フレーム54aと54bにおいて、ピラミッド58に対して、相対的に異なる方向、および、位置に示されるので、ビデオ配列の中で示された場合「外見的に」動き、回転しているように見えるものである。画像特徴58と60のこれらの見かけの動きは、図示的なもので、誇張されている。ビデオ配列の画像フレームは通常、30~60Hzの範囲にある頻度で表示される。ビデオの動きに対するヒトの感覚は、通常、2個以上のフレームを必要とする。画像フレーム54aと54bとはしたがって、本発明を例示するために、従来からのビデオ配列の模式図を表わす。さらに、本発明は決して単純なビデオ画像、画像特徴、または、配列のみに限定されないこと、逆に、任意の複雑さを持つビデオ画像やビデオ配列に適用可能であることが了解されるであろう。

ビデオ圧縮符号化工程概論

第3A図は、ビデオ圧縮符号化工程64の全体機能ブロック図であり、この工程は、複数の画像フレームを持つビデオ配列において、表示の動きを表わすデジタル化ビデオ信号を圧縮するためのものである。ビデオ情報(すなわち、ビデオ配列や信号)の圧縮は、例えば、インタラクティブないしデジタル・テレビや、マルチメディア・コンピュータ・アプリケーションを含む各種用途におけるデジタル・ビデオ情報の経済的なメモリや転送を可能にする。簡単のために、符号化工程64の機能ブロックに割り当てた参照数字は、その機能ブロックによって生成される結果に関しても互換的に用いることとする。

従来のビデオ圧縮法は、時間的相関、または、フレーム間相関と呼ばれる、連続する画像フレーム間の類似性を利用し、それによって、画像フレームの画素ベースの表現値を、動き表現値に変換するフレーム間圧縮を実行する。さらに、従来のビデオ圧縮法は、空間的相関、または、フレーム内相関と呼ばれる、画像フレーム内部の類似性を利用し、それによって、ある画像フレーム内の動きの表示をさらに圧縮するフレーム内圧縮を実行する。

このような従来のビデオ圧縮法においては、この中には、MPEG-1、MPEG-2、および、H.26Xが含まれるが、時間的・空間的相関は、固定した、規則的な（例えば、方形）画素配列の単純な線形移動に対して決定される。しかしながら、ビデオ情報は通常、画素の正方形配列を線形移動することによって正確に表わすことのできない任意のビデオの動きを含む。そのため、従来のビデオ圧縮法は通常、無視できない誤差成分を含むこととなり、これが、圧縮比と精度に制限を与える。

逆に、符号化工程64は、オブジェクトベースのビデオ圧縮を利用し、それによって、フレーム間の動き、および、フレーム内画像特徴を符号化する際の精度と融通性とを改善する。符号化工程64は、固定の規則的な画素配列ではなく、任意の形態を持つオブジェクトに対してビデオ情報を圧縮する。これによって、誤差成分は減少するから、圧縮効率と精度は改善される。もうひとつの利点として、オブジェクトベースのビデオ圧縮は、圧縮ビデオ情報を処理する際に有効な、インタラクティブビデオ編集能力をもたらす。

第3A図を参照すると、機能ブロック66は、あるビデオ配列の画像フレームにおけるユーザー定義によるオブジェクトを、同じ画像フレーム内の他のオブジェクトからセグメント化することを示す。オブジェクトは、任意の形態を持っていてよいが、できれば、ディスプレイ画像において、個別の画像特徴を表わしていることが好ましい。セグメント化とは、その画像フレームにおける、オブジェクトに対応する画素を特定することを含む。ユーザー定義によるオブジェクトは、ビデオ配列の中の画像フレームのそれぞれにおいて定義される。例えば、第2A、2B図において、直方体オブジェクト56aと56b、および、ピラミッド型オブジェクト58aと58bとは別々にセグメント化される。

セグメントされたオブジェクトは、そのオブジェクトの、2値又は多値ビット（例えば、8ビット）「アルファ・チャンネル」マスクで表わされる。オブジェクトマスクは、あるオブジェクトの、大きさ、形、位置を個別の画素として表わすものである。簡単のために、以下の説明では、2値マスクを取扱うこととする。2値マスクにおいては、オブジェクトの各画素は、通常の24ビット（3個の色成分値のそれぞれに対して8ビットずつ）ではなく、ただ一つの2値ビットで表わされる。多値ビット（例えば8ビット）マスクも従来使用されている。

機能ブロック68は、各オブジェクトの「特徴点」が、ユーザーによって定義されることを示す。特徴点はできれば、オブジェクトの際立った特徴又は態様であることが好ましい。例えば、隅角70a~70c、および、隅角72a~72cを、それぞれ直方体56とピラミッド58の特徴点としてユーザーが定義することができる。各画像フレームにおける、各オブジェクトマスクと、その特徴点に対応する画素は、メモリ装置24に含まれるオブジェクトデータベースに保存される。

機能ブロック74は、連続する画像フレームにおける特徴点の位置の変化を特定し、連続する画像フレーム間の特徴点に対して軌跡が求められることを示す。軌跡とは、特徴点の移動の方向及び程度を表わす。機能ブロック76は、直前フレームN-1と現在フレームNとの間におけるオブジェクトの特徴点の軌跡も、オブジェクトデータベースから取り出されることを示す。

機能ブロック78は、直前フレームN-1と現在フレームNの間において、オブジェクトに対して、粗動き変換を求めることを示す。粗動き変換は、フレームN-1とN間における特徴点軌跡に基づいて実行される。粗動き変換は、直前フレームN1と現在フレームNの間におけるオブジェクトの変化の近似を提供する。

機能ブロック80は、現在フレームNにおけるオブジェクトのマスクが、メモリ装置24のオブジェクトデータベースから取り出されることを示す。

機能ブロック90は、量子化マスターオブジェクト（即ち、「スプライト」）が、画像フレーム配列における画像特徴に対応するオブジェクト（即ち、マスク）66と、特徴点軌跡74から形成されることを示す。このマスターオブジェクトは、できれば、複数フレームに表われることになる、オブジェクトの態様又は特徴のすべてを含んでいることが好ましい。例えば、第2A、2B図に関して言えば、フレーム54bの直方体56は、フレーム54aに示されていない側面78bを含む。同様に、直方体56は、フレーム54aにおいて、フレーム54bには示

10

20

30

40

50

されていない側面78aを含む。直方体56のマスターオブジェクトは、両側面78aと78bとを含む。

粗動き変換78は、フレームN-1とN間におけるオブジェクトの変化を完全に表わさないことがよくある。例えば、直前フレームN-1におけるオブジェクト、例えば、直方体54aは、現在フレームNにおいてオブジェクトのすべての特徴を含んでいないかもしれない。例えば、直方体54bの側面78bのように。

したがって、変換の精度を上げるために、直前フレームN-1と現在フレームNにおけるオブジェクトのマスクの交点を、当業者には既知の論理AND機能を用いて求める。現在フレームNにおけるオブジェクトのマスクを、得られた交点から差し引き、直前フレームN-1のオブジェクトに含まれない、現在フレームNのオブジェクトの部分又は特徴(例えば、前述の直方体54bの側面78b)を特定する。オブジェクトの新たに特定された部分を、マスターオブジェクト90に取り込み、それによって、マスターオブジェクトが、フレームN-1とNにおいて、オブジェクトの完全な表現値を含めることができるようにする。

機能ブロック96は、直前フレームN-1におけるオブジェクト98の量子化形態(例えば、画像フレーム54aにおける直方体56a)が、密な動き変換によって変換され、それによって、現在フレームNにおけるオブジェクト102の予測形(例えば、画像フレーム54bにおける直方体56b)を与えることを示す。この変換は、オブジェクトベースのフレーム間圧縮を与える。

密な動き変換は、できれば、フレームN-1の量子化した直前フレームのオブジェクト98と現在フレームNのオブジェクトとの間のアフィン変換を求め、かつ、そのアフィン変換を、量子化した直前フレームのオブジェクト98に適用することから成ることが好ましい。好ましいアフィン変換は、アフィン変換係数104によって表わされ、かつ、線形移動、回転、拡大、および、斜行を記述することができる。アフィン変換は、密な動き推定から求められるが、できれば、直前量子化オブジェクト98と、現在フレームNのオブジェクトとの間において、画素単位のマッピングを含むことが好ましい。

予測した現在オブジェクト102は、密な動き変換96によって修正された量子化した直前フレームのオブジェクト98によって表わされるが、これは、比較的複雑な動きを表わすことができるだけでなく、マスターオブジェクト90から得られるどのような新しい画像局面をも表わすことができる。このようなオブジェクトベースの表現は、比較的精度が高い。というのは、オブジェクトに関連付けられる知覚的・空間的な連続性が、異なる画像フレームにおける異なるオブジェクト間に通常見られる関係変化から生ずる誤差を除去するからである。さらに、オブジェクトベースの表現の場合、ユーザーは、異なるオブジェクトを、異なる解像度レベルで表わすことが可能であるから、それによって、様々の複雑さを持つ、複数のオブジェクトを表わす際に関わる効率と精度を最適化することができる。

機能ブロック106は、画像フレームNに関して、予測した現在オブジェクト102を、現在フレームNのもともとのオブジェクト108から差し引き、それによって、予測オブジェクト102における推定誤差110を求めることを示す。推定誤差110は、量子化した直前フレームのオブジェクト98に対する、画像フレームNにおける現在オブジェクト108の圧縮表現である。もっと具体的に言うと、現在オブジェクト108は、推定誤差110と量子化した直前フレームのオブジェクト98とから復号化、すなわち、再構成することができる。

機能ブロック112は、推定誤差110が、従来の「損失性」(ロス - 損失を伴う)静止画像圧縮法、例えば、格子によるサブバンド圧縮や、その他のウェーブレット圧縮法、または、符号化法によって、または、離散コサイン変換(DCT)符号化法によって圧縮、すなわち、符号化されることを示す。これら圧縮法については、前者は、バイジャナタン(Vaidyanathan)著「マルチレート装置、および、フィルタ・バンク(Multirate Systems and Filter Banks)」(PTR Prentice-Hall, Inc., Englewood Cliffs, ニュージャージー州、アメリカ、1993)に、後者は、ペンベーカーら(Pennebaker et al.)著「JPEG - 静止画像圧縮基準」(Van Nostrand Reinhold, ニューヨーク、1993)に記載の通りである。

当業者には既知のように、「損失性」圧縮法は、データ圧縮を増そうとすると、データ歪を生ずる。データ歪とは、圧縮前の元データと、圧縮し、展開した後に得られるデータと

10

20

30

40

50

の間の変異を指す。下記の例示のために、機能ブロック102の圧縮、すなわち、符号化は、ウェーブレット符号化法であると想定する。

機能ブロック114は、機能ブロック112によって得られるウェーブレット符号化の推定誤差が、従来の「ロスレス」静止画像圧縮法によってさらに圧縮、即ち「符号化」され、それによって圧縮データ116を形成することを示す。好ましい通例の「ロスレス」静止画像圧縮法としては、ペンベーカーら(Pennebaker et al.)著「JPEG - 静止画像圧縮基準」に記載のエントロピー符号化法がよい。当業者には既知のように、「ロスレス」圧縮法はデータ歪を誘発しない。

誤差フィードバック・ループ118は、フレームNのオブジェクトについて機能ブロック112で得られたウェーブレット符号化推定誤差を利用して、直後フレームN+1に対する直前量子化オブジェクトを得る。フィードバック・ループの第1手順として、機能ブロック120は、機能ブロック112で得られたウェーブレット符号化推定誤差が逆ウェーブレット符号化、即ちウェーブレット復号化され、それによって画像フレームNのオブジェクトにたいする量子化誤差122を形成することを示す。

10

「損失性」静止画像圧縮法によって、推定誤差110を連続的に符号化し、復号化することの効果は、量子化誤差122から、一般に視認者には知覚されないビデオ情報を取り除くことにある。このため、そのような高い周波数成分を取り除くことによって、画像品質をほとんど損なわずに、最大約200%の画像圧縮率を得ることができる。

機能ブロック124は、画像フレームNに対して、量子化誤差122と予測オブジェクト102の双方が加算され、それによって、画像フレームNに対する量子化オブジェクト126を形成することを示す。タイミング調整遅延128の後、量子化オブジェクト126は、量子化した直前フレームのオブジェクト98となり、画像フレームN+1の対応するオブジェクトを処理する基礎として用いられる。

20

符号化工程64は、連続する画像フレームにおける対応するオブジェクトの時間的相関を用いて、改善されたフレーム間圧縮を獲得し、さらに、オブジェクト内の空間的相関を用いて、高精度の、効率的なフレーム内圧縮を獲得する。フレーム間圧縮に関して言えば、動き推定と補償を実行し、それによって、あるフレームにおいて定義されたオブジェクトを、直後のフレームにおいて推定する。直後フレームにおけるオブジェクトの動きベースの推定は、従来のブロックベースのオブジェクト表現よりもはるかに少ない情報しか必要としない。フレーム内圧縮に関して言えば、各オブジェクトに対する推定誤差信号を圧縮して、それによって、あるフレーム内の、そのオブジェクトの空間的相関を利用することができ、また、異なるオブジェクトを、異なる解像度で表わすことができる。フィードバック・ループ118によって、連続するフレームにおけるオブジェクトを、完全に展開されたオブジェクトから予測することができるから、そのため推定誤差の累積を防止することができる。

30

符号化工程64は、出力として、複数画像フレームのビデオ配列における表示の動きを表わす量子化ビデオ信号の圧縮表現、即ち符号化表現を提供する。この圧縮表現、即ち符号化表現は、オブジェクトマスク66、特徴点68、アフィン変換係数104、および、圧縮誤差データ116を含む。この符号化表現は、ビデオ情報の使用される特定の用途に従って、保存されてもよいし、転送されてもよい。

40

第3B図は、マスターオブジェクト90を符号化、即ち圧縮するためのマスターオブジェクト符号化工程130の機能ブロック図である。機能ブロック132は、マスターオブジェクト90が、従来の「損失性」静止画像圧縮法、例えば、格子によるサブバンド圧縮法ないし他のウェーブレット圧縮法、または、離散コサイン変換(DCT)符号化法によって圧縮、または、符号化されることを示す。できれば、機能ブロック132は、ウェーブレット符号化法を採用することが好ましい。

機能ブロック134は、機能ブロック132によって得られたウェーブレット符号化のマスターオブジェクトがさらに従来の「ロスレス」静止画像圧縮法によって圧縮、即ち符号化され、圧縮マスターオブジェクトデータ136を形成することを示す。好ましい通例のロスレス静止画像圧縮法はエントロピー符号化法である。

50

符号化工程130は、出力として、圧縮マスターオブジェクト136を提供する。符号化工程64によって与えられる圧縮表現、即ち符号化表現と共に、圧縮マスターオブジェクト136は、保存ないし転送後に展開、即ち復号化し、複数画像フレームのビデオ配列を生成させてもよい。

符号化工程64を、ある画像フレーム内部における単一オブジェクトに対応するビデオ情報を符号化することに関連して説明する。第2A図、2B図に示したように、また、上述のように、符号化工程64は、ある画像フレームにおける個々のオブジェクトごとに（例えば、第2A図、2B図のオブジェクト56と58）別々に実行される。さらに、多くのビデオ画像は、その上に任意の数の画像特徴又はオブジェクトをレンダリングする背景を含む。できれば、背景は、すべてのユーザー指定によるオブジェクトを処理した後、本発明によるオブジェクトとして処理されることが好ましい。

10

画像フレーム内のオブジェクトの処理のためには、それらオブジェクトが個別に特定されることが必要である。できれば、符号化工程64は、ある画像フレームについて、最前面のオブジェクト（単数または複数）から始まって、順々に進行していき最後尾のオブジェクト（例えば、背景）に至るオブジェクトに適用することが好ましい。符号化オブジェクトのビデオ画像への構成は、できれば、最後尾のオブジェクト（例えば、背景）から始まって、順々に最前面のオブジェクト（例えば、第2A、2B図の直方体56）に向かって進行することが好ましい。符号化オブジェクトのレイヤ化は、ある画像フレームのオブジェクトと関連する個別のレイヤ化データとして通信してもよいし、別法として、レイヤ化又は圧縮化シーケンスに対応する、あるシーケンスの中の符号化オブジェクトを転送、または、獲得することによって通信してもよい。

20

オブジェクトセグメントと追跡

一つの好ましい実施態様においては、機能ブロック66において参照される画像フレーム内の画像のオブジェクトをセグメント化することによって、ユーザーはインタラクティブなセグメント化を得ることができる。本発明のオブジェクトのセグメントは、オブジェクトをセグメント化するに当たって高精度を可能にし、比較的速く、かつ、セグメントされるオブジェクトを定義するにあたって、ユーザーに最適の柔軟性を提供する。

第4図は、オブジェクトのセグメント工程140の機能ブロック図であり、この工程によって、あるビデオ配列のある画像フレームの選ばれたオブジェクトをセグメント化する。工程140によるオブジェクトのセグメントは、高精度で速く、ユーザーにとって定義の簡単な、オブジェクトの感覚的グループ化を実現させる。

30

第5A図は、画像フレーム54aと直方体オブジェクト56aを示すビデオ・ディスプレイ装置52のディスプレイ画面50の模式図である。ディスプレイ画面50へのレンダリング時、直方体オブジェクト56aはオブジェクト辺縁（perimeter）142（図では、はっきりと分かるようにオブジェクト56aから離して描かれる）を含む。このものは、オブジェクト内部144の境界となる。オブジェクト内部144とは、ディスプレイ画面50におけるオブジェクト56aの輪郭を指し、一般に、内部表面、または、図示のように、画像特徴の外表面に一致する。下記の説明は、特に、直方体オブジェクト56aに関してなされるものであるが、ある画像フレームにおいてセグメントされる各オブジェクトに同様に適用可能である。

機能ブロック146は、ユーザーが、オブジェクト内部144内に、オブジェクト辺縁142の内部輪郭148を形成することを示す。ユーザーはできれば内部輪郭148を、マウスや、追跡ボールのような通例のポインター、または、カーソル制御装置で形成することが好ましい。内部輪郭148は、オブジェクト辺縁142から指定の距離150以内に形成される。指定距離150は、ユーザーによって選ばれるが、その大きさは十分大きく、ユーザーは、内部輪郭148を、辺縁142の指定距離150以内に、比較的速やかに形成することができる。指定距離150は、例えば、約4～10画素に相当する。

