

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第4306256号  
(P4306256)

(45) 発行日 平成21年7月29日(2009.7.29)

(24) 登録日 平成21年5月15日(2009.5.15)

(51) Int.Cl.

F 1

GO6T	7/20	(2006.01)
HO4N	5/232	(2006.01)
HO4N	7/32	(2006.01)

GO6T	7/20	B
HO4N	5/232	C
HO4N	5/232	Z
HO4N	7/137	Z

請求項の数 7 (全 29 頁)

(21) 出願番号

特願2003-8377 (P2003-8377)

(22) 出願日

平成15年1月16日 (2003.1.16)

(65) 公開番号

特開2004-220419 (P2004-220419A)

(43) 公開日

平成16年8月5日 (2004.8.5)

審査請求日

平成17年12月20日 (2005.12.20)

(73) 特許権者 000002185

ソニー株式会社

東京都港区港南1丁目7番1号

(74) 代理人 100086841

弁理士 脇 篤夫

(74) 代理人 100114122

弁理士 鈴木 伸夫

(72) 発明者 佐藤 浩

東京都品川区北品川6丁目7番35号 ソ  
ニー株式会社内

(72) 発明者 近藤 哲二郎

東京都品川区北品川6丁目7番35号 ソ  
ニー株式会社内

審査官 松尾 俊介

最終頁に続く

(54) 【発明の名称】動きベクトル検出装置、動きベクトル検出方法、プログラム、記録媒体

## (57) 【特許請求の範囲】

## 【請求項 1】

動画映像データの過去フレーム内の或る画素を代表点として設定する代表点設定手段と、

現在フレームの画素データ値と、上記代表点の画素データ値を比較して、比較結果に基づいて上記代表点との対応が推定される候補点としての画素データを判定する比較判定手段と、

上記候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定する線分判定手段と、

上記線分判定手段で最大線分と判定された候補線分の中点を算出する中点算出手段と、  
上記代表点を始点とし、上記中点を終点として動きベクトルを検出するベクトル検出手段と、

を備えた動きベクトル検出装置。

## 【請求項 2】

上記比較判定手段は、現在フレーム内で、上記代表点に相当する画素位置を中心とした所定画素範囲を探索領域とし、該探索領域内の各画素データについて、上記代表点の画素データ値との比較を行う請求項1に記載の動きベクトル検出装置。

## 【請求項 3】

上記代表点設定手段は、上記中点算出手段で算出された中点に対応する現在フレームの画素を、次のフレームに対する代表点とする請求項1に記載の動きベクトル検出装置。

**【請求項 4】**

動画映像データの過去フレーム内の複数の代表点を設定し、上記各代表点を用いて動きベクトルを検出するために、

上記比較判定手段、上記線分判定手段、上記中点算出手段、及び上記ベクトル検出手段が、上記複数の代表点のそれぞれに対応するように設けられる請求項1に記載の動きベクトル検出装置。

**【請求項 5】**

動画映像データの過去フレーム内の或る画素を代表点として設定する代表点設定手段と、現在フレームの画素データ値と、上記代表点の画素データ値を比較して、比較結果に基づいて上記代表点との対応が推定される候補点としての画素データを判定する比較判定手段と、上記候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定する線分判定手段と、上記線分判定手段で最大線分と判定された候補線分の中点を算出する中点算出手段と、上記代表点を始点とし、上記中点を終点として動きベクトルを検出するベクトル検出手段とを備えた動きベクトル検出装置における動きベクトル検出方法において、

動画映像データの過去フレーム内の或る画素を代表点として設定する代表点設定ステップと、

現在フレームの画素データ値と、上記代表点の画素データ値を比較して、比較結果に基づいて上記代表点との対応が推定される候補点としての画素データを判定する比較判定ステップと、

上記候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定する線分判定ステップと、

上記線分判定ステップで最大線分と判定された候補線分の中点を算出する中点算出手段と、

上記代表点を始点とし、上記中点を終点として動きベクトルを検出するベクトル検出手段と、

を実行する動きベクトル検出方法。

**【請求項 6】**

動画映像データの過去フレーム内の或る画素を代表点として設定する代表点設定手段と、現在フレームの画素データ値と、上記代表点の画素データ値を比較して、比較結果に基づいて上記代表点との対応が推定される候補点としての画素データを判定する比較判定手段と、上記候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定する線分判定手段と、上記線分判定手段で最大線分と判定された候補線分の中点を算出する中点算出手段と、上記代表点を始点とし、上記中点を終点として動きベクトルを検出するベクトル検出手段とを備えた動きベクトル検出装置において実行されるプログラムにおいて、