40

機能ブロック146は、ビデオ配列の中のキー・フレームに関連して実行される。例えば、従来の活動映画の場面に関して言えば、キー・フレームとは、ある場面の複数のフレームのうちの第1フレームである可能性が多い。この機能にユーザーを参加させることによって、オブジェクトセグメント工程140は半ば自動化されるが、それでいて、オブジェクト

50

のセグメントされる精度と柔軟性は目立って増す。キー・フレームを除いては、その後の画像フレームにおけるオブジェクトは、以下にさらに詳細に述べるように自動的にセグメント化される。

機能ブロック152は、内部輪郭148が自動的に拡大され、外部輪郭156を形成することを示す。外部輪郭156の形成は、外部輪郭156が、内部輪郭148からユーザー定義数の画素だけ増えるように、輪郭148の比較的簡単な画像拡大として実行される。できれば、内部輪郭148と外部輪郭156との間の距離はおおよそ距離150の2倍となることが好ましい。

機能ブロック158は、内部輪郭148と外部輪郭156の間の画素が、それらがオブジェクト内部144の中にあるかどうかに関する、あらかじめ定義された属性に従って分類され、それによって、第3A図に関連して記載された型のオブジェクト辺縁142と、対応マスク80を自動的に特定することを示す。できれば、画像属性は、画素・カラーと位置を含むことが好ましく、かつ、いずれの属性も、それだけで、または、他の属性と併せて使用できるものであることが好ましい。

この好ましい実施態様では、内部輪郭148と外部輪郭156における画素の各々が、(r、g、b、x、y)の形の5次元ベクトルで表わされる「クラスター中心」を定義する。用語r、g、およびbは、画素の各々に関連する赤、緑、および、青のカラー成分に対応し、用語x、および、yは、画素の位置に対応する。内部輪郭148の中の画素に対応するm個のクラスター中心ベクトルは、 $\{l_0, l_1, \dots, l_{m-1}\}$ で表わされ、外部輪郭156の中のn個のクラスター中心ベクトルは、 $\{o_0, o_1, \dots, o_{n-1}\}$ で表わされる。

クラスター中心ベクトル l_i と o_j の間の画素は、5次元ベクトル空間において、各画素のもっとも近いベクトルを特定することによって分類される。各画素について、それぞれのクラスター中心ベクトル l_i および o_j の各々にたいする絶対距離 d_i および d_j は、下記の式によって計算される。

$$d_i = w_{\text{color}}(|r-r_i| + |g-g_i| + |b-b_i|) + w_{\text{coord}}(|x-x_i| + |y-y_i|), \quad 0 \leq i < m,$$

$$d_j = w_{\text{color}}(|r-r_j| + |g-g_j| + |b-b_j|) + w_{\text{coord}}(|x-x_j| + |y-y_j|), \quad 0 \leq j < n,$$

ここに、 w_{color} 、および、 w_{coord} は、カラー、および、画素位置情報それぞれに対する重みづけ係数である。重みづけ係数 w_{color} 、および、 w_{coord} は、合計が1、そうでなければユーザーの選んだ数となる値を持つ。できれば、重みづけ係数 w_{color} 、および、 w_{coord} は0.5という等値を持つことが好ましい。各画素は、クラスター中心ベクトル l_i および o_j の一つにたいし5次元最小距離によるオブジェクト内部144または外部に関連する。

機能ブロック162は、ユーザーが、先頭フレームまたはキー・フレームの各オブジェクトにおいて、少なくとも2個の、できればそれよりたくさんの（例えば4から6個の）特徴点を選択することを示す。できれば、特徴点は、オブジェクトの比較的確立した局面であることが好ましい。例えば、直方体画像特徴56に関連して言えば、隅角70a~70cを特徴点として選ぶことも可能である。

機能ブロック164は、各選ばれた特徴点（例えば、隅角70a~70c）を取り囲む複数の画素から成るブロック166が定義され、後続画像フレーム（例えば、直後画像フレーム）の対応するブロックにマッチングされることを示す。画素ブロック166は、ユーザー定義によるものではあるが、できれば、画像内部144内の画素だけを含む32x32画素配列を含むものであることが好ましい。機能ブロック158で決められることであるが、（例えば、隅角70bと70c）画素ブロック166の内、オブジェクト内部144の外にはみ出す画素168（斜線で示す）は除外される。画素ブロック166は、従来のブロックマッチング工程、または、下記にさらに詳細に述べるポリゴンマッチング工程によって特定される最小絶対誤差によって、直後画像フレームの対応する画素ブロックにマッチングされる。機能ブロック170は、あるオブジェクトの粗動き変換が、2個の連続する画像フレームにおける対応する特徴点から求められることを示す。機能ブロック172は、現在画像フレームのマスク80が粗動き変換によって変換され、それによって、直後画像フレームのマスク80に対する推定の得られることを示す。後続画像フレームの中に特定されない、現在フレーム中の特徴点はいかなるも

10

20

30

40

50

のも無視される。

機能ブロック174は、直後画像フレームにおけるマスク80に関して得られた推定がフレーム1個分遅らせられ、かつ、その直後サイクルに対して輪郭176として機能することを示す。同様に、機能ブロック178は、対応する特徴点もフレーム1個分遅らせられ、同じ直後のフレームに対して初期特徴点180として利用されることを示す。

ポリゴン・マッチング

ポリゴン・マッチング200の機能ブロック図であり、この方法は、連続する画像フレームにおける、各対応画素のペアについて動きベクトルを求めるものである。このような密な動きベクトル計算は、第3A図の密な動き変換96を求める基礎となる。

ポリゴン・マッチング工程200は、従来のブロック・マッチング工程同様、連続する画像フレーム間の広範な動きを求めることができる。しかも、従来のブロックマッチング工程と違って、ポリゴンマッチング工程200は、オブジェクト辺縁近傍、または、直上に位置する画素に対する高精度を維持するから、その生成する誤差は目立って少ない。ポリゴン法200の好ましい実施態様においては、計算効率が向上する。

ポリゴンブロック法200を、第7A、および、7B図を参照しながら説明する。この図は、2個の連続する画像フレーム202a、202bを示すディスプレイ画面504の模式図であり、それら画像フレームにおいて、画像特徴204が、それぞれ、オブジェクト204a、および、204bとしてレンダリングされる。

機能ブロック206は、画像フレーム202aと202bにおけるオブジェクト204aと204bとが特定され、例えば、オブジェクトセグメント法140によってセグメントされることを示す。

機能ブロック208は、オブジェクト204bに対して適用される画素ブロック210b（例えば、15x15画素）と、オブジェクト204aの周囲の探索領域212に対する次元数が求められることを示す。画素ブロック210bは、それに対して、オブジェクト204aにおける対応画素ブロック210aが特定される、そのような、オブジェクト204bの周囲の領域を定義する。探索領域212は、対応画素ブロック210aが求められる領域を定める。できれば、画素ブロック210bと探索領域212とは、直角の規則的な画素配列であって、ユーザーによって定義される大きさを持つものであることが好ましい。

機能ブロック214は、オブジェクト204b内の初期画素216が特定され、現在画素と指定されたことを示す。初期画素216は、各種の基準の内のいずれで定義されてもよい。そのような基準としては、例えば、垂直最大延長位置の画素、および、水平最小延長を持つ画素などがある。図示の座標軸220に従って配置された、ディスプレイ画面50上の画素に関して言えば、初期画素216は、最大y座標値と、最小x座標値を持つ、オブジェクト214bの画素として表わしてもよい。

機能ブロック222は、画素ブロック210bが、現在画素を中心とし、その周辺に広がっていることを示す。

機能ブロック224は、画素ブロック210bが、オブジェクト204bに含まれない画素（例えば、第7B図の斜線で示した画素226）を含むかどうかに関する調査を表わす。この調査は、機能ブロック206によって特定されたオブジェクトを参照して実行される。現在画素に位置づけられる画素ブロック210b内の画素がオブジェクト204b外にはみ出した場合は必ず、機能ブロック224は、機能ブロック228に進み、その他の場合は機能ブロック232に進む。

機能ブロック228は、画素ブロック210bのうち、オブジェクト204bの外にはみだす画素（例えば、画素226）は、画素ブロック210bによって定義される領域から除外し、それによって、画素ブロック210bが、オブジェクト204b内部の画素だけを含むようにする。その結果、画素ブロック210bは、もともと定義された正方形、ないし、長方形領域よりもさらに複雑なポリゴンとなる。

機能ブロック232は、オブジェクト204aの中のある画素が、オブジェクト204bにおける現在画素と対応すると特定されたことを示す。オブジェクト204aにおけるその画素を、直前対応画素と言う。できれば、直前対応画素は、探索領域212の各画素の周囲に画素ブロック210aを形成し、かつ、画素ブロック210aと、オブジェクト204bにおける現在画素の周囲の画素ブロック210bの間の相関を求めることによって特定することが好ましい。画素プロ

10

20

30

40

50

ック210aと210bの間の各相関は、例えば、絶対誤差によって求めてもよい。直前対応画素は、探索領域212の中に、画素ブロック210bにたいする絶対誤差が最小となる画素ブロック210aを特定することによって特定される。画素ブロック210bに対する画素ブロック210aの合計絶対誤差は、次のようにして求めることができる。

$$E = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} (|r_{ij} - r'_{ij}| + |g_{ij} - g'_{ij}| + |b_{ij} - b'_{ij}|),$$

ここに、項 r_{ij} 、 g_{ij} 、 b_{ij} は、画素ブロック210b中の画素の各々に関連付けられた、それぞれ、赤、緑、および、青のカラー成分に対応し、項 r'_{ij} 、 g'_{ij} 、 b'_{ij} は、画素ブロック210aの画素の各々に関連付けられた、それぞれ、赤、緑、および、青のカラー成分に対応する。

上に規定したように、絶対誤差Eの総計は、 $m \times n$ の画素次元を持つ画素配列を持つ画素ブロックを含む。ポリゴンを持つ画素ブロック210bは、ポリゴン画素ブロック210bの外にはみ出すすべての画素のカラー成分に対してゼロ値を定義することによって、比較的簡単に取り込むことができる。

機能ブロック234は、オブジェクト204b内の各画素と、オブジェクト204a内の対応直前画素間の動きベクトルMVが求められることを示す。動きベクトルは、オブジェクト204bの画素の位置と、オブジェクト204aの対応直前画素の位置の間の差と定義される。

$$MV = (x_i - x_k', y_j - y_l')$$

ここに項 x_i と y_j は、それぞれ、画素ブロック210bにおける画素の、 x -座標位置、および、 y -座標位置に一致し、項 x_k' と y_l' は、それぞれ、画素ブロック210aの対応直前画素の x -座標位置、 y -座標位置に一致する。

機能ブロック236は、オブジェクト204bが、何らかの残存画素を含むかどうかに関する調査を表わす。オブジェクト204bが残存画素を含む場合は必ず、機能ブロック236は、機能ブロック238に進み、そうでなければ終了のブロック240に進む。

機能ブロック238は、オブジェクト204bの次の画素が、あらかじめ指定された方式、または、配列によって特定されることを示す。初期画素は、機能ブロック214に関連して前述のように選ばれたが、後続画素は、ある列において（すなわち、共通の y -座標軸の）、まずすぐ次の画素を特定することによって定義し、かつ、もしもオブジェクト204が列の中に他に画素を含んでいなければ、1段下の列の、第1、すなわち、最左画素（すなわち、最小 x -座標値を持つ）に進んでもよい。このようにして特定された画素は、現在画素に指定され、機能ブロック238は、機能ブロック222に戻る。

ポリゴンブロック法200は、対応画素が、仮に、オブジェクトの辺縁上に、または、辺縁近傍に位置していても、対応画素を正確に特定する。従来のブロック・マッチング工程に見られたかなりの誤差の原因は、画素ブロック210bの内、オブジェクト204bの外にはみ出した画素を除外しないし無視することによって除去される。従来のブロック・マッチング工程は、均一なブロック形には厳密に適用されるが、セグメントオブジェクトに対しては厳密には適用されない。均一ブロック形態は、オブジェクトの辺縁近傍の画素に対しては重大な誤差を引き起こす。なぜなら、オブジェクトの外にはみ出す画素は、オブジェクトが動くにつれ、または、その背景が変化するにつれ、かなりの変化を受けるからである。従来のブロック・マッチング工程においてはそのようにはみ出し画素の変動が含まれるのであるから、オブジェクト辺縁近傍の画素は、厳密には、直前画像フレームの対応画素と相関されない。

オブジェクト204bの各画素に関しては、オブジェクト204aの対応直前画素が特定される。これは、直前オブジェクト204aの画素の各々について、画素ブロック210bと、画素ブロック210aとを比較することによって実行される。直前対応画素とは、オブジェクト204aにおいて、画素ブロック210bともっとも高い相関を持つ画素ブロック210Aを持つ画素である。従来のやり方で処理するならば、この解決には、各直前対応画素を特定するために実質的な計算が要求される。具体例を挙げるならば、 $m \times n$ 画素の次元を持つ探索領域212よりも

10

20

30

40

50

はるかに小さい $n \times n$ 画素の次元を持つ画素ブロックについて、直前オブジェクト204aにおける各直前対応画素を特定するために、約 $n^2 \times m^2$ 個の計算が必要である。

画素ブロック関連工程

第8図は、修正画素ブロック関連工程260の機能ブロック図であり、できれば、これを、機能ブロック232に関連して記載したものの代わりとして用いることが好ましい。修正関連工程260は、画素ブロック210bと210aを関連する際に内在的に認められる冗長度を利用し、それによって、必要計算数を際立って減少させる。

関連工程を、第9A~9G、および、10A~10Gを参照しながら説明する。これらの図は、連続する画像フレーム202aと202bに対応する任意の画素群を模式的に表わす。特に、第9A図は、 5×5 画素の次元を持つ画素ブロックの模式図であり、この図において、各文字は、個別の画素に相当する。画素ブロック262の画素は、個別の行264を含む、画素の直角規則配列として配置される。第9B図は、 $q \times q$ の次元を持ち、直前画像フレーム202aの探索領域212に対応する画素配列266を表わす。第9B図の各々の数字は個別の画素を表わす。従来の直角規則画素ブロック262と関連して説明するけれども、関連工程260は、ポリゴン・マッチング工程200に関連して説明した型のポリゴン画素ブロックにも同様に適用できる。

機能ブロック268は、初期画素ブロック（例えば、画素ブロック262）が、中央画素Mに関して定義され、探索領域212（例えば、画素配列266）を亘って走査されることを示す。この走査は、一般に、従来のブロック・マッチング工程に見られるように、ラスター・パターン（一部第7A図に示す）の形で実行される。第9C~9Gは、画素ブロック262と画素配列266の間のブロック・マッチング工程におけるおよそ q^2 個の手順の内の5個を模式的に図示する。

画素ブロック262の、画素配列266にたいする横断走査は、従来のやり方で行なわれるけれども、両者の間の関連に関する計算は、本発明にしたがって、別様のやり方で実行される。特に、関連（例えば、絶対誤差）は、各走査位置において、画素ブロック262の各行264に対して求められ、される。各走査位置において、画素ブロック262の各行264について求められ、保存された関連は、行相関270と呼ばれる。そのうちのいくつかを、関連された画素に関連させることによって、第9C~9G図に記号で示す。具体的に示すと、第9C図は、行相関270(1)を示すが、これは、画素配列266に揃えられた、画素ブロック262の単一行264に対して求められたものである。同様に、第9D図は、行相関270(2)と270(3)とを示し、これらは、画素配列266に揃えられた画素ブロックの2個の行264に対して求められたものである。第9E~9G図は、画素配列266にたいする、上記に続く3個の連続走査位置における、画素ブロック262との、同様の行相関を示す。

画素配列266上を、初期画素ブロック262を走査することによって、行相関のメモリ配列、または、データベースが得られる。画素ブロック262が、 r 数の行264を持ち、かつ、画素配列266が $q \times q$ 個の画素を持つ場合、行相関データベースは、約 rq^2 個の行相関を含む。この行相関の数はほんの近似値である。なぜなら画素ブロック262は、できれば最初に画素配列266の上を、画素Mが画素配列266の画素の内の第1列に揃うように、走査されることが好ましいからである。

第9C図に示されるものから始まる残余の手順は、画素配列266の上を画素ブロック262が完全に2回に亘って走査した後に行われる（すなわち、画素配列266の第1列、第2列に画素Mが揃った後）。

機能ブロック274は、次の画素ブロック（第10A図）が、例えば、画像フレーム202bにおいて、画素Mと同じ列の中心画素Nに関して定義されることを示す。画素ブロック276は、画素ブロック262に含まれない画素行278と、画素ブロック262に含まれる画素行280とを含む。画素ブロック276は、画素ブロック262に含まれる行282（第9A図）を含まない。次の画素ブロック276におけるこのような増加させる定義のやり方は、実質的に、従来のブロック・マッチング工程に用いられるものと同じである。

機能ブロック284は、画素ブロック276が、機能ブロック268に関連して前述したようなやり方で画素配列266の上を亘って走査されることを示す。第9C~9G図と同様に、第10B~10G図も、画素ブロック276が、画素配列266の上を走査される様を示す。

10

20

30

40

50

機能ブロック286は、行278について、各走査位置において、行相関が求められ、保存されることを示す。従って、行相関288(1)～288(5)は、それぞれ、第10B～10Fに示す行278の走査位置に関連して求められる。

機能ブロック290は、画素ブロック276の行280の各々について、メモリ行計算値が、以前に計算された各走査位置に関して取り出され、機能ブロック268に保存されることを示す。例えば、第9C図の行相関270(1)は、第10C図の270'(1)と同じである。同様に、第10D～10F図の行相関270'(2)、270'(3)、270'(5)～270'(8)および270'(15)～270'(18)は、第9D、9E、および、9Gの対応する行相関と同じである。従って、画素ブロック276について、ただ行相関288のみが各走査位置について計算される。この結果、画素ブロック276に必要な計算の数は、ほぼ80%減少される。