動画映像データの過去フレーム内の或る画素を代表点として設定する代表点設定ステップと、

現在フレームの画素データ値と、上記代表点の画素データ値を比較して、比較結果に基づいて上記代表点との対応が推定される候補点としての画素データを判定する比較判定ステップと、

上記候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定する線分判定ステップと、

上記線分判定ステップで最大線分と判定された候補線分の中点を算出する中点算出手段と、

上記代表点を始点とし、上記中点を終点として動きベクトルを検出するベクトル検出手段と、

を実行させるプログラム。

**【請求項 7】**

動画映像データの過去フレーム内の或る画素を代表点として設定する代表点設定手段と

10

20

30

40

50

、現在フレームの画素データ値と、上記代表点の画素データ値を比較して、比較結果に基づいて上記代表点との対応が推定される候補点としての画素データを判定する比較判定手段と、上記候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定する線分判定手段と、上記線分判定手段で最大線分と判定された候補線分の中点を算出する中点算出手段と、上記代表点を始点とし、上記中点を終点として動きベクトルを検出するベクトル検出手段とを備えた動きベクトル検出装置において実行されるプログラムを記録した記録媒体において、

動画映像データの過去フレーム内の或る画素を代表点として設定する代表点設定ステップと、

現在フレームの画素データ値と、上記代表点の画素データ値を比較して、比較結果に基づいて上記代表点との対応が推定される候補点としての画素データを判定する比較判定ステップと、

上記候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定する線分判定ステップと、

上記線分判定ステップで最大線分と判定された候補線分の中点を算出する中点算出ステップと、

上記代表点を始点とし、上記中点を終点として動きベクトルを検出するベクトル検出ステップと、

を実行させるプログラムを記録した記録媒体。

#### 【発明の詳細な説明】

##### 【0001】

##### 【発明の属する技術分野】

本発明は、動画映像における動きベクトルを検出するための動きベクトル検出装置、動きベクトル検出方法、さらには動きベクトル検出方法を実行させるプログラム、該プログラムを記録した記録媒体に関するものである。

##### 【0002】

##### 【従来の技術】

動画映像データにかかる処理として、映像の動きベクトルを検出する方法が各種提案され、例えばビデオカメラにおける被写体追尾や手ブレ補正などのために利用されている。

例えばロックマッチング法、勾配法、代表点マッチング法などが知られている。先行技術文献としては次のものが挙げられる。

##### 【0003】

##### 【特許文献1】

特開平5-30495号公報

##### 【特許文献2】

特開平8-237536号公報

##### 【特許文献3】

特開平6-86149号公報

##### 【0004】

ロックマッチング法では、過去フレーム内において或る代表ブロックを設定する。例えば水平・垂直方向の $n \times n$ の画素範囲として代表ブロックを設定する。そして次フレーム(現在フレーム)で、上記代表ブロックに相当する画素範囲を探し、水平・垂直空間上で、代表ブロックの位置から、対応するブロックに至る方向及び距離を動きベクトルとする方式である。

この場合、例えば代表ブロックの画素値総計を基準に、画素値総計がほぼ同じである $n \times n$ のブロックを、対応するブロックと判断する。

##### 【0005】

勾配法は、画素データの輝度勾配を利用して過去フレームと現在フレームでの対応ポイントを判別し、動きベクトルを検出する。例えば上記特許文献1に勾配法を利用した動きベクトル検出装置が示されている。

10

20

30

40

50

**【0006】**

代表点マッチング法については、例えば上記特許文献2,3などに記載されている。この代表点マッチング法を、図15により説明する。図15(a)(b)は、時間的に連続するフレームとして、過去フレームF#1と現在フレームF#2を示している。図15(a)(b)のように、過去フレームF#1に注目領域ARaを設定し、現在フレームF#2には探索領域ARSを設定する。注目領域ARaと探索領域ARSは大きさと位置が同じものとする。