10

機能ブロック292は、後続画素ブロック294(第11A図)が、画素Mに関して次に続く列の中心画素Rに関して定義されることを示す。画素ブロック294は、第9A図の画素ブロック262の行264と類似ではあるが、はっきりと異なる画素から成る行296を含む。特に、行296は、行264には含まれない画素A'～E'を含む。後続画素ブロック294のこのような増加させる定義のやり方は、実質的に従来のブロック・マッチング工程に用いられるものと同じである。

機能ブロック298は、画素ブロック294が、機能ブロック268、および、276に関連して前述したやり方で、画素配列266(第9B図)の上を亘って走査される。第11B～11F図は、画素ブロック294を、画素配列266上を走査する様を表わす。

機能ブロック300は、行296の各々について、行相関が求められ、保存されることを示す。したがって、行相関302(1)～302(18)は、第11B～11Fに示した行296の走査位置に関連して求められる。

20

行相関302(1)～302(18)の各々は、画素ブロック262(第9A図)に関して実行された行相関に関連して、短縮したやり方に基づいて計算してもよい。

例えば、第11D図の行相関302(1)～302(18)は、第9E図の副行相関304(4)～304(8)と同じ副行相関304'(4)～304'(8)を含む。したがって、行相関302(4)～302(8)は、それぞれの行相関270(4)～270(8)から、画素01A、02B、03C、04D、および、05Eに対する後者の相関値を差し引き、それによって、それぞれ、副行相関304(4)～304(8)を形成してもよい。行相関302(4)～302(8)は画素のペア56A'、57B'、58C'、59D'、および、50E'に対する相関値を、それぞれの副行相関値304(4)～304(8)に加えて得てもよい。

30

それぞれの行相関270(4)～270(8)から、行相関302(4)～302(8)を求めるには、画素ブロック294には含まれない、画素ブロック262の画素A～E列に対応する、個々の画素相関値を差し引き、かつ、画素ブロック294には含まれるが、画素ブロック262には含まれない画素A'～E'列に対する画素相関値を足し加える必要がある。この方法は、行相関302(4)～302(8)の各々に対し、従来のやり方であれば、各行相関を求めるのに5回の加算が必要とされるのに対し、1回の引き算、1回の足し算で代用する。さらに大きな次元の画素ブロックが好まれば好まれる程、従来の計算法に優るこの方法の利点がますます際立つ。従来の画素マッチング工程は、画素配列266にたいする初期画素ブロック262の各走査位置について、総合ブロック相関のみを特定するだけである。そのため、全画素にたいする全相関値を、各走査位置について個別に計算しなければならない。逆に、相関工程260は、メモ

40

リ行相関270を利用し、それによって、必要な計算数を目立って減少させる。相関工程260によってもたらされる速度と、プロセッサのリソース要求度における改善は、行相関を保存するために必要とされる装置条件を補って余りある。

相関工程260は、第9～11図を参照しながら記載してきたが、これは、本発明の特徴を例示するためのものであることが了解されるであろう。例示したように、本発明は、コンピュータ装置で実行するのに特に相応しい繰り返し、または、周期的性質を含む。この繰り返し、または、周期的性質は、画素ブロックと画素配列に依存するものであるが、当業者であれば、十分に理解し、導入可能なものである。

多次元変換

第12図は、第1と第2の連続する画像フレーム間において多次元変換を生成し、かつ、転

50

送ないし保存のためにマッピングを量子化することから成る、変換法350の機能ブロック図である。この多次元変換は、できれば第3図の機能ブロック96と関連させて使用するのが好ましい。変換法350は、第7A図と第13図に基づいて説明される。これらの図の内の後者は、第7B図と同様、ディスプレイ画面50の模式図であって、画像特徴204がオブジェクト204bとしてレンダリングされる画像フレーム202bを示す。

変換法350は、できれば、線形移動、回転、拡大、および、斜行の内のいずれか、または、その全てを含む複雑な動きを表わすことのできる多次元アフィン変換を実行するものであることが好ましい。変換法350は、1次元のみで、かつ、線形移動にしか関わらない従来のビデオ圧縮法、例えば、MPEG-1、MPEG-2、および、H.26Xよりもはるかに優る。この点において、変換の次元性とは、後で詳しく述べるように、変換の一般形における座標の

10

数を指す。複雑な動きを、本発明に従って表現することによって精度は増すけれども、それによる誤差は、従来の表現法によるものよりも少ない。従って、圧縮効率が増す。機能ブロック352は、オブジェクト204aおよび204bの画素の密な動き推定を求めることを示す。できれば、密な動き推定は、ポリゴン・マッチング工程200で入手することが好ましい。前述したように、密な動き推定は、座標 (x_i, y_j) における、画像フレーム202bのオブジェクト204bの画素と、座標 (x_i', y_j') における、画像フレーム202aのオブジェクト204aの画素との間の動きベクトルを含む。

機能ブロック354は、変換ブロック356の配列が、オブジェクト204bを見込むように定義されることを示す。できれば、変換ブロック356は、例えば、 32×32 画素の次元を持つ、画素の直角規則配列であることが好ましい。

20

機能ブロック358は、各変換ブロック356に対して一つの多次元アフィン変換が生成されることを示す。できれば、アフィン変換は、1次であって、次のように表わされるものであることが好ましい。すなわち、

$$x_i' = ax_i + by_i + c$$

$$y_i' = dx_i + ey_i + f$$

かつ、動きベクトルがそれに対して比較的高い信頼度を持つすべての画素について求めることが好ましい。このアフィン変換は、 x_i と y_i とが、二つの座標 x_i と y_i に関して定義されるという点で2次元を持つ。

動きベクトルの相対的な信頼度とは、対応する画素間の動きベクトルが、他の画素に対して、一意的に決定される、その精度を指す。例えば、比較的大きな画素配列の中にあり、かつ、均一な色を持つ(例えば、黒)、特定の画素間の動きベクトルは、通常、高精度に求めることはできない。特に、第1画像フレームにおける黒画素について言えば、後続画像フレームの画素配列の中の多数の画素が同一相関を持つ(すなわち、画素ブロック間の絶対値誤差)。

30

逆に、構成画素が、個別的な特徴に対応する画素配列は、後続の画像フレームにおいて、特定の対応画素に対して比較的高い相関を持つ。

比較的高い相関は、特定の画素に対して、極小絶対誤差計算として表わされることが好ましい。従って、比較的信頼度の高い動きベクトルは、このように一意的に低い誤差値に対して求めることができる。例えば、信頼度の高い動きベクトルとは次のようなものとして定義される。すなわち、その動きベクトルに対する最小絶対値誤差が、その画素に関連する次に大きい誤差値よりも、閾値差分を上回る差分少ないもの、である。別法として、信頼度の高い動きベクトルは、相関係数を求める基礎となる絶対誤差値の2次微分に関して定義することもできる。ある特定の値を上回る2次微分は、特定の対応画素間に比較的高い相関のあることを示す。

40

このような高信頼度の動きベクトルを含む n 個の画素が与えられた時、好ましいアフィン変換方程式は、画像フレーム202aと202bにおける n 個の対応画素に関して求められる。画像フレームは、好ましいアフィン変換方程式の、6個の未知係数 a 、 b 、 c 、 d 、 e 、および、 f を求めるには、画像フレーム202aと202bに、高信頼度の動きベクトルを含む少なくとも3個の対応画素を含んでいなければならない。好ましい次元に関して言えば、変換ブロック356の各々が 2^{10} 個の画素を含み、その内の相当数が通常比較的高い信頼度の動きベク

50

トルを持っていないなければならない。これによって得られる n 個の方程式は、以下の直線代数式で表わされる。

$$\begin{bmatrix} X_0 & Y_0 & 1 \\ X_1 & Y_1 & 1 \\ X_1 & Y_1 & 1 \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ X_{n-1} & Y_{n-1} & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} X'_0 \\ X'_1 \\ \vdots \\ \vdots \\ X'_{n-1} \end{bmatrix}$$

10

$$\begin{bmatrix} X_0 & Y_0 & 1 \\ X_1 & Y_1 & 1 \\ X_1 & Y_1 & 1 \\ \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots \\ X_{n-1} & Y_{n-1} & 1 \end{bmatrix} \begin{bmatrix} d \\ e \\ f \end{bmatrix} = \begin{bmatrix} Y'_0 \\ Y'_1 \\ \vdots \\ \vdots \\ Y'_{n-1} \end{bmatrix}$$

20

できれば、これらの方程式は、従来の、特異数値分解法(Singular value decomposition, SVD)によって解くのが好ましい。この方法により、密な動きベクトルの近似値にたいする最小二乗誤差が得られる。従来のSVD法は、例えば、プレスら(Press et al.)著「Cにおける数字処理(Numerical Recipes in C)」Cambridge University Press (1992)に記載されている。

30

上述のように、この好ましい二次元アフィン変換方程式は、連続する画像フレーム202aと202b間において、変換ブロック356の線形移動、回転、拡大、および、斜行を表わすことができる。逆に、従来の圧縮標準として用いられる通例の動き変換法は、下記の形の、簡単な変換方程式を用いる。

$$x_i = x_i + g$$

$$y_i' = y_i + h$$

従来の単純化された変換方程式は、動きを、たった二つの係数 g と h で表わす。これは、好ましい多次元変換方程式によって得られる情報量(すなわち係数)の僅か3分の1にすぎない。従来の圧縮法と比べて、変換法350によって得られる情報の優れた圧縮効果を現実のものとするには、変換ブロック356の次元を、できれば、MPEG-1やMPEG-2圧縮法によって採用されている16x16の3倍を超えるものとすることが好ましい。変換ブロック356の好ましい32x32画素次元は、従来の変換法の変換ブロックで採用される画素数の4倍に渡る。変換ブロック356の従来に優る大きな次元数は、アフィン変換係数が変換ブロック356の動きを表わす際の高精度とあいまって、変換法350が、従来の圧縮法に優る大きな圧縮を実行するのを可能にする。

40

本発明によって生成されるアフィン係数は、通常、非整数の、浮動小数点値であって、十分に圧縮しようとするとき必ずその精度に悪影響を与えるものであることが了解されるであろう。従って、アフィン変換係数を保存ないし転送する際に要求される帯域を減らすためには、それら係数を量子化することが好ましい。

50

機能ブロック362は、機能ブロック358に関連して生成されたアフィン変換係数が量子化され、それによって、それら係数を保存ないし転送する際に必要とされる帯域を減少させることを示す。第14図は、変換ブロック356の断片的に拡大表現したものであって、3個の選ばれた画素364a、364b、および、364cを示す。すなわち、これらの画素から、6個の好ましいアフィン変換係数a~fが求められる。

画素364a~364cは、それぞれ、それぞれ (x_1, y_1) 、 (x_2, y_2) 、 (x_3, y_3) の画素座標として表わされる。機能ブロック352の密な動き推定に基づき、画素364a~364cは、直前画像フレーム202aにおいて、それぞれの対応画素 (x_1', y_1') 、 (x_2', y_2') 、 (x_3', y_3') を持つ。従来のやり方と同様に、画素位置 (x_i, y_i) は、整数で表わされ、アフィン変換方程式の解答となる。そして、それに基づき、好ましいアフィン変換係数が求められる。従って、選ばれた画素364a~364cは、直前画像フレーム202aの対応画素を計算するのに用いられるが、この画素は、通常、浮動小数点値となる。

これらの浮動小数点値の量子化は、対応する画素間の差 $(x_i - x_i', y_i - y_i')$ を、整数形に変換することによって実行する。アフィン変換係数は、まず、差ベクトルと画素値 (x_i, y_i) から画素値 (x_i', y_i') を計算し、次に、画素値 (x_i', y_i') に関して機能ブロック358の多次元変換方程式を解くことによって求められる。

第14図に示したように、画素364a~364cは、できれば、変換ブロック356内部の局所的変動にたいする量子化の感度を最小にするように、変換ブロック356内のあちこちに分布される。できれば、画素364aは変換ブロック356の中心か、中心近傍に、画素364bと364cとは上隅に位置するのが好ましい。また、好ましい実施態様においては、オブジェクト204bの変換ブロック356の各々における選ばれた画素が、同じ位置を持つことが好ましい。それによって量子化工程が効率的に実施できるからである。

機能ブロック362の量子化法のもう一つの態様は、異なるレベルの量子化を用いて、様々の動きの程度を表わすことができる、ということである。その結果、比較的単純な動き（例えば、線形移動）は、複雑な動きに要求されるものよりも少なく選ばれた画素を用いて表わすことができる。前述のアフィン変換方程式に関して言えば、オブジェクト204bからの画素364a (x_1, y_1) と、オブジェクト204a対応画素 (x_1', y_1') で、下記の形の単純化されたアフィン変換方程式を解くのに十分である。

$$x_1' = y_1 + c$$

$$y_1' = y_1 + f$$

これは、連続する画像フレーム間の線形移動を表わす。画素364aは、その中心位置が、一般に、他の種類の動きとは独立に線形移動を表わすという理由から、特異的に使用される。したがって、ユーザーは、線形移動のような単純な動きは、単純化されたアフィン変換方程式で選択的に表現することができる。単純化されたアフィン変換方程式は、複雑な動きを表現するのに必要とされるデータの3分の1しか要求しないからである。

同様に、オブジェクト204bに関する、一对の選択画素 (x_1, y_1) （例えば、画素364a）と (x_2, y_2) （すなわち、画素364bと364cのどちらか）、および、オブジェクト204aに関する対応画素 (x_1', y_1') および (x_2', y_2') は、下記の形の単純化されたアフィン変換方程式を解くのに十分である。

$$x_i' = ax_i + c$$

$$y_i' = ey_i + f$$

これは、連続する画像フレーム間の線形移動および拡大を含む動きを表わすことができる。下記の単純化された形において、

$$x' = a \cos x + \sin y + c$$

$$y' = -\sin x + a \cos y + f$$

選択画素の対応ペアは、線形移動、回転、等方拡大を含む動きを表わすことができる。この単純形においては、変数 x 、 y が共通の係数を持つので、この方程式は、2個の対応ペアによって解くことが可能となる。

従って、ユーザーは、線形移動、回転、および、拡大を含む比較的複雑な動きを、部分的に単純化されたアフィン変換方程式によって選択的に表わすことができる。このような方

程式は、複雑な動きを表わすのに必要とされるデータの3分の2しか要求しない。オブジェクト204bの、第3の選択画素(x_3, y_3)を加えると、オブジェクト204aの対応画素(x_3', y_3')と、完全な、好ましいアフィン変換方程式とによって、ユーザーは、さらに、連続する画像フレーム間の斜行をも表わすことができるようになる。

変換法350(第12図)の好ましい実施態様を、例えば、32x32画素の次元数を持つ、一樣変換ブロック356使用の場合を例にとって説明する。機能ブロック358に関連して説明した好ましい多次元アフィン変換は、変換ブロック356に関して求められる。変換ブロック356の次元は、本方法によって得られる圧縮比に直接影響することが了解されるであろう。

あるオブジェクトについて、画像フレーム間の変換を表わすためには、比較的大きな次元数を持つ変換ブロック356の場合、より少ない次元数を持つ変換ブロック356の持つ数よりも、数が少なくなる。一樣に大きな変換ブロック356を設けると、その結果は、それに相当する大きな誤差が、各変換ブロックあたりに導入されることになる。したがって、一樣な大きさの変換ブロック356は、通常、この相矛盾する性能制限のバランスを取るために、中等度の次元数を持つ。

変換ブロックの最適化

第15図は、変換ブロック最適化法370の機能ブロック図である。この方法は、最小誤差閾値を与える変換ブロック次元を自動的に選択するものである。最適化法370は、第16図に基づいて説明する。第16図は、オブジェクト204bを含む画像フレーム202bの一部を示すディスプレイ画面50の模式図である。

機能ブロック372は、初期変換ブロック374が、オブジェクト204bに関して定義されることを示す。初期変換ブロック374は、できれば、ユーザーの選択可能な最大次元数のもの、例えば、64x64画素であることが好ましい。初期変換ブロック374は、現在変換ブロックと呼ばれる。

機能ブロック376は、現在の、信号対雑音比(SNR)ピークが、現在の変換ブロックに関して計算されることを示す。この信号対雑音比は、できれば、現在変換ブロック内部の画素のカラー成分値の変動(すなわち、信号)の、推定誤差110(第3図)に関連する画素のカラー成分値の変動にたいする比として計算されるのが好ましい。

機能ブロック378は、現在変換ブロック(例えば変換ブロック374)が、例えば、4個の、相等的なサブブロック380a~380dに分割され、サブブロック380a~380dの各々についてアフィン変換が求められ、かつ、そのアフィン変換に関して将来の信号対雑音比が求められることを示す。将来の信号対雑音比は、機能ブロック376に関連して説明した現在の信号対雑音比について行なったものと実質的に同じやり方で求められる。

調査ブロック382は、将来信号対雑音比が、現在信号対雑音比よりも、ユーザーの選択した閾値量を越えて、上回るかどうかに関する調査を表わす。この調査は、現在変換ブロック(例えば、変換ブロック374)をさらに細分割したならば、それによって、アフィン変換の精度を、少なくとも閾値分だけ改善することができるという決定を表わす。将来信号対雑音比が、現在信号対雑音比よりも、閾値以上に上回る場合は必ず、調査ブロック382は、機能ブロック384に進むが、そうでなければ、機能ブロック388に進む。

機能ブロック384は、サブブロック380a~380dが、連続的に現在変換ブロックを指定され、かつ、各々が、さらに細分割されるべきか否かを調べられることを示す。具体例を挙げると、サブブロック380aが現在変換に指定され、機能ブロック376に従って処理され、さらに、サブブロック386a~386dに細分割される。機能ブロック388は、次に続く変換ブロック374'が特定され、かつ、初期、ないし、現在変換ブロックに指定されることを示す。

圧縮前外挿法

第17AおよびB図は、圧縮前外挿法400の機能ブロック図であり、これは、任意の形の画像特徴を、あらかじめ定義された形に外挿し、それによって、符号化工程64(第3両図)の機能ブロック112に従って圧縮する手順を容易にするものである。外挿法400によって、機能ブロック112の圧縮が、前述のDCT、格子による圧縮、その他のウェーブレット圧縮のような従来のやり方で実行することが可能となる。

10

20

30

40

50

格子による圧縮、その他のウェーブレット圧縮、または、離散コサイン変換(DCT)のような、従来の静止画像圧縮法は、画素の方形配列に基づいて操作される。しかしながら、前述のように、本発明の方法は、任意の形の画像特徴、即ちオブジェクトに適用可能である。そのようなオブジェクト、または、画像特徴を、方形画素配列形に外挿すれば、格子による圧縮、その他のウェーブレット圧縮、または、DCTのような従来の静止画像圧縮法の使用が可能となる。外挿法400は、第18A～18Dに基づいて説明される。これらの図は、単純なオブジェクト402がレンダリングされるディスプレイ画面50を表わすもので、外挿法400の様々な局面を示すものである。

機能ブロック404は、外挿ブロック境界406が、オブジェクト402の周囲に定義されることを示す。外挿ブロック境界406は、できれば、方形であることが好ましい。第18A図を参照しながら説明すると、オブジェクト402の周囲における外挿ブロック境界406の形成は、オブジェクト402の辺縁408の特定に基づいて実行され、その特定は、例えば、オブジェクトセグメント法140(第4図)によって実行される。外挿ブロック境界406は、説明のために、オブジェクト402をその全体に渡って取り囲むように図示される。外挿ブロック境界406は、また別法として、オブジェクト402の一部のみを取り込むことができることが了解されるであろう。オブジェクトセグメント法140に関連して前述したように、オブジェクト402に含まれる画素は、オブジェクト402に含まれない画素のカラー成分値とは異なるカラー成分値を持つ。

機能ブロック410は、外挿ブロック境界406によって囲まれるが、オブジェクト402には含まれない全ての画素412について、そのカラー成分の各々に対してあらかじめ定義された値、例えば、ゼロ値が与えられることを示す。

機能ブロック414は、外挿ブロック境界406内の画素の水平ラインが走査され、それによってゼロの、および、非ゼロのカラー成分値を持つ水平画素セグメントを持つ水平ラインが特定されることを示す。

機能ブロック416は、ゼロのカラー成分値を持つ水平画素セグメントが、その両端において、辺縁408に区画されているかどうかの調査を表わす。領域420は、ゼロのカラー成分値を持ち、かつ、ただ1端で辺縁408で区画される水平画素セグメントを表わす。機能ブロック416は、画素セグメントがゼロのカラー成分値を持ち、両端においてオブジェクト402の辺縁408で区画される領域418に対しては、機能ブロック426に進み、それ以外の場合には機能ブロック422に進む。

機能ブロック422は、領域420の各水平画素セグメントにおける画素には、対応する水平ラインにおける画素424(例示のもののみ図示)とオブジェクト402の辺縁408のカラー成分値が割り当てられることを示す。別法として、領域420の画素に割り当てられるカラー成分値は、画素424のカラー成分値に対して関数的な関係を持つ。

機能ブロック426は、領域418の各水平画素セグメントにおける画素には、対応する水平ラインと辺縁408に存在する画素428aと428bのカラー成分値の平均に相当する、できれば、その平均に等しいものが割り当てられることを示す。

機能ブロック430は、外挿ブロック境界406内の画素の垂直ラインが走査され、それによってゼロの、および、非ゼロのカラー成分値を持つ垂直画素セグメントを持つ垂直ラインが特定されることを示す。

機能ブロック432は、ゼロのカラー成分値を持つ垂直ラインにおける画素セグメントが、その両端において、オブジェクト402の辺縁408に区画されているかどうかの調査を表わす。第18C図に基づいて述べると、領域434は、ゼロのカラー成分値を持ち、かつ、両端で辺縁408で区画される垂直画素セグメントを表わす。領域436は、ゼロのカラー成分値を持ち、かつ、ただ1端で辺縁408によって区画される垂直画素セグメントを表わす。機能ブロック432は、画素セグメントがゼロのカラー成分値を持ち、両端においてオブジェクト402の辺縁408で区画される領域434に対しては、機能ブロック444に進み、それ以外の場合には機能ブロック438に進む。

機能ブロック438は、領域436の各垂直画素セグメントにおける画素には、対応する垂直ラインにおける画素442(例示のもののみ図示)とオブジェクト402の辺縁408のカラー成分値

10

20

30

40

50

が割り当てられることを示す。別法として、領域436の画素に割り当てられるカラー成分値は、画素442のカラー成分値に対して関数的な関係を持つ。

機能ブロック444は、領域434の各垂直画素セグメントにおける画素には、対応する垂直ラインと辺縁408に存在する画素446aと446bのカラー成分値の平均に相当する、できれば、その平均に等しいものが割り当てられることを示す。

機能ブロック448は、本法によってカラー成分値を割り当てられた水平ならびに垂直画素セグメント中の画素に対して、もしそうでなければその水平、および、垂直画素セグメントにしたがってその画素に割り当てられたはずのカラー成分値と比例する、できれば、その平均である複合カラー成分値が割り当てられることを示す。

そのような複合カラー成分値を割り当てられた画素の例は、領域418と434の中の画素である。

10

機能ブロック450は、外挿ブロック境界406によって区画はされるが、オブジェクト402の辺縁408とは水平ラインに沿っても、垂直ラインに沿っても交差しない領域452に対して、近傍の画素に割り当てられたカラー成分値に比例する、できれば、その平均値に等しい、複合カラー成分値が割り当てられることを示す。第18D図に基づいて説明すると、領域452の画素454の各々に対しては、それぞれ水平ラインと垂直ラインにおいて画素454と同軸にあって、本法によってあらかじめ非ゼロのカラー成分値を割り当てられている画素456aと456bのそのカラー成分値の、できれば、平均であるカラー成分値が割り当てられる。

オブジェクト外挿工程400の利点は、オブジェクト402に含まれない画素に対して、滑らかに変化するカラー成分値を割り当てることができること、したがって、従来の静止画像圧縮法の圧縮性能と精度を最適化することができることである。逆に、チャンら(Chang et al.)著の表題「Transform Coding of Arbitrary-Shaped Image Segments」、ACM Multimedia, 83~88ページ、1993年6月発行に記載されているような、ゼロ平滑化法や鏡像法のような従来技術では、機能ブロック410で実行したように、ゼロのカラー成分値を持つ画素で充満している外挿オブジェクトに対して圧縮を適用する。オブジェクトとゼロ平滑化領域の間に見られる急激な画像変化は、圧縮の困難な高周波の変化を導入させるか、または、圧縮によって画像アーティファクトを導入させる。オブジェクト外挿法400はそのような欠点を克服するものである。

20

符号化法別法

第19A図は、符号化法500の機能ブロック図であるが、この方法は、独特のフィルターを持つラプラシアン・ピラミッド符号化器を用いる。このフィルターは、高度の圧縮を実現しながらしかも、画像特徴の非直線局面、例えば、辺縁を保存する。従来のラプラシアン・ピラミッド符号化法については、例えば、パートとエイデルソン(Burt and Adleson)著「緊密画像符号としてのラプラシアン・ピラミッド(Laplacian Pyramid as a Compact Image Code)」IEEE Trans. Comm. 31巻7号523~540ページ、1983年4月号に記載されている。符号化法500は、第3図に示したビデオ圧縮符号化工程64の機能ブロック112に関連して記述した符号化法を実行することもできるし、もちろん、それ以外にも、ウェーブレット符号化法によるDCTの使用が示唆される、または、実際に使用されるところでは必ず実行可能である。実例として、符号化法500を、推定誤差110の符号化(第3図)に関連して説明することにする。

30

40

第1 デシメーションフィルター502は、推定誤差(第3図)に相当する画素情報を受け取り、この画素を、フィルター基準に従って濾過する。従来のラプラシアン・ピラミッド法では、デシメーションフィルターは、ガウス重みづけ関数のような低域フィルターである。しかしながら、符号化法500に従うならば、デシメーションフィルター502は、できれば、メディアン・フィルター、さらに詳しく言うと、3x3非分離メディアン・フィルターを用いることが好ましい。

具体的に言うと、第20A図は、画素504の、任意の一組、一配列の一つのカラー成分(例えば、赤)にたいするカラー成分値を単純化して示したものである。赤のカラー成分値に特に関連して記述しているけれども、この説明は、画素504の緑や青のカラー成分値にも同様に当てはまる。

50

デシメーションフィルタ502の好ましい実施態様に関して言えば、3x3画素の次元を持つフィルタ・ブロック506が、画素504の中に定義される。各画素ブロック506について、画素強度中央値が特定され、選択される。画素ブロック506a~506cに関して言えば、例えば、デシメーションフィルタ502は、それぞれの値8、9および10を与え、これらは、第20B図の、最初の3個の画素512に並べられる。

しかしながら、デシメーションフィルタ502は、本発明に従って、他のメディアン・フィルタを採用することもできることが了解されるであろう。従って、関連カラー成分値 $\{a_0, a_1, \dots, a_{n-1}\}$ を持つ各グループの画素に対して、メディアン・フィルタは、中央値 a_M を選択する。

第1の2x2ダウンサンプル・フィルタ514が、垂直、水平方向に一つおきの画素512をサンプルし、さらに圧縮を行なう。第20C図は、得られた、圧縮された一組の画素515を示す。2x2アップサンプル・フィルタ516は、ダウンサンプル・フィルタ514で除外された各画素512の代わりにゼロ値画素を挿入する。また、補間フィルタ518は、ゼロ値画素に対して、対立する隣接画素の平均の画素値を割り当てる、または、もしもゼロ値画素が、非ゼロ値を持つ画素の対立ペアの間には、以前に割り当てた値を割り当てる。具体的に示すと、第20D図は、そのようにして得られた、一組の、または、一配列の画素値520を示す。

差522が、一組の画素504のカラー成分値と、一組の画素520の対応するカラー成分値との間で求められ、ゼロ次画像成分 I_0 を与える。

第2のデシメーションフィルタ526は、第1の2x2ダウンサンプル・フィルタ514によって生成された圧縮された一組の画素515に対応するカラー成分値を受け取る。デシメーションフィルタ526は、できれば、デシメーションフィルタ502(例えば、3x3非分離メディアン・フィルタ)と同じものであることが好ましい。したがって、デシメーションフィルタ526は、デシメーションフィルタ502と同様に機能し、第2の2x2ダウンサンプル・フィルタ528に対して、圧縮された一組の、ないし、一配列の画素(図示せず)を転送する。

ダウンサンプル・フィルタ528は、ダウンサンプル・フィルタ514と同様に機能し、2次画像成分 L_2 を形成する。この成分はさらに2x2アップサンプル・フィルタ530と補間フィルタ531に転送される。これらのフィルタも、それぞれ、アップサンプル・フィルタ516および補間フィルタ518と同様に機能する。差532が、一組の画素515のカラー成分値と、補間フィルタ531によって与えられるカラー成分値との間で求められ、1次画像成分 I_1 を提供する。

画像成分 I_0 、 I_1 、および L_2 は、それぞれ、画素504の $n \times n$ 配列のカラー成分値を表わす $n \times n$ 、 $n/2 \times n/2$ 、 $n/4 \times n/4$

複数组のカラー成分値である。

画像成分 I_0 は、元の一組の画素504によって表わされる画像の高周波成分(例えば、辺縁)を保持する。画像成分 I_1 と L_2 は、もとの画像の低周波面を表わす。画像成分 I_0 、 I_1 、および、 L_2 は、元の画像の相対的な圧縮を実現する。画像成分 I_0 と I_1 は、隣接画素値間に比較的高い相関があるために、高い圧縮度を持つ形式に高周波特徴(例えば、辺縁)を保持する。画像成分 L_2 は、そのままではすぐには圧縮されない。なぜなら、主に低周波特徴から成るが、比較的大きさの小さい組みだからである。

第19B図は、符号化法500によって生成された画像成分 I_0 、 I_1 、および L_2 を復号する、すなわち、符号化を逆行する復号化法536を示す機能ブロック図である。復号化法536は、第1の2x2アップサンプル・フィルタ538を含む。このフィルタは、画像成分 L_2 を受け取り、各隣接ペアの画素の間にゼロ値の画素を挿入する。補間フィルタ539は、ゼロ値画素に、できれば隣接画素の値の平均の画素値を割り当てる、または、もしもゼロ値画素が、非ゼロ値画素の対立ペアの間には、以前に割り当てた値を割り当てる。第1の2x2アップサンプル・フィルタ538は、第19A図のアップサンプル・フィルタ516と530と実質的に同じやり方で動作する。また、補間フィルタ539は、補間フィルタ518と531と実質的に同じやり方で動作する。

10

20

30

40

50

合計540が、第1の2x2アップサンプル・フィルタ-538と補間フィルタ-539によって生成された、展開された一組の画素に対応する画像成分 I_1 と、カラー成分値の間で求められる。第2の2x2アップサンプル・フィルタ-542は、合計540によって生成された画素の各隣接ペアの間にゼロ値の画素を挿入する。補間フィルタ-543は、ゼロ値画素に、隣接画素値の平均を、または、ゼロ値画素が、非ゼロ値画素の対立ペアの間にはない場合には、以前に割り当てられた値を、割り当てる。アップサンプル・フィルタ-542と補間フィルタ-543は、それぞれ、アップサンプル・フィルタ-538と補間フィルタ-539と実質的に同様である。

合計544は、画像成分 I_0 と、第2の2x2アップサンプル・フィルタ-542と補間フィルタ-543によって生成される、展開した一組の画素に対応するカラー成分値とを加える。合計544は、符号化工程500に供給される推定誤差110に対応する展開した推定誤差110を提供する。

10

動きベクトルの変換符号化

従来のビデオ圧縮符号化工程、例えば、MPEG-1やMPEG-2は、粗動きベクトルのフィールドのみを用いて、規則的な大きさを持つ、相対的に大きな画素配列の動きを表わそうとする。動きベクトルのフィールドが粗であるというのは、例えば、16x16画素の大きさを持つ画素配列の動きを表わすのにたった一つの動きベクトルしか用いないからである。この粗動きベクトルのフィールドが、例えば、離散コサイン変換(DCT)符号化法によって、元となる画像ないし画素の縦列符号化と協働して、従来のビデオ圧縮符号化を実現する。

それと対照的に、ビデオ圧縮符号化法64(第3図)は、密な動きベクトルのフィールドを用いる。この場合、動きベクトルは、オブジェクトの全ベクトルに対して、または、ほとんど全てのベクトルに対して求められる。このような密な動きベクトルは、対応する画素間の動きが表現される精度を目立って向上させる。このように改善された精度は、従来の粗動きベクトル・フィールド表現に伴う誤差を目立って減少させるが、密な動きベクトル・フィールドに含まれる情報増加は、ビデオ配列を表わす情報量の増加となる。従って、本発明によれば、細密ビデオ・フィールドそれ自体が圧縮され、符号化されているわけであり、それによって、本発明の与える圧縮比を向上させる。

20

第21図は、動きベクトル符号化工程の機能ブロック図であり、この工程は、動きベクトル・フィールド、できれば、第3図の密な動き変換96に従って生成されるものと同様の密な動きベクトルであることが好ましいが、そのようなフィールドを符号化する、すなわち、圧縮するものである。ある選ばれたオブジェクトから得られるこのような密な動きベクトルは、通常、そのオブジェクトに相当する元となる画素よりも大きな連続性、または、「滑らかさ」を持っていることが了解されるであろう。そのため、密な動きベクトル・フィールドの圧縮、ないし、符号化は、元の画素の圧縮ないし符号化よりも大きな圧縮比を実現することになる。

30

機能ブロック562は、ある密な動きベクトル・フィールドが、あるオブジェクト、または、オブジェクトの一部分について、例えば、第3図に関連して上述した機能ブロック96の工程に従って得られることを示す。従って、密な動きベクトル・フィールドは、任意の形、または、大きさのオブジェクト、または、その他の画像部分に一致する。

機能ブロック564は、密な動きベクトル・フィールドの形が、規則的な形、できれば方形に外挿され、それによって、符号化ないし圧縮をやりやすくすることを示す。できれば、密な動きベクトル・フィールド形は、第17Aと17Bに関連して上述した圧縮前外挿法400によって規則的な形に外挿されることが好ましい。従来の外挿法、例えば、鏡像法も別法として用いることができることが了解されるであろう。

40

機能ブロック566は、外挿された規則的な形を持つ密な動きベクトル・フィールドが、従来の符号化変換によって符号化、または、圧縮されることを示す。そのような従来の方法としては、例えば、離散コサイン変換(DCT)、格子による圧縮、または、その他のウェーブレット圧縮があるが、このうち、前者が好ましい。

機能ブロック568は、符号化された密な動きベクトル・フィールドが、エントロピー符号化のような、従来のロスレス静止画圧縮法によってさらに圧縮ないし符号化され、それに

50

よって符号化された密な動きベクトル・フィールド570を形成することを示す。このような静止画像圧縮法は、第3図の機能ブロック114に関連して上述した。

以前のビデオ・フレームの量子化されたオブジェクトの圧縮

第3A図を参照して説明すると、ビデオ圧縮符号化工程64は、直前フレームN-1に関して求められた量子化された直前オブジェクト126を用いて、次に続くフレームNの対応するオブジェクトを符号化する。従って、符号化工程64は、量子化された直前オブジェクト126が、アクセス可能なメモリ・バッファの中に保存されていることを要求する。従来のビデオ・ディスプレイ解像度であれば、そのようなメモリ・バッファは、一つのビデオ・フレームのために、その量子化した直前フレームのオブジェクト126を保存するのに、少なくとも半メガバイトの容量を必要とする。高解像度のディスプレイ形式であれば、それに
10

第22図は、量子化オブジェクト符号化・復号化（コーデック）工程600の機能ブロック・図である。この工程は、量子化された直前オブジェクト126を圧縮し、かつ、選択的に展開し、それによって、量子化オブジェクトメモリ・バッファの必要容量の減少を実現する。

機能ブロック602は、ある画像フレーム中の各量子化オブジェクト126が、損失性符号化、ないし、圧縮法によって、ブロックごとに符号化されることを示す。そのような符号化法としては、例えば、離散コサイン変換(DCT)符号化、格子によるサブバンド圧縮法、または、その他のウェーブレット圧縮がある。第21図に示すように、損失性符号化情報は、さらにロスレス符号化を受ける。別法として、ロスレス符号化だけを用いてもよい。
20