また、注目領域ARaの中心にある画素を代表点rとする。注目領域ARaと代表点rの関係を拡大して図15(c)に示している。

**【0007】**

このように代表点rが決められた後、現在フレームF#2において、探索領域ARSにある全ての画素と代表点rの画素との、輝度値同士の差分絶対値を計算する。そして、その差分絶対値のうち最も小さい差分絶対値を与える画素をその探索領域ARSにおける候補点eとする。

候補点eの数が1つであればその候補点eを終点、代表点rを始点とするベクトルを、求める動きベクトルとする。但し実際には候補点eの数は1つとは限らない。例えば図15(d)では、3つの候補点e1, e2, e3が検出された場合を示している。このような場合には、探索領域ARSの中心(=代表点r相当位置)に最も近い候補点をベクトルの終点とするのが一般的である。従って図15(d)の場合、図15(e)のように候補点e2が採用され、図示するように候補点e2を終点として動きベクトルVが求められる。

**【0008】****【発明が解決しようとする課題】**

上記各方法では、次のような問題がある。

まずブロックマッチング法の場合、動きベクトルの検出精度は高いが、ブロック単位で対応する位置を検出しなければならないため、演算処理量やそのためのデータ記憶量が多く、実現するための装置構成の規模が大きくなる。

また、画像の拡大や縮小、回転運動があった場合に不利である。さらにブロックが画像の複数のオブジェクトにまたがる場合や、例えば背景とオブジェクトにまたがる場合など、間違ったベクトルを検出する可能性が高い。

**【0009】**

勾配法では、画素値が急激に変化するところでは検出誤差が大きくなる。またノイズ成分によって輝度勾配の値に影響を受けやすい。

さらに、大きな動きを検出することが困難であり、大きな動きを検出しようとハードウェアの大規模化が必要になることが知られている。

**【0010】**

代表点マッチング法の場合は、ブロックマッチング法と比べてハードウェア構成の規模を小さくできるという利点があるが、代表点に対応する候補点を、画素差分が最小の画素とするだけでは物理的な意味に乏しい。特に画像内容として画素値が平坦な部分では、候補点が数多く検出されることになり、どの画素をベクトルの終点とするかの判定は困難である。上記のように、例えば代表点に最も近い候補点をベクトル終点とするが、これは明確な物理的な意味を持つものではない。

なお一般的には、画像の急峻なエッヂにおいては微妙なサンプリング位置の違いにより画素値が大きく変動する。そのため、画素差分を元にしたマッチングは検出誤差の要因となりやすい。

**【0011】**

つまり、従来の動きベクトル検出方法では、検出精度を高めるにはハードウェア規模を大きくすることが必要となり、逆にハードウェア規模を小さくするには検出精度を犠牲にする必要があった。

**【0012】****【課題を解決するための手段】**

10

20

30

40

50

そこで本発明は、ハードウエアの大規模化を招かず、精度の良い動きベクトル検出を実現することを目的とする。

【0017】

本発明の動きベクトル検出装置は、動画映像データの過去フレーム内の或る画素を代表点として設定する代表点設定手段と、現在フレームの画素データ値と上記代表点の画素データ値とを比較して、比較結果に基づいて上記代表点との対応が推定される候補点としての画素データを判定する比較判定手段と、上記候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定する線分判定手段と、上記線分判定手段で最大線分と判定された候補線分の中点を算出する中点算出手段と、上記代表点を始点とし、上記中点を終点として動きベクトルを検出するベクトル検出手段とを備える。

10

【0018】

また本発明の動きベクトル検出方法は、動画映像データの過去フレーム内の或る画素を代表点として設定する代表点設定ステップと、現在フレームの画素データ値と、上記代表点の画素データ値を比較して、比較結果に基づいて上記代表点との対応が推定される候補点としての画素データを判定する比較判定ステップと、上記候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定する線分判定ステップと、上記線分判定ステップで最大線分と判定された候補線分の中点を算出する中点算出ステップと、上記代表点を始点とし、上記中点を終点として動きベクトルを検出するベクトル検出手段とを備える。

【0019】

20

これらの装置及び方法において、上記比較判定手段（比較判定ステップ）は、現在フレーム内で、上記代表点に相当する画素位置を中心とした所定画素範囲を探索領域とし、該探索領域内の各画素データについて、上記代表点の画素データ値との比較を行う。