機能ブロック604は、符号化ないし圧縮量子化オブジェクトがメモリ・バッファ(図示せず)に保存されることを示す。

機能ブロック606は、次に続くビデオ・フレームの対応するオブジェクトの処理を予測して、メモリ・バッファから、符号化量子化オブジェクトが取り出されることを示す。

機能ブロック608は、符号化量子化オブジェクトが、機能ブロック602に関連して採用された符号化法に従って、例えば、DCTまたはウェーブレット復号化によって、逆符号化されることを示す。

コーデック工程600は、対応するメモリ・バッファの容量が、全体のビデオ圧縮比と、得られるビデオの所望の画質に依存するのではあるが、最大80%まで減少するのを可能にする。さらに、コーデック工程600は、ビデオ圧縮符号化工程64に相当する復号化工程にも同様に適用可能であることが了解されるであろう。
30

ビデオ圧縮復号化工程概観

第3図のビデオ圧縮符号化工程64は、複数の画像フレームから成るビデオ配列に対応するビデオ信号の符号化ないし圧縮表現を提供する。この圧縮表現は、オブジェクトのマスク66、特徴点68、アフィン変換係数104、および、符号化工程64から得られる圧縮誤差データ116と符号化工程130から得られる圧縮マスターオブジェクト136とを含む。これらの圧縮表現は、ビデオ情報のメモリ、あるいは、転送を促進し、MPEG-2のような従来のビデオ圧縮法で実現可能な圧縮比よりも最大300%優る圧縮比を実現することができる。

しかしながら、このような圧縮ビデオ情報をデータメモリ部から取り出すこと、または、そのようなビデオ情報の転送を受け取ることは、それが復号化され、展開され、それによってもとのビデオ信号が再構成され、それによって、ビデオ・ディスプレイ装置52(第2Aと2B図)のようなディスプレイ装置によってレンダリングされることが必要であることが了解されるであろう。MPEG-1、MPEG-2およびH.26Xのような従来の符号化工程に見られるように、ビデオ情報の展開、即ち復号化は、実質的に、もとのビデオ信号が符号化され、ないし、圧縮される工程の逆工程である。
40

第23A図は、ビデオ圧縮復号化工程700の機能ブロック・図であるが、この工程は、第3図のビデオ圧縮符号化工程64によって生成されるビデオ情報を展開するものである。符号化工程64の記述と一貫させるために、復号化工程700は、第2A、2B図を参照しながら説明する。復号化工程700は、メモリから、または、転送されたものとして、符号化ビデオ情報を受け取る。この情報は、オブジェクトマスク66、特徴点68、圧縮マスターオブジェクト
50

136、アフィン変換係数104、および、圧縮誤差データ166を含む。

復号化工程700は、符号化工程64(第3図)の操作の逆の操作を実行する。したがって、復号化工程を備えた符号化工程64の、前述の好ましい操作のそれぞれも、同様に逆転される。機能ブロック702は、マスク66、特徴点68、変換係数104、および、圧縮誤差データ116がメモリから取り出され、または、転送物として受け取られ、それによって、復号化工程700による処理が可能になることを示す。

第23B図は、マスターオブジェクト復号化工程704の機能ブロック図であるが、この工程は、圧縮されたマスターオブジェクト136を復号ないし展開する。機能ブロック706は、圧縮されたマスターオブジェクトデータ136が、第3B図の機能ブロック134の、従来のロスレス・エントロピー符号化法を逆行することによって、エントロピー復号化されることを示す。機能ブロック708は、機能ブロック706においてエントロピー復号化されたマスターオブジェクトが、第3B図の機能ブロック132で用いられた、従来の損失性ウェーブレット符号化工程の逆のことを行うことによって復号化されることを示す。

機能ブロック712は、密な動き変換、できれば、多次元アフィン変換であることが好ましいが、それがアフィン係数104から生成されることを示す。できれば、アフィン係数104は、変換法350(第12図)に従って量子化されることが好ましく、かつ、そのアフィン変換は、機能ブロック362(第12図)に関連して上述した操作の逆のことを行うことによって、その量子化アフィン係数から生成されることが好ましい。

機能ブロック714は、タイミング遅延を介して与えられた、直前フレームN-1のオブジェクト716の量子化された形(例えば、画像フレーム54aの直方体56a)が、密な動き変換によって変換され、それによって、現在フレームNのオブジェクト720の予測形(例えば、画像フレーム54bの直方体50b)を提供することを示す。

機能ブロック722は、画像フレームNについて、予測した現在オブジェクト720が、圧縮誤差データ116から生成された量子化誤差724に加えられることを示す。特に、機能ブロック726は、圧縮誤差データ116が、圧縮工程114(第3A図)の逆工程によって復号化されることを示す。好ましい実施態様においては、機能ブロック114と726は、エントロピー符号化法のような、従来のロスレス静止画像圧縮法に基づいて実行される。

機能ブロック728は、機能ブロック726で得られたエントロピー復号化誤差データが、機能ブロック112(第3A図)で使用されるものに一致する、従来の損失性静止画像圧縮法によってさらに展開、即ち復号化されることを示す。好ましい実施態様においては、機能ブロック728の展開、即ち復号化は、格子化の工程ないしその他のウェーブレット工程、または、離散コサイン変換(DCT)工程によって実行される。

機能ブロック722は、フレームNの量子化オブジェクト730を、予測オブジェクト720と量子化誤差724の合計として与える。これは、再構成ないし展開されたオブジェクト723を表わし、このオブジェクトは、後続のフレームのオブジェクト再構成のために機能ブロック718に配送される。

機能ブロック734は、量子化されたオブジェクト732が、現在画像フレームNの他のオブジェクトと組み合わせられて、展開したビデオ信号を形成することを示す。

単純化したチェーン符号化

マスク、オブジェクト、スプライト、および、その他のグラフィック性の特徴は、通常、その輪郭で表現される。第5A図に関連して図示し、説明したように、例えば、直方体56aは、オブジェクト辺縁ないし輪郭142によって区画される。輪郭を符号化ないし圧縮するための従来の工程は、チェーン符号化法と呼ばれる。

第24A図は、従来の8点チェーンコード800を示す。すなわち、このコードに基づいて、従来の方形画素配列上の輪郭が定義される。現在画素位置Xに基づいて、輪郭における、次に続く画素位置が方向802a~802hの内の一つに沿って延びる。直後画素のチェーンコード値は、特定方向802に一致する数値である。例を挙げると、右、水平方向802aはチェーンコード値0に相当し、下向き、垂直方向802gはチェーンコード値6に相当する。どのような連続輪郭であれ、8点チェーンコード800によって記載することができる。

第24B図を参照しながら説明すると、XおよびA~Gと名づけられる、画素806によって表

10

20

30

40

50

わされる輪郭804は、従来のやり方では、チェーンコード{00764432}で符号化される。特に、画素Xから始めると、画素Aと画素Bは、画素XとAに対して、方向0に位置づけられる。画素Cは、画素Bに対して方向7に位置づけられる。残りの画素D~Gも同様に、上述のチェーンコードに一致する方向に位置づけられる。2 値表現では、従来の、各チェーンコード値は、3個のデジタル・ビットで表わされる。

第25A図は、本発明のチェーンコード化工程810の機能ブロック図であるが、この工程は、従来のチェーンコード化工程のものの少なくとも約2倍の圧縮比を提供することができる。チェーンコード化工程810は、このように高い圧縮比を、チェーンコードの数を制限することによって、また、隣接ペア・画素の一致軸に対してチェーンコードを定義することによって、実現する。実験によると、チェーンコード化工程810のチェーンコードは、制限されてはいるものの、オブジェクトまたはマスク輪郭の画素軸99.8%以上を表わすことが判明した。特殊ケースとしてのチェーンコード修正を設けることによって、下にさらに詳述するように、画素軸の、残りの0.2%未満を収容することができる。

10

機能ブロック816は、あるマスク、オブジェクト、または、スプライトに対して輪郭が得られることを示す。輪郭は、例えば、第4と5図に関連して上述したオブジェクトセグメント工程140によって得てもよい。

機能ブロック818は、輪郭の初期画素が特定されることを示す。初期画素は、通常の方法、例えば、最小のX軸座標値、Y軸座標値を持つ画素をもって特定してもよい。

機能ブロック820は、あらかじめ定められたチェーンコードが、初期画素と、輪郭において次に隣接する画素との間の関係を表わすように割り当てられることを示す。できれば、あらかじめ定められたチェーンコードは、順方向に一致するように定義されることが好ましい。

20

第25B図は、本発明の3点チェーンコード822を示す模式図である。チェーンコード822は、3個のチェーンコード824a、824b、および、824cを含む。これら3個のチェーンコードは、それぞれ、順方向826a、左方向826b、および、右方向826cに一致する。方向826a~826cは、現在画素830と、このチェーンコードにおいて直前画素を表わす隣接画素832との間の、直前同軸方向828に対して定義される。

直前同軸方向828は、第24A図に示す方向802のいずれに沿って延びることもできるが、説明の都合上、特定方向(すなわち、右、水平)を持つものとして示される。方向826aは、したがって、方向828と同じである。方向826bと826cは、方向828から、1 画素左、および、右変位分だけ異なる。

30

チェーンコード824の50%をわずかに上回る分が、順方向826aに一致し、チェーンコード824の25%をわずかに下回る分が、方向826bおよび826cのそれぞれに一致することが、実験的に求められた。

機能ブロック836は、輪郭において次に隣接する画素が、方向826の内の一に一致するかどうかを尋ねる調査を示す。輪郭において次に隣接する画素が、方向826の内の一に一致する場合には必ず、機能ブロック836は、機能ブロック838に進み、それ以外の場合には機能ブロック840に進む。

機能ブロック838は、次に隣接する画素に、直前隣接画素のペアの揃う軸に沿った方向828に対する、そのものの方向826に相当するチェーンコード824が割り当てられることを示す。

40

機能ブロック840は、方向826の一に一致する画素配列を、現実には一致しなかった画素配列の代用とされることを示す。実験によれば、このような代用は、通常、輪郭の画素配列の0.2%未満に起こるが、6個の特殊ケースの修正のうちの一つを用いることによって適応できることが定められた。

第25C図は、6個の特殊ケース修正842の模式図であるが、この修正は、非一致性画素配列を、方向826に一致する画素配列に変換するものである。各修正842の内部において、画素配列844は、画素配列846に変換される。それぞれの隣接画素X¹、X²、A、Bの画素配列844の各々において、画素A、B間の方向は、方向826の一つと一致しない。それは、画素X¹とX¹の軸に対する画素Aの軸の相対的位置のためである。

50

画素配列844aにおいて、初期画素軸850aと852aは、非一致的な直角方向変化を表わす。したがって、画素配列846aにおいては、画素配列844aの画素Aが除去される。これによって、画素方向854aが得られ、これは画素方向826aに一致する。画素配列修正842b～842fも同様に非一致性画素配列844b～844fを、それぞれ一致性配列846b～846fに変換する。

画素配列修正842は、直前隣接画素X1およびX2の軸に対して90°以上変化する画素方向軸をもたらす画素を除外する。一つの効果は、直角を表わす輪郭の最小曲率半径を、3画素まで増すことである。したがって、画素修正842は、極めて微妙な輪郭細部の極小の損失を招く。しかしながら、本発明によれば、このような細部の損失は、大抵の視認条件では受け入れ可能なものであることが定められた。

機能ブロック862は、機能ブロック840によって導入された、または、惹起された非一致性の画素軸方向が除去されることを示す。ある好ましい実施態様では、非一致性の方向変化は簡単に次のようにして除去される。すなわち、機能ブロック816に戻り、工程810を繰り返し、これを、非一致性の画素配列が見られなくなるまで続ける。これは通常、8回未満の繰り返しで実現される。また、別様の実施態様では、このような惹起された非一致性の方向は次のようにして「リアルタイム」で修正される。すなわち、非一致性の方向変化が修正される度ごとに、何か惹起された非一致性の方向変化がないかどうかをチェックし、あれば修正する。

機能ブロック864は、ハフマン・コードが、このようにして得られた単純化チェーンコードから生成されることを示す。方向826A～826Cは、ある輪郭において、それぞれ約50%、25%、および、25%の画素に起こるのであるが、この方向に一致するチェーンコード824a～824cに関しては、それぞれ0、11、および、10のハフマン・コードが割り当てられる。このような一次ハフマン・コードによって、チェーンコード化工程810は、輪郭の画素当たり1.5ビット未満のビット速度で輪郭を表わすことが可能となる。このようなビット速度は、従来のチェーンコード化工程にたいし、ほぼ50%の圧縮比改善となる。

さらに高次のハフマン・コードを用いることによって、さらに高い圧縮比が得られることが了解されるであろう。さらに高次のハフマン・コードとは、例えば、第1次のハフマン・コードのあらかじめ選ばれた配列に対して、あらかじめ定められた値を割り当てることを含む。

スプライト生成

本発明は、実体の動きあるビデオ（映画）を符号化するのに関連して用いられるスプライトの生成を含む。ビットマップは、ある画像源から発する連続画像の複数の連続ビットマップからなるビットマップ列に併合させる。この併合したものは、オブジェクトないし人物が、互いに相対的に動く場合、または、ある人物が、同様の他の人物を、ちょうど前景人物が背景を閉塞するような具合に閉塞する場合、そのような時の閉塞画素の問題を克服するために使用される。例えば、前景人物が移動し、何かの新しい背景を露呈する場合、次のことがなされない限り、この新しい背景を、以前のビットマップから構築することはできない。すなわち、後続のビットマップにおいて露呈される予定の画素を、先行ビットマップに含めることによって、まずその先行ビットマップを強調しておくことである。この方法は、人物の不完全な画像を捕らえ、その人物に属してはいるが、直接には見ることのできない画素について、それをいつか見る時がくるかも知れないと予測する。このような画素を用いて、その人物に対する複合ビットマップを生成する。この複合ビットマップがあれば、その人物の将来のどのような外見も、その複合ビットマップを変形することによって生成することができる。

符号化工程は、オペレーターが、人物の特定と、現在ビットマップ列からの現在ビットマップのうちのその人物の部分を選択することから始まる。特徴または変形点は、人物の部分が動き回る部分の特徴の上に、オペレーターによって選択される。そして、三角形の現在の格子を、現在ビットマップの部分に重畳させる。三角形の現在の格子を構成する三角形は、隣接変形点を結んで形成される。現在ビットマップにおける各三角形の現在位置が求められ、メモリ装置に保存される。各三角形の現在位置内にある第1画像を定義する現在ビットマップのデータの一部は、将来の使用に備えて保持される。

10

20

30

40

50

現在のビットマップ列の第2画像を定義する次続ビットマップは画像源から受け、図形および図形の部分はオペレータにより識別される。次に、現在のビットマップからの三角形の現在の格子が、次続ビットマップ上に重畳される。三角形の現在の格子の変形点は、次続ビットマップの対応図形の特徴と一致する様に再整列される。再整列された変形点は、第2画像の次続ビットマップ上に三角形の次続格子を形成する。次続ビットマップ上の各三角形の次続位置が決定されると共に記憶装置に記憶される。各三角形の次続位置内で第2画像を定義する次続ビットマップのデータの部分は、更なる使用の為に保持される。

各三角形の現在位置および次続位置を決定且つ記憶するプロセスは現在のビットマップ列の複数のビットマップ列に対して反復される。このプロセスが完了したとき、現在のビットマップ列における各三角形の平均画像は別個に保持されたデータから決定される。各三角形の平均画像は記憶装置に記憶される。

10

再生の間、現在のビットマップ列の各三角形の平均画像および現在のビットマップの各三角形の現在位置は記憶装置から検索される。予測ビットマップが生成されるが、これは、現在のビットマップ列内の各三角形の平均画像を現在のビットマップの各三角形の現在位置に変換する変換解を計算すると共に、変換解を各三角形の平均画像に適用することで行われる。予測ビットマップは、表示の為にモニタに引渡される。

再生時には制御プログラムにより画像が決定されるという再生確定モーションビデオ（ビデオゲーム）に関し、スプライト・ビットマップはその全体が記憶装置に記憶される。スプライト・ビットマップは、スプライト・画像を定義する複数のデータ・ビットから成る。スプライト・ビットマップはモニタ上に表示され、スプライトの各部分はオペレータにより識別されると共に変形点がスプライトの各部分に対して選択される。

20

三角形の格子は、スプライト・ビットマップの各部分上に重畳される。三角形の格子を構成する三角形は、隣接する変形点を接続することにより形成される。変形点は三角形の各頂点である。スプライト・ビットマップの各三角形の位置が決定されて記憶装置に記憶される。

再生の間、各三角形の次続位置は制御プログラムから受け取る。スプライト・ビットマップと、スプライト・ビットマップ上の各三角形の次続位置は、記憶装置から呼び出されてディスプレイ・プロセッサに引渡される。各三角形の次続位置もまた、ディスプレイ・プロセッサに引渡される。

スプライト・ビットマップ上の各三角形に対しては、変換解が計算される。次にディスプレイ・プロセッサ内に次続ビットマップが生成されるが、これは、各三角形の位置内のスプライト画像を定義する、スプライトのビットマップから導かれた各三角形の変換解を適用することにより行われる。ディスプレイ・プロセッサは、次続スプライトのビットマップをモニタに引渡して表示させる。このプロセスは、制御プログラムにより要求された各三角形の次続位置の各々に対して反復される。

30

図26に示される如く、動画ビデオに対する符号化処理は、画像源から現在のビットマップ列を受けるCPU22によりステップ900にて開始される。現在のビットマップ列は、連続の画像の複数の連続のビットマップから成る。また、現在のビットマップ列は、画像源からの第1画像を定義する複数のデータ・ビットからなる現在のビットマップを有する。第1画像は、少なくとも1個の部分を有する少なくとも1個の図形を備えて成る。

40

ステップ902に進み、第1画像はモニタ28上でオペレータに対して表示される。モニタ28から、現在のビットマップ上の第1画像の図形がオペレータにより識別される。現在のビットマップ上の図形の各部分は次にステップ904にてオペレータにより識別される。