また上記代表点設定手段（代表点設定ステップ）は、上記中点算出手段で算出された中点に対応する現在フレームの画素を、次のフレームに対する代表点とする。

また上記動きベクトル検出装置では、動画映像データの過去フレーム内での複数の代表点を設定し、上記各代表点を用いて動きベクトルを検出するために、上記比較判定手段、上記線分判定手段、上記中点算出手段、及び上記ベクトル検出手段が、上記複数の代表点のそれぞれに対応するように設けられる。

【0020】

30

また本発明のプログラムは上記動きベクトル検出方法を演算装置によって実行させるプログラムであり、本発明の記録媒体は、そのようなプログラムを記録した記録媒体である。

【0021】

以上のような本発明では、動きベクトル検出として、従来方式に比べて小規模のハードウェアで精度の良い検出を行うことのできる2つの新規なアルゴリズムとして「重心検出」と「線分検出」を提供するものである。

「重心検出」は、過去フレームにおける代表点に対してこの代表点の画素値に近い画素を候補点として検出し、その検出された候補点が空間的に連続する候補領域の重心点を検出する。そして代表点を始点とし、重心点を終点として動きベクトルを検出するものである。この場合、「線分検出」に比べればハードウェア規模が大きくなるが、精度の高い動きベクトル検出が可能となる。

40

「線分検出」は、過去フレームにおける代表点に対してこの代表点の画素値に近い画素を候補点として検出し、候補点が走査方向（水平方向）に最も長く連続した候補線分の中点を検出する。そして代表点を始点とし、中点を終点として動きベクトルを検出するものである。この場合、「重心検出」に検出精度がやや落ちるが、ハードウェア規模は小さくなる。

【0022】

【発明の実施の形態】

以下、本発明の実施の形態として、重心検出及び線分検出としての動きベクトル検出について説明していく。説明は次の順序で行う。

50

1 . 重心検出による動きベクトル検出	
1 - 1 重心検出による動きベクトル検出方式	
1 - 2 重心検出にかかる第1の装置構成例及び処理	
1 - 3 重心検出にかかる第2の装置構成例及び処理	
2 . 線分検出による動きベクトル検出	
2 - 1 線分検出による動きベクトル検出方式	
2 - 2 線分検出にかかる第1の装置構成例及び処理	
2 - 3 線分検出にかかる第2の装置構成例及び処理	
3 . 代表点設定方式例	
4 . 動きベクトル検出情報の用途	10
5 . プログラム及び記録媒体	

## 【0023】

1 . 重心検出による動きベクトル検出	
1 - 1 重心検出による動きベクトル検出方式	

図1, 図2により、重心検出による動きベクトル検出方式を説明する。

図1(a)(b)は、時間的に連続するフレームとして、過去フレームF#1と現在フレームF#2を示している。

図1(a)(b)のように、過去フレームF#1に注目領域ARaを設定し、現在フレームF#2には探索領域ARSを設定する。注目領域ARaと探索領域ARSは大きさと位置が同じものとする。

また、注目領域ARaの中心にある画素を代表点rとする。注目領域ARaと代表点rの関係を拡大して図1(c)に示している。 は画素を示し、そのうちで代表点rとされた画素を で示している。

なお、ここではあくまで説明上の例として水平/垂直方向に7画素×7画素の領域を注目領域ARaとしているが、この場合、探索領域ARSも7画素×7画素の領域となる。但し実際には、注目領域ARa及び探索領域ARSは、7画素×7画素の領域とされるとは限らず、例えば実際には代表点rとしての画素を中心として水平・垂直方向に(±32画素)×(±32画素)などとされる。

## 【0024】

なお、代表点rの設定方式としては各種あるが、それについては後述する。ここでは、任意の注目領域ARaの中心画素を、最初の代表点rとするとして説明を続ける。

## 【0025】

このように代表点rが決められた後、現在フレームF#2において、探索領域ARSにある全ての画素と代表点rの画素との、輝度値同士の差分絶対値を計算する。そして求めた差分絶対値がある一定の閾値よりも小さければ、その画素を候補点eとする比較判定処理を行う。

図1(d)には、探索領域ARS内の全ての画素について、代表点rの値との比較判定を行い、候補点eを調べた結果を示す。ここでは の画素として示すように候補点eが検出されたとしている。

## 【0026】

ここで、候補点rが空間的(水平・垂直方向)に連続した領域を候補領域eAとする。図1(d)のように候補点eが検出された場合、探索領域ARS内には、図2(a)のように2つの候補領域eA1、eA2が存在することになる。

このように複数の候補領域eAが認められた場合は、その中で最大の候補領域を選択する。図2(a)の場合、候補領域eA1が選択される。なお、候補領域eAが1つであった場合は、当然、その候補領域が選択される。

そして、選択された候補領域eA1について、重心点を求める。図2(b)に重心点wを示す。

そして重心点wを求めたら、その重心点wを動きベクトルの終点とする。すなわち、求める動きベクトルは代表点rを始点とし重心点wを終点とするベクトルである。図2(c)

10

20

30

40

50

に拡大して示すように、代表点  $r$  に相当する画素位置を  $e_x$  とすると、この画素位置  $e_x$  重心点  $w$  のベクトルが、動きベクトル  $V$  となる。

#### 【0027】

本実施の形態の重心検出による動きベクトル検出は、以上のように候補領域  $e_A$  内の重心をベクトル終点とするようにして行われる。

なお、図1のフレーム  $F \# 2$  に続く、図示していないフレーム  $F \# 3$  の画素データが入力される期間は、フレーム  $F \# 2$  が過去フレームとなる。その場合、代表点  $r$  は、例えば求めた重心点  $w$  に基づいて設定される。例えば重心点  $w$  に最も近い画素として、図2(c)の画素位置  $e_y$  を代表点  $r$  とし、その代表点  $r$  を中心として  $n \times n$  の領域が注目領域  $A_R a$  とされる。

10

そしてフレーム  $F \# 3$  においては、注目領域  $A_R a$  と同一位置が探索領域  $A_R s$  とされ、上記同様に動きベクトル検出が行われる。

#### 【0028】

##### 1 - 2 重心検出にかかる第1の装置構成例及び処理

上記図1、図2で説明した重心検出方式を用いる動きベクトル検出装置の構成を図3に示す。

図3に示すように、動きベクトル検出装置は、判定部1、フレームメモリ2A、2B、アドレス生成部、スイッチSW1～SW4を備える。

#### 【0029】

フレームメモリ2A、2Bは、動画映像の各フレームを形成する画素データ  $D_g$  を記憶するメモリであり、1フレーム毎に交互に順番に用いられる。例えば偶数番目のフレームはフレームメモリ2Aに記憶され、奇数番目のフレームはフレームメモリ2Bに記憶される。

20

スイッチSW1～SW4は、例えば入力画素データ  $D_g$  として偶数番目のフレームメモリ2Aが入力される期間は、それぞれ図示するように  $t_1$  端子が選択される。そして次のフレーム期間では  $t_2$  端子が接続されるというように、フレーム期間単位で切り替えられる。

#### 【0030】

スイッチSW1～SW4が  $t_1$  端子に接続されているフレーム期間においては、入力画素データ  $D_g$  は、スイッチSW1を介してフレームメモリ2Aに供給される。このとき、クロックCKに基づいてアドレス生成部3でアドレスADが生成される。即ち水平、垂直方向の画素位置を示すrow(ロウ)/column(カラム)の値による画素アドレスである。

30

このアドレスADは、スイッチSW2を介して入力画素データ  $D_g$  に対応するアドレスとしてフレームメモリ2Aに供給される。フレームメモリ2AではアドレスADに基づいて入力画素データ  $D_g$  を記憶する。

#### 【0031】

また、入力画素データ  $D_g$  及びアドレスADは、判定部1にも入力される。判定部1は、フレームを構成する入力画素データ  $D_g$  について上述した重心検出の処理を行って、前フレームの代表点を基に動きベクトルを検出する。

40

#### 【0032】

判定部1は、図示するように代表点情報保持部11、比較判定部12、閾値保持部13、フラグメモリ14、重心計算部15、動きベクトル計算部16、動きベクトルメモリ17を有する。

なお、これらの各部は、ソフトウェア、ハードウェアのいずれでも実現できる。例えばハードウェアで構成する場合、代表点情報保持部11、閾値保持部13、フラグメモリ14、動きベクトルメモリ17としてのレジスタやメモリを備え、また比較判定部12、重心計算部15、動きベクトル計算部16としての演算回路が形成されるようすればよい。

また判定部1をDSPやCPUなどの演算処理デバイスにより構成し、内部RAMやレジスタとして、代表点情報保持部11、閾