次にステップ906にて、オペレータは現在のビットマップ上の特徴点もしくは変形点を選択する。変形点は、部分の相対的な動きが生じ得るところのビットマップ上の特徴と変形点が一一致する如く選択される。当業者であれば、図形、図形の各部分およびビットマップ上の変形点はコンピュータシステム20もしくはその助けを借りて識別され得ることを理解し得よう。しかし乍ら、図形、図形の各部分およびビットマップ上の変形点をオペレータが識別すれば好適である。

ステップ908に進むと、三角形の現在の格子がコンピュータシステム20により、現在のビ

50

ットマップの各部分に重畳される。図 2 7 (A)を参照すると、現在の格子は、隣接変形点を接続して形成された三角形から成っている。変形点は三角形の頂点を形成する。より詳細に述べると、現在のビットマップの第 1 画像は、人物970である図形から成る。人物970は、頭972、胴体974、右腕976、左腕978、右足980および左足982に対応する 6 個の部分有している。変形点は人物970に各部分に対して選択され、従って、変形点は、部分の相対的な動きが生じ得るところの特徴に一致する。各部分上にて、現在の格子には、隣接変形点を接続することにより形成された現在の格子の各々の三角形が重畳される。従って、変形点は三角形の頂点を形成する。

ステップ910にては、コンピュータシステム20が、現在のビットマップ上の各三角形の現在位置を決定する。現在のビットマップ上の各三角形の現在位置は、三角形の頂点を形成する変形点の位置により定義される。ステップ912においては、各三角形の現在位置が記憶装置に記憶される。各三角形の現在位置内の第 1 画像を定義する現在のビットマップから導出されたデータの部分は、ステップ914にて保持される。

次に、ステップ916にては、現在のビットマップ列の次続ビットマップがCPU22により受け取られる。次続ビットマップは、現在のビットマップ列の第 2 画像を定義する複数のデータ・ビットから成る。第 2 画像は、第 1 画像内の図形に対応する図形を含むこともあり含まないこともある。但し、次のステップに対しては、第 1 画像内の図形に対応する図形を第 2 画像が有しているものと仮定する。ステップ918にて、三角形の現在の格子が次続ビットマップ上に重畳される。重畳された三角形形状格子を備えた第 2 画像は、モニタ28上でオペレータに対して表示される。

ステップ920にて、変形点は、コンピュータシステム20の助けを借りたオペレータにより、次続ビットマップ上の対応特徴と一致する如く再整列される。コンピュータシステム20は、ブロック整合を用いて変形点を再整列する。誤りがあれば、オペレータにより訂正される。図 2 7 (B)を参照すると、再整列された変形点は、三角形の次続格子を形成する。再整列された変形点は、三角形の頂点である。より詳細に述べると、人物970の次続ビットマップの第 2 画像は、頭972、胴体974、右腕976、左腕978、右足980および左足982を含んでいる。但し、第 2 画像においては右腕980が上がっている。

第 1 画像の現在の格子は、各部分に重畳されると共に、それらの変形点は、第 2 画像上の対応特徴に一致すべく再整列されている。再整列された変形点は、三角形の次続格子を定義する。次続格子は、再整列された変形点を接続して形成された三角形から成る。故に、再整列された変形点は、次続格子の三角形の頂点を形成する。

ステップ922に進むと、次続ビットマップの各三角形の次続位置がコンピュータシステム20により決定される。ステップ924では、次続ビットマップ上の各三角形の次続位置が記憶装置に記憶される。各三角形の次続位置内の第 2 画像を定義する、次続ビットマップから導出されたデータの部分は、ステップ926で保持される。ステップ926は、次の次続ビットマップが存在するか否かを決定する判断ステップ928に続いている。

もし次の次続ビットマップが存在すれば、判断ステップ928のYES分岐は、次続ビットマップが現在のビットマップになるステップ930に進む。ステップ930は、現在のビットマップ列の次続ビットマップがCPU22により受け取られるステップ916に戻る。もし次の次続ビットマップが存在しなければ、判断ステップ928のNO分岐は、現在のビットマップ列の各三角形に対する平均画像が決定されるステップ932に進む。平均画像は、三角形の各画素の中央値である。平均画像を使用することにより、プロセスの劣化が生じにくくなる。ステップ934に進み、現在のビットマップ列の各三角形の平均画像は記憶装置に記憶される。

次に、ステップ936にては、現在のビットマップ上の各三角形の現在位置が記憶装置から検索される。次に、ステップ938にて、各三角形の平均画像を現在のビットマップ上の三角形の現在位置に変換する為のアファイン変換解がコンピュータシステム20により計算される。ステップ940にては、各三角形の平均画像の変換解を、現在のビットマップ上の各三角形の現在位置に適用することにより、予測ビットマップが生成される。ステップ942にて、予測ビットマップは現在のビットマップと比較される。

ステップ944にては、修正ビットマップが生成される。修正ビットマップは、予測ビット

10

20

30

40

50

マップにより正確に予測されなかった現在のビットマップのデータ・ビットから成る。修正ビットマップはステップ948にて記憶装置に記憶される。ステップ948は、次続ビットマップが存在するか否かを決定する判断ステップ950に進む。

もし次続ビットマップが存在すれば、判断ステップ950のYES分岐は、次続ビットマップが現在のビットマップになるステップ952に行く。ステップ952は、現在のビットマップ上の各三角形の現在位置が記憶装置から検索されるステップ936に戻る。もし次の次続ビットマップが存在しなければ、判断ステップ950のNO分岐は、次続ビットマップ列が存在するか否かを決定する判断ステップ954に行く。もし次続ビットマップ列が存在しなければ、符号化は終了すると共に判断ステップ954のNO分岐はステップ956に行く。もし次続ビットマップ列が存在すれば、判断ステップ954のYES分岐は、次続ビットマップ列を現在のビットマップ列としてCPU22が受け取るステップ958に行く。ステップ956は、現在のビットマップ列の第1画像の図形がオペレータにより識別されるステップ902に戻る。

図26のプロセスは、図3のエンコードプロセス64により使用されるスプライト・オブジェクトもしくはマスターオブジェクト90の生成法を記述している。マスターオブジェクト90を利用して予測オブジェクト102を形成するプロセスは、図28を参照して記述する。

図28に示される様に、現在のビットマップ列を検索することにより処理はステップ1000から開始される。現在のビットマップ列は連続画像の複数の連続ビットマップから成る。現在のビットマップ列は、画像源から第1画像を定義する複数のデータ・ビットから成る現在のビットマップを有する。第1画像は、少なくとも1個の部分を有する少なくとも1個の図形から成る。

ステップ1002において、現在のビットマップ列の各三角形の平均画像は記憶装置から検索される。各三角形の平均画像は次にステップ1004にて（不図示の）ディスプレイ・プロセッサに引渡される。（図1の）コンピュータシステム20が、本発明のプロセスを実行する為のディスプレイ・プロセッサもしくは他の専用要素を選択的に含み得ることは理解される。ステップ1006に進み、現在のビットマップ上の各三角形の現在位置が記憶装置から検索される。各三角形の現在位置はステップ1008でディスプレイ・プロセッサに引渡される。次に、各三角形の平均画像を、現在のビットマップ上の各三角形の現在位置に変換する為のアフィン変換解がステップ1010でディスプレイ・プロセッサにより計算される。ステップ1012に進み、ディスプレイ・プロセッサにより予測ビットマップが生成されるが、これは、各三角形の平均画像を、現在のビットマップ上の各三角形の現在位置に変換する為の変換解を適用することで行われる。

ステップ1014にて、現在のビットマップに対する修正ビットマップが記憶装置から検索される。修正ビットマップは、ステップ1016にてディスプレイ・プロセッサに引渡される。次に、予測ビットマップに修正ビットマップを被せることにより、ディスプレイ・プロセッサ内で表示用ビットマップが生成される。ディスプレイ・プロセッサは、各三角形の平均画像のコピーを保持すると共に、モニタ上の表示の為にフレームバッファに表示用ビットマップを引渡す。

次に、判断ステップ1020にて、現在のビットマップ列の次続ビットマップが存在するか否かが決定される。もし現在のビットマップ列の次続ビットマップが存在すれば、判断ステップ1020のYES分岐はステップ1022に行く。ステップ1022にて、次続ビットマップは現在のビットマップとなる。ステップ1022では、現在のビットマップの各三角形の位置が記憶装置から検索される。

判断ステップ1020に戻り、もし現在のビットマップ列の次続ビットマップが存在しなければ、判断ステップ1020のNO分岐は判断ステップ1024に行く。判断ステップ1024にては、次続ビットマップ列が存在するか否かが決定される。もし次続ビットマップ列が存在しなければ、プロセスは終了すると共に、判断ステップ1024のNO分岐はステップ1026に行く。もし次続ビットマップ列が存在すれば、判断ステップ1024のYES分岐はステップ1028に行く。ステップ1028にては、次続ビットマップ列は現在のビットマップ列になる。ステップ1028はステップ1000に戻る。

本発明の原理を好適実施例に関して図示すると共に説明したが、当業者であれば、斯かる

10

20

30

40

50

原理から逸脱することなく実施例の配置構成および詳細を改変し得ることは自明であろう。従って、本出願人は、本願の特許請求の範囲およびその均等物の範囲および精神の範囲内に収まる全ての実施例に対して権利を主張するものである。

【図面の簡単な説明】

第1図は、本発明を体現する方法、ならびに装置を導入するのに用いられるコンピュータ装置のブロック図である。

第2A図は、あるビデオ・ディスプレイ装置のディスプレイ画面の模式図であって、あるビデオ信号に対応する、二つの連続する画像フレームを示す図である。

第2B図は、あるビデオ・ディスプレイ装置のディスプレイ画面の模式図であって、あるビデオ信号に対応する、二つの連続する画像フレームを示す図である。

第3A図は、本方法によって、複数画像フレームから成るビデオ配列において表示の動きを表わすデジタル化ビデオ信号を圧縮する、ビデオ圧縮符号化方法の総合機能ブロック図である。

第3B図は、本発明によるマスターオブジェクト符号化方法の機能ブロック図である。

第4図は、あるビデオ配列のある画像フレームにおける選ばれたオブジェクトをセグメントするための、オブジェクトセグメント工程の機能ブロック図である。

第5A図は、第2A図のビデオ・ディスプレイ装置のディスプレイ画面の模式図である。

第5B図は、第5A図のディスプレイ画面の一部の拡大表示例である。

第6図は、連続する画像フレームの対応オブジェクトにおける、対応する画素・ペアにたいする動きベクトルを求める、ポリゴン・マッチングを示す機能ブロック図である。

第7A図は、二つの対応するオブジェクトを含む、二つの連続する画像フレームを示すディスプレイ画面の模式図である。

第7B図は、二つの対応するオブジェクトを含む、二つの連続する画像フレームを示すディスプレイ画面の模式図である。

第8図は、インタラクティブな画素ブロック関連工程の機能ブロック・図である。

第9A図は、異なる画像フレームにおいて、対応する画素を特定するために使用される第1画素ブロックの模式図である。

第9B図は、対応画素が探索される前の画像フレームにおける探索域に対応する画素配列の模式図である。

第9C図は、対応画素を特定するために、第9Bの画素配列において走査される第1画素ブロックの模式図である。

第9D図は、対応画素を特定するために、第9Bの画素配列において走査される第1画素ブロックの模式図である。

第9E図は、対応画素を特定するために、第9Bの画素配列において走査される第1画素ブロックの模式図である。

第9F図は、対応画素を特定するために、第9Bの画素配列において走査される第1画素ブロックの模式図である。

第9G図は、対応画素を特定するために、第9Bの画素配列において走査される第1画素ブロックの模式図である。

第10A図は、異なる画像フレームにおいて、対応する画素を特定するために使用される第2画素ブロックの模式図である。

第10B図は、対応画素を特定するために、第9Bの画素配列において走査される第2画素ブロックの模式図である。

第10C図は、対応画素を特定するために、第9Bの画素配列において走査される第2画素ブロックの模式図である。

第10D図は、対応画素を特定するために、第9Bの画素配列において走査される第2画素ブロックの模式図である。

第10E図は、対応画素を特定するために、第9Bの画素配列において走査される第2画素ブロックの模式図である。

第10F図は、対応画素を特定するために、第9Bの画素配列において走査される第2画素

10

20

30

40

50

ブロックの模式図である。

第11A図は、異なる画像フレームにおいて、対応する画素を特定するために使用される第3画素ブロックの模式図である。

第11B図は、対応画素を特定するために、第9Bの画素配列において走査される第3画素ブロックの模式図である。

第11C図は、対応画素を特定するために、第9Bの画素配列において走査される第3画素ブロックの模式図である。

第11D図は、対応画素を特定するために、第9Bの画素配列において走査される第3画素ブロックの模式図である。

第11E図は、対応画素を特定するために、第9Bの画素配列において走査される第3画素ブロックの模式図である。

第11F図は、対応画素を特定するために、第9Bの画素配列において走査される第3画素ブロックの模式図である。

第12図は、多次元変換法を示す機能ブロック・図であり、この方法は、連続する第1、第2画像フレームにおけるオブジェクト間のマッピングを生成し、このマッピングを転送またはメモリのために量子化するものである。

第13図は、第7図の画像フレームを表わすディスプレイ画面の模式図であって、第12図の多次元変換法を例示するためのものである。

第14図は、第12図の方法で求めたアフィン変換係数の量子化に使用される変換ブロックの3種の選ばれた画素を表わす拡大模式図である。

第15図は、第12図の多次元変換法の別の実施態様に用いられる変換ブロック最適化法の機能ブロック図である。

第16図は、第7図の画像フレームを表わすディスプレイ画面の、断片模式図であって、第15図の変換ブロック最適化法を例示する図である。

第17A図は、圧縮前外挿法の機能ブロック図であり、この方法は、圧縮をやり易くするために、任意の形の画像特徴を、特定の形に外挿するための図である。

第17B図は、圧縮前外挿法の機能ブロック図であり、この方法は、圧縮をやり易くするために、任意の形の画像特徴を、特定の形に外挿するための図である。

第18A図は、単純なオブジェクトを、第14図の外挿法の様々な面を示すようにレンダリングさせたディスプレイ画面を表わす図である。

第18B図は、単純なオブジェクトを、第14図の外挿法の様々な面を示すようにレンダリングさせたディスプレイ画面を表わす図である。

第18C図は、単純なオブジェクトを、第14図の外挿法の様々な面を示すようにレンダリングさせたディスプレイ画面を表わす図である。

第18D図は、単純なオブジェクトを、第14図の外挿法の様々な面を示すようにレンダリングさせたディスプレイ画面を表わす図である。

第19A図は、本発明にしたがってラプラシアン・ピラミッド符号化法を用いた、符号化法および復号化法を示す機能ブロック図である。

第19B図は、本発明にしたがってラプラシアン・ピラミッド符号化法を用いた、符号化法および復号化法を示す機能ブロック図である。

第20A図は、第19Aの符号化工程によって処理された任意の一組の、または、一配列の画素のカラー成分値を示す模式図である。

第20B図は、第19Aの符号化工程によって処理された任意の一組の、または、一配列の画素のカラー成分値を示す模式図である。

第20C図は、第19Aの符号化工程によって処理された任意の一組の、または、一配列の画素のカラー成分値を示す模式図である。

第20D図は、第19Aの符号化工程によって処理された任意の一組の、または、一配列の画素のカラー成分値を示す模式図である。

第21図は、本発明による動きベクトル符号化法の機能ブロック図である。

第22図は、量子化されたオブジェクト符号化・復号化別様工程を示す機能ブロック図で

10

20

30

40

50

ある。

第 2 3 A 図は、第 3 図の復号化工程に適合するビデオ圧縮復号化工程の総合機能ブロック・図である。

第 2 3 B 図は、第 3 図の復号化工程に適合するビデオ圧縮復号化工程の総合機能ブロック・図である。

第 2 4 A 図は、従来のチェーン・コード形式の模式図である。

第 2 4 B 図は、第 24A のチェーン・コード形式処理された場合の輪郭を例示する模式図である。

第 2 5 A 図は、本発明のチェーンコード化工程の機能ブロック図である。

第 2 5 B 図は、本発明のチェーンコード形式の模式図である。

第 2 5 C 図は、第 25A の工程で使用される特別ケース・チェーンコード修正を示す模式図である。

第 2 6 図は、スプライト生成、ないし、符号化工程を示す機能ブロック・図である。

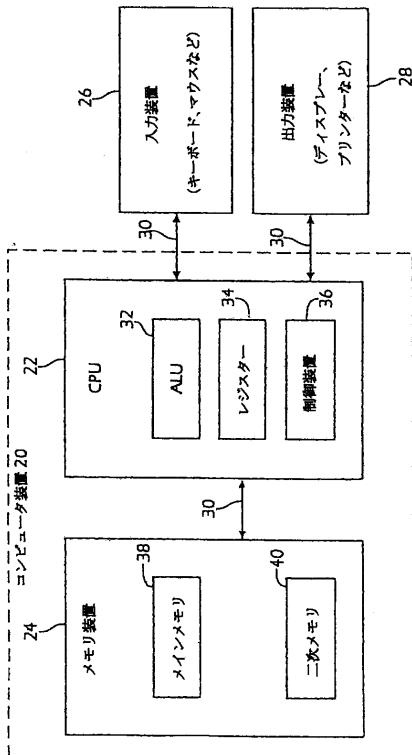
第 2 7 A 図は、ビットマップによって定義される、それぞれ、第 1 および第 2 のオブジェクトであり、第 26 の工程に従ってオブジェクトに重畳した三角の格子を示す図である。

第 2 7 B 図は、ビットマップによって定義される、それぞれ、第 1 および第 2 のオブジェクトであり、第 26 の工程に従ってオブジェクトに重畳した三角の格子を示す図である。

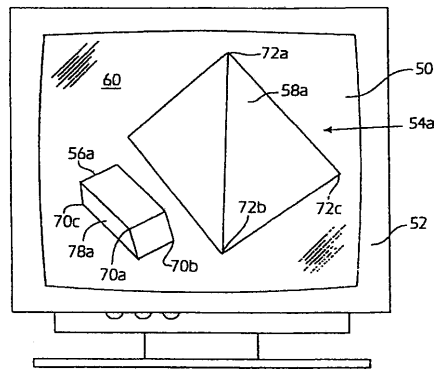
第 2 8 図は、第 26 図の符号化工程に対応する、スプライト復号化工程の機能ブロック図である。

10

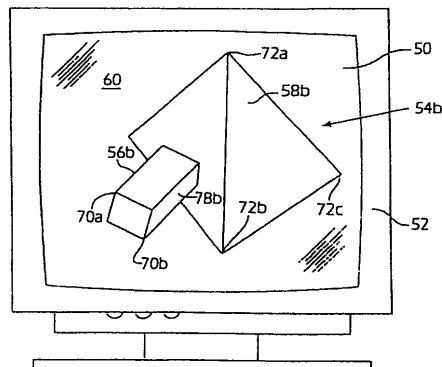
【図 1】



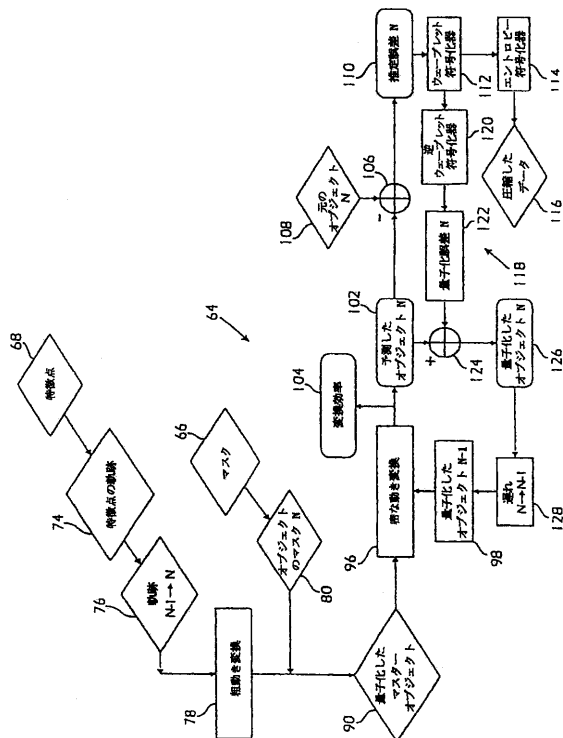
【図 2 A】



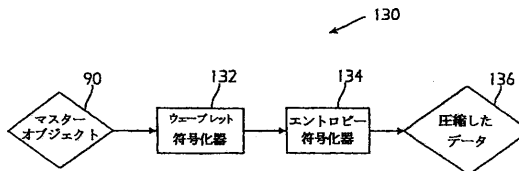
【図 2 B】



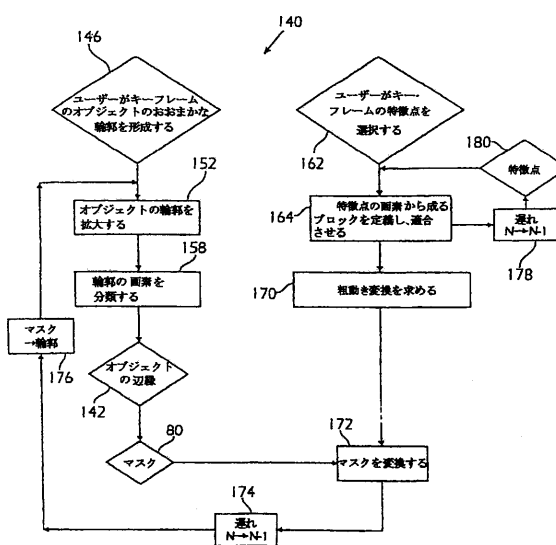
【 図 3 A 】



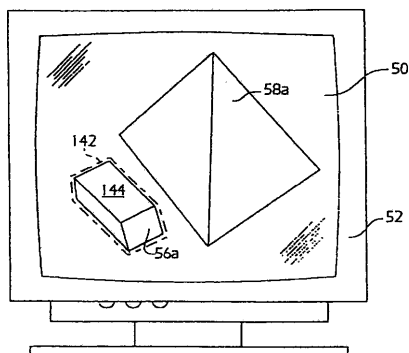
【 ㄨ 3 B 】



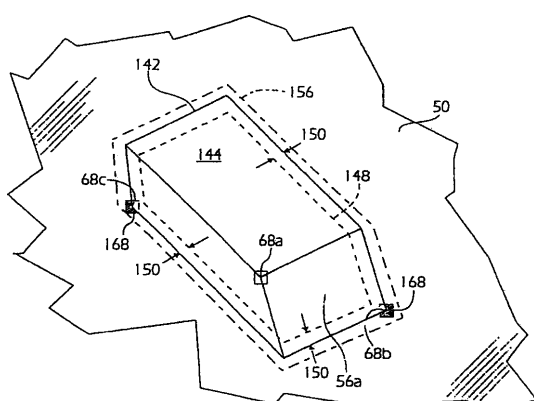
【 図 4 】



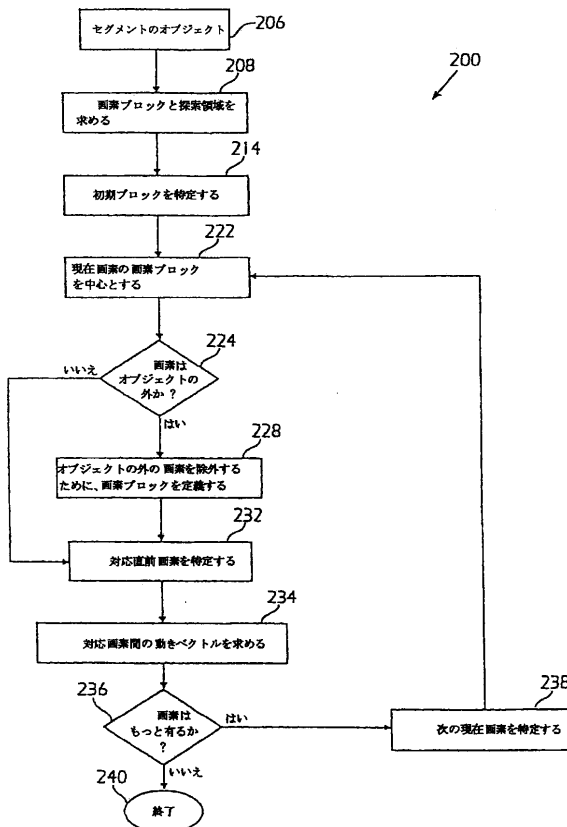
【 図 5 A 】



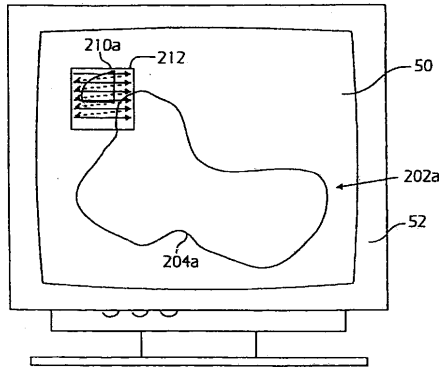
【 図 5 B 】



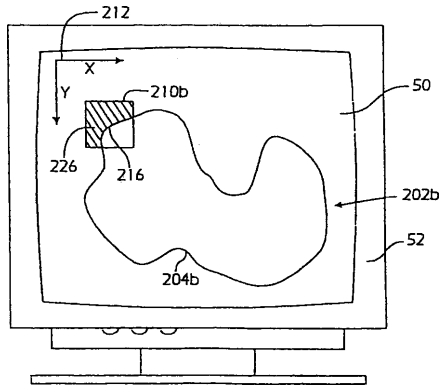
【 図 6 】



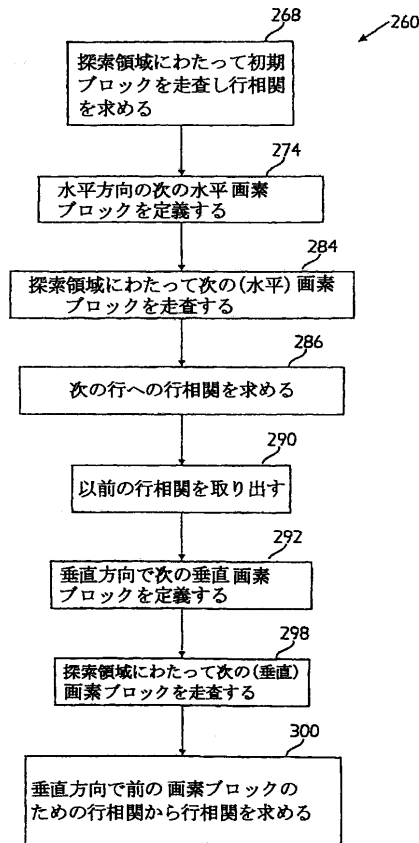
【図 7 A】



【図 7 B】



【図 8】



【図 9 A】

開始ブロック					262
A	B	C	D	E	
F	G	H	I	J	
K	L	M	N	O	
P	Q	R	S	T	
U	V	W	X	Y	

【図 9 B】

オブジェクト										266
01	02	03	04	05	06	07	08	09	00	
12	13	14	15	16	17	18	19	10	11	
23	24	25	26	27	28	29	20	21	22	
34	35	36	37	38	39	30	31	32	33	
45	46	47	48	49	40	41	42	43	44	
56	57	58	59	50	51	52	53	54	55	
67	68	69	60	61	62	63	64	65	66	
78	79	70	71	72	73	74	75	76	77	
89	80	81	82	83	84	85	86	87	88	
90	91	92	93	94	95	96	97	98	99	

【図 9 C】

開始ブロック・スキャン・オブジェクト (ステップ1)										266
01E	02	03	04	05	06	07	08	09	00	
12J	13	14	15	16	17	18	19	10	11	
23O	24	25	26	27	28	29	20	21	22	
34T	35	36	37	38	39	30	31	32	33	
45Y	46	47	48	49	40	41	42	43	44	
56	57	58	59	50	51	52	53	54	55	
67	68	69	60	61	62	63	64	65	66	
78	79	70	71	72	73	74	75	76	77	
89	80	81	82	83	84	85	86	87	88	
90	91	92	93	94	95	96	97	98	99	

【図 9 D】

開始ブロック・スキャン・オブジェクト (ステップ2)										266
01D	02E	03	04	05	06	07	08	09	00	
12I	13J	14	15	16	17	18	19	10	11	
23N	24O	25	26	27	28	29	20	21	22	
34S	35T	36	37	38	39	30	31	32	33	
45X	46Y	47	48	49	40	41	42	43	44	
56	57	58	59	50	51	52	53	54	55	
67	68	69	60	61	62	63	64	65	66	
78	79	70	71	72	73	74	75	76	77	
89	80	81	82	83	84	85	86	87	88	
90	91	92	93	94	95	96	97	98	99	

【図 9 E】

開始 ブロック・スキャン・オブジェクト (ステップ5)										266
270(4)	02B	03C	04D	05E	06	07	08	09	00	
01A	12F	13G	14H	15I	16J	17	18	19	10	
23K	24L	25M	26N	27O	28	29	20	21	22	
34P	35Q	36R	37S	38T	39	30	31	32	33	
45U	46V	47W	48X	49Y	40	41	42	43	44	
56	57	58	59	50	51	52	53	54	55	
67	68	69	60	61	62	63	64	65	66	
78	79	70	71	72	73	74	75	76	77	
89	80	81	82	83	84	85	86	87	88	
90	91	92	93	94	95	96	97	98	99	

【図 9 F】

開始 ブロック・スキャン・オブジェクト (ステップ6)										266
270(9)	02A	03B	04C	05D	06E	07	08	09	00	
01	12F	13G	14H	15I	16J	17	18	19	10	
23	24K	25L	26M	27N	28O	29	20	21	22	
34	35P	36Q	37R	38S	39T	30	31	32	33	
45	46U	47V	48W	49X	40Y	41	42	43	44	
56	57	58	59	50	51	52	53	54	55	
67	68	69	60	61	62	63	64	65	66	
78	79	70	71	72	73	74	75	76	77	
89	80	81	82	83	84	85	86	87	88	
90	91	92	93	94	95	96	97	98	99	

【図 9 G】

開始 ブロック・スキャン・オブジェクト (ステップ7)										266
270(14)	02	03	04	05	06	07	08	09	00	
01	12A	13B	14C	15D	16E	17	18	19	10	
23F	24G	25H	26I	27J	28	29	20	21	22	
34K	35L	36M	37N	38O	39	30	31	32	33	
45P	46Q	47R	48S	49T	40	41	42	43	44	
56U	57V	58W	59X	50Y	51	52	53	54	55	
67	68	69	60	61	62	63	64	65	66	
78	79	70	71	72	73	74	75	76	77	
89	80	81	82	83	84	85	86	87	88	
90	91	92	93	94	95	96	97	98	99	

【図10A】

次続水平ブロック

B	C	D	E	U'
G	H	I	J	V'
L	M	N	O	W'
Q	R	S	T	X'
V	W	X	Y	Y'

280 278

276

【図10B】

次続水平ブロック・スキャン・オブジェクト (スキャン1)

01U'	02	03	04	05	06	07	08	09	00	...
12V'	13	14	15	16	17	18	19	10	11	...
23W'	24	25	26	27	28	29	20	21	22	...
34X'	35	36	37	38	39	30	31	32	33	...
45Y'	46	47	48	49	40	41	42	43	44	...
56	57	58	59	50	51	52	53	54	55	...
67	68	69	60	61	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

【図10C】

次続水平ブロック・スキャン・オブジェクト (スキャン2)

01E'	02U'	03	04	05	06	07	08	09	00	...
12J'	13V'	14	15	16	17	18	19	10	11	...
23O'	24W'	25	26	27	28	29	20	21	22	...
34T'	35X'	36	37	38	39	30	31	32	33	...
45Y'	46Y'	47	48	49	40	41	42	43	44	...
56	57	58	59	50	51	52	53	54	55	...
67	68	69	60	61	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

【図11A】

次続垂直ブロック

F	G	H	I	J
K	L	M	N	O
P	Q	R	S	T
U	V	W	X	Y
A'	B'	C'	D'	E'

296

294

【図11B】

開始ブロック・スキャン・オブジェクト (スキャンQ+1)

01	02	03	04	05	06	07	08	09	00	...
12J'	13	14	15	16	17	18	19	10	11	...
23O'	24	25	26	27	28	29	20	21	22	...
34T'	35	36	37	38	39	30	31	32	33	...
45Y'	46	47	48	49	40	41	42	43	44	...
56E'	57	58	59	50	51	52	53	54	55	...
67	68	69	60	61	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

【図11C】

開始ブロック・スキャン・オブジェクト (スキャンQ+2)

01	02	03	04	05	06	07	08	09	00	...
12J'	13J'	14	15	16	17	18	19	10	11	...
23N	24O	25	26	27	28	29	20	21	22	...
34S	35T	36	37	38	39	30	31	32	33	...
45X	46Y	47	48	49	40	41	42	43	44	...
56D'	57E'	58	59	50	51	52	53	54	55	...
67	68	69	60	61	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

【図10D】

次続水平ブロック・スキャン・オブジェクト (スキャン3)

01D'	02E'	03U'	04	05	06	07	08	09	00	...
12I'	13J'	14V'	15	16	17	18	19	10	11	...
23N	24O	25W'	26	27	28	29	20	21	22	...
34S	35T	36X'	37	38	39	30	31	32	33	...
45X	46Y	47Y'	48	49	40	41	42	43	44	...
56	57	58	59	50	51	52	53	54	55	...
67	68	69	60	61	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

【図10E】

次続水平ブロック・スキャン・オブジェクト (スキャン6)

01	02B'	03C'	04D'	05E'	06U'	07	08	09	00	...
12	13G	14H	15I	16J	17V'	18	19	10	11	...
23	24L	25M	26N	27O	28W'	29	20	21	22	...
34	35Q	36R	37S	38T	39X'	30	31	32	33	...
45	46V	47W	48X	49Y	40Y'	41	42	43	44	...
56	57	58	59	50	51	52	53	54	55	...
67	68	69	60	61	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

【図10F】

次続水平ブロック・スキャン・オブジェクト (スキャンQ+6)

01	02	03	04	05	06	07	08	09	00	...
12	13G	14C	15D	16E	17U'	18	19	10	11	...
23	24G	25H	26I	27J	28V'	29	20	21	22	...
34	35L	36M	37N	38O	39W'	30	31	32	33	...
45	46Q	47R	48S	49T	40X'	41	42	43	44	...
56	57V	58W	59X	50Y	51Y'	52	53	54	55	...
67	68	69	60	61	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

【図11D】

開始ブロック・スキャン・オブジェクト (スキャンQ+5)

01	02	03	04	05	06	07	08	09	00	...
12F	13G	14H	15I	16J	17	18	19	10	11	...
23K	24L	25M	26N	27O	28	29	20	21	22	...
34P	35Q	36R	37S	38T	39	30	31	32	33	...
45U	46V	47W	48X	49Y	40	41	42	43	44	...
56A'	57B'	58C'	59D'	50E'	51	52	53	54	55	...
67	68	69	60	61	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

【図11E】

開始ブロック・スキャン・オブジェクト (スキャンQ+6)

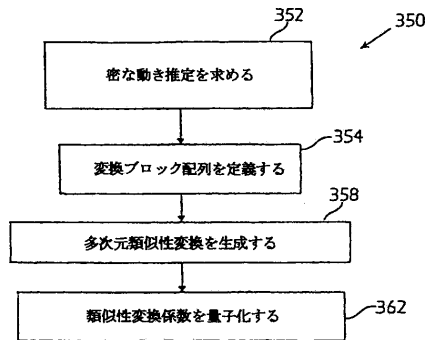
01	02	03	04	05	06	07	08	09	00	...
12	13F	14G	15H	16O	17J	18	19	10	11	...
23	24K	25L	26M	27N	28O	29	20	21	22	...
34	35P	36Q	37R	38S	39T	30	31	32	33	...
45	46U	47V	48W	49X	40Y	41	42	43	44	...
56	57A'	58B'	59C'	50D'	51E'	52	53	54	55	...
67	68	69	60	61	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

【図11F】

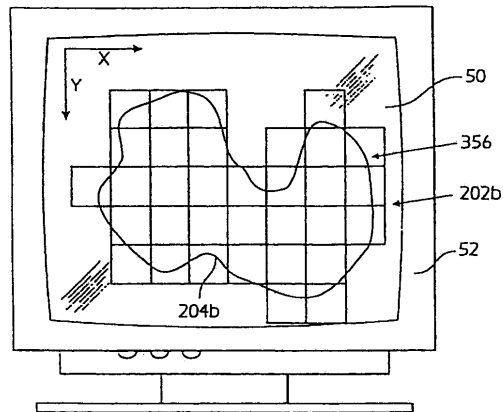
開始ブロック・スキャン・オブジェクト (スキャンQ+5)

01	02	03	04	05	06	07	08	09	00	...
18	13	14	15	16	17	18	19	10	11	...
22F	24G	25H	26I	J	28	29	20	21	22	...
34K	35L	36M	37N	O	39	30	31	32	33	...
49P	46Q	47R	48S	T	40	41	42	43	44	...
56U	57V	58W	59X	Y	51	52	53	54	55	...
61A'	68B'	69C'	60D'	E'	62	63	64	65	66	...
78	79	70	71	72	73	74	75	76	77	...
89	80	81	82	83	84	85	86	87	88	...
90	91	92	93	94	95	96	97	98	99	...

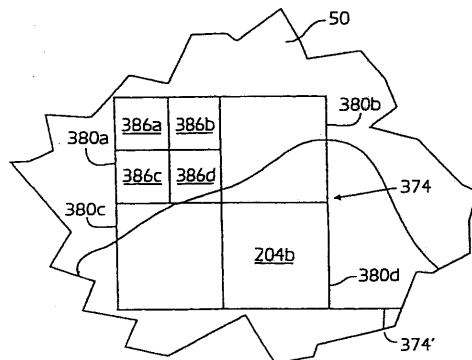
【図 12】



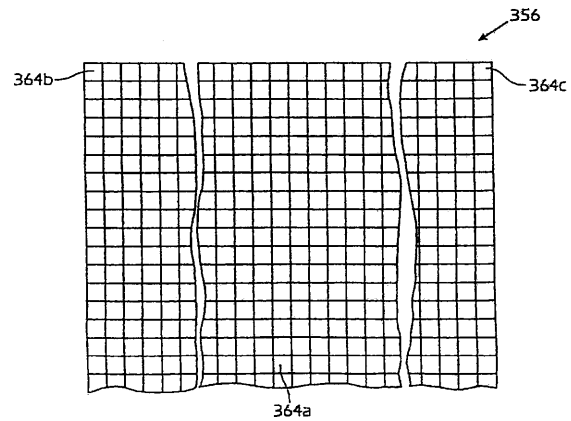
【図 13】



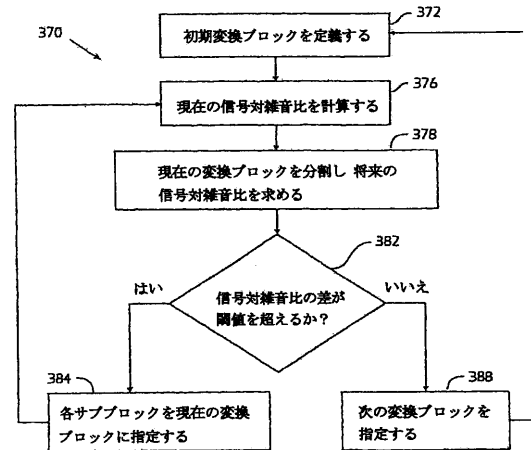
【図 16】



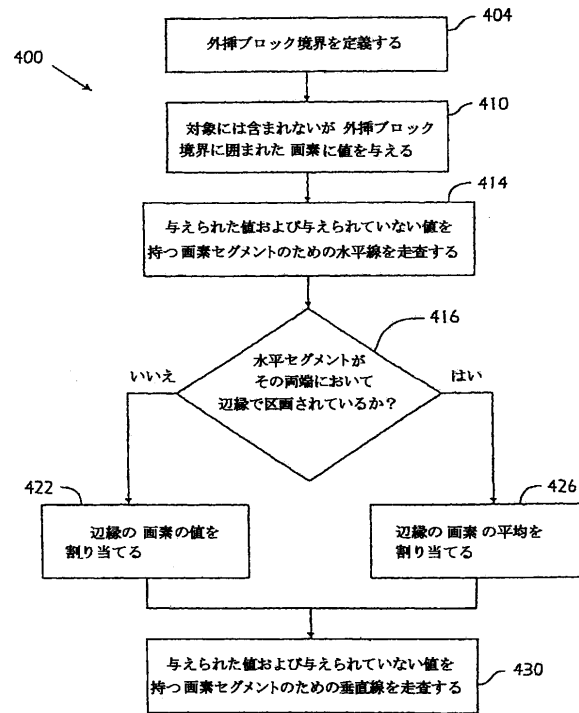
【図 14】



【図 15】

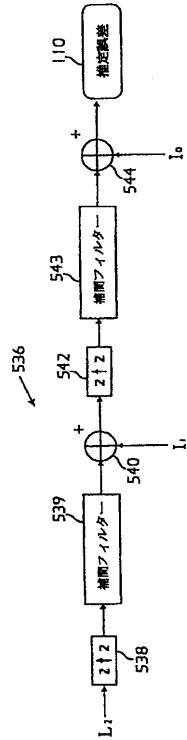


【図 17A】

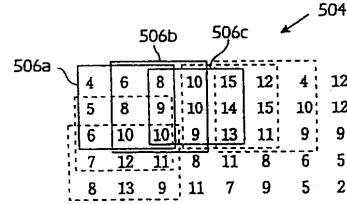


(Fig. 17B へ)

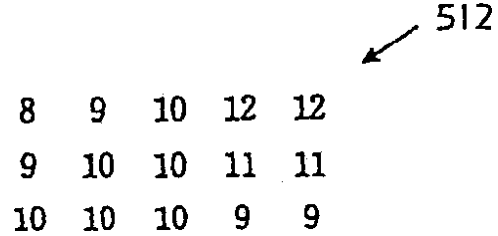
【図 19 B】



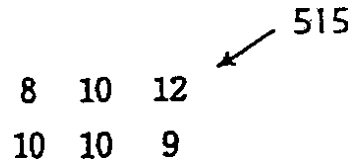
【図 20 A】



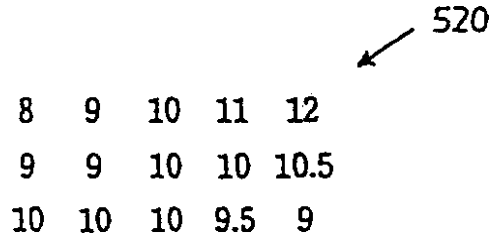
【図 20 B】



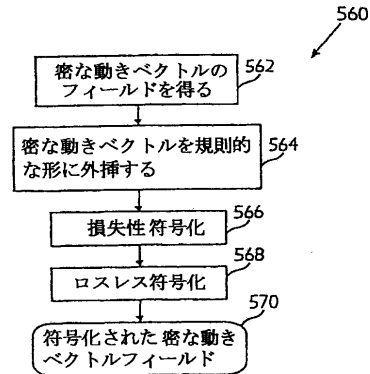
【図 20 C】



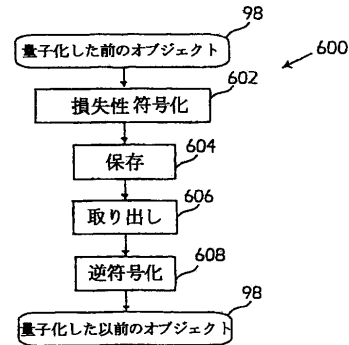
【図 20 D】



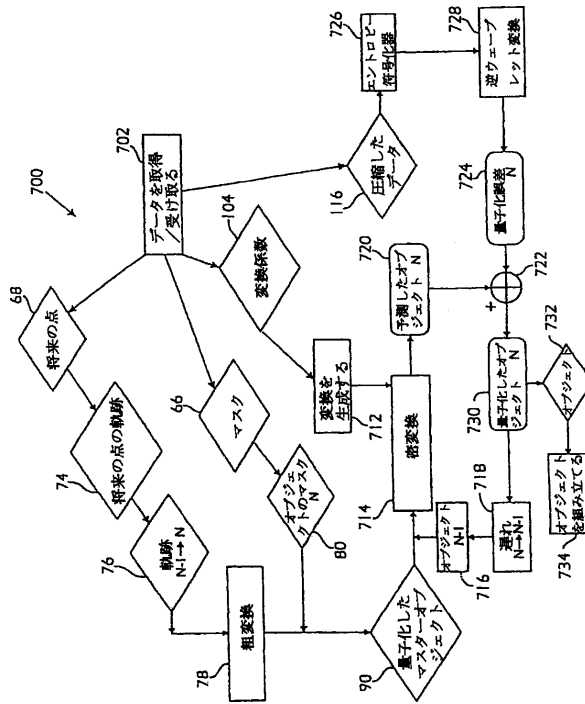
【図 21】



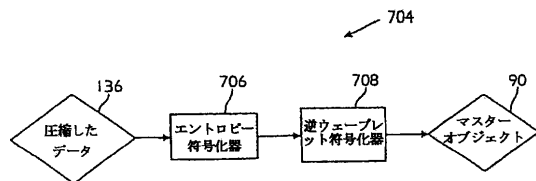
【図 22】



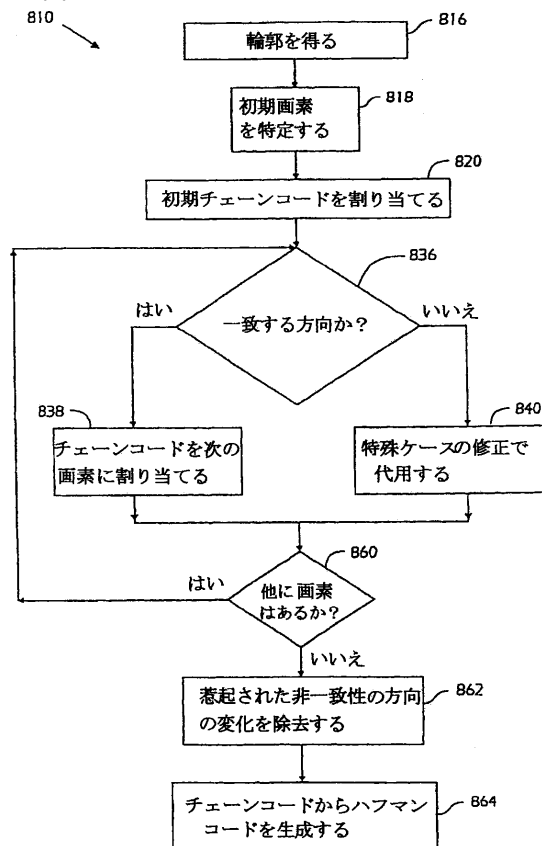
【図 23 A】



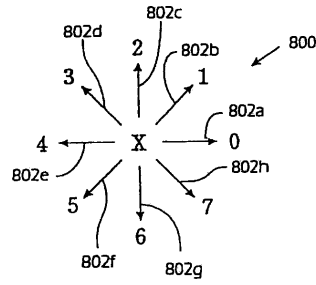
【図 23 B】



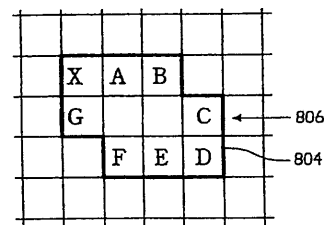
【図 25 A】



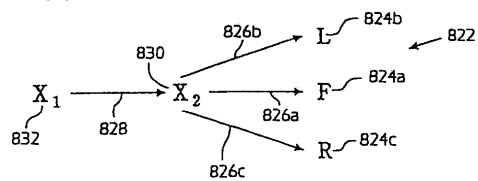
【図 24 A】



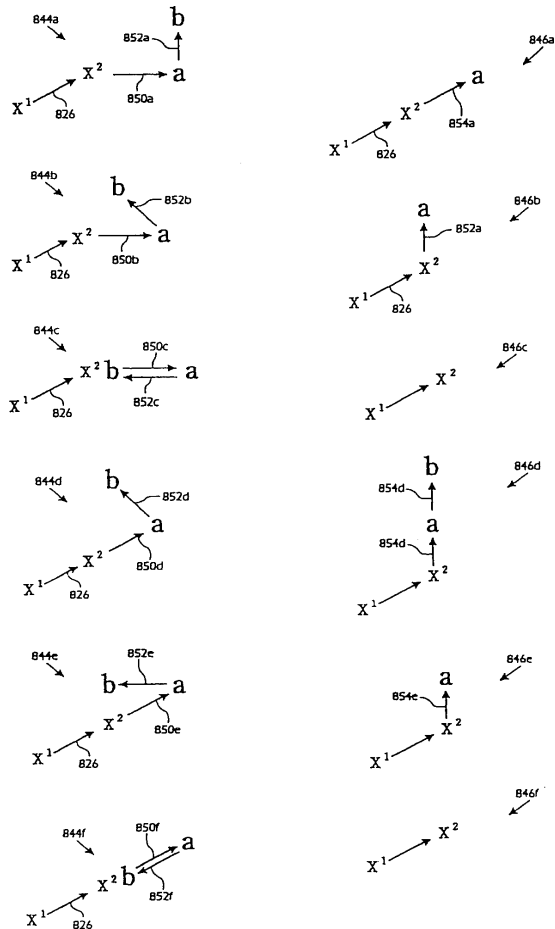
【図 24 B】



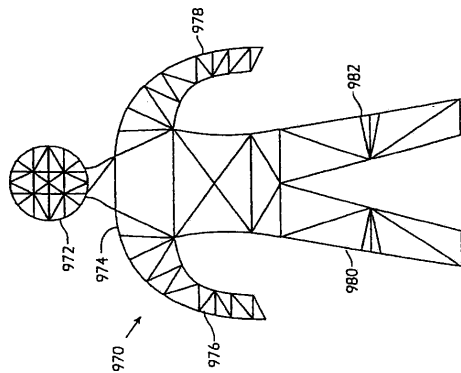
【図 25 B】



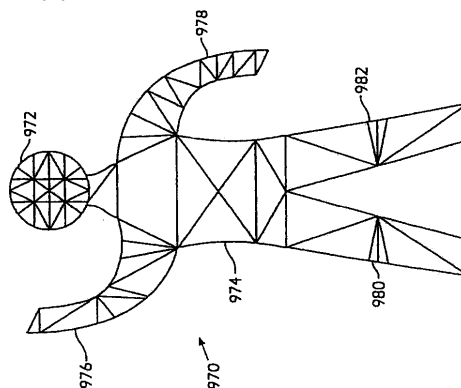
【図 25 C】



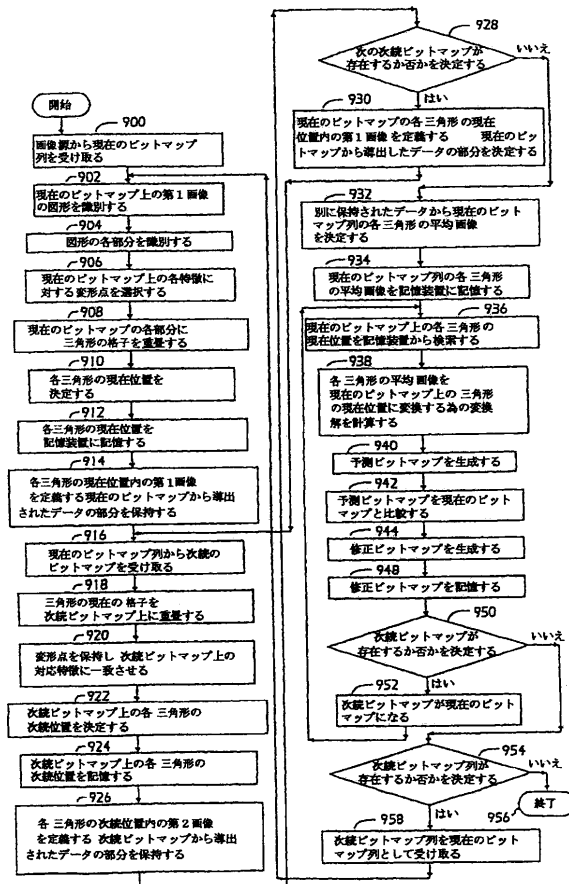
【図 27 A】



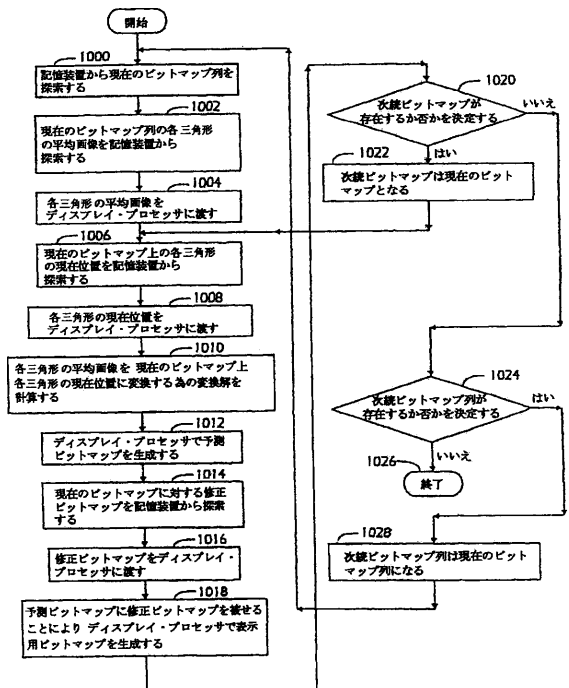
【図 27 B】



【図 26】



【図 28】



フロントページの続き

(74)代理人

弁理士 梅本 政夫

(72)発明者 リー ミン - チー

アメリカ合衆国 ワシントン州 98006 ベルビュー エスイー ワンハンドレッドシックス
ティシックスス プレイス 5588

(72)発明者 チェン ウェイ - ゲ

アメリカ合衆国 ワシントン州 98121 シアトル ファースト アヴェニュー 2900
エス - 509

審査官 國分 直樹

(56)参考文献 特開平03 - 089792 (JP, A)

(58)調査した分野(Int.Cl., DB名)

H04N 7/24 - 7/68

H04N 1/41 - 1/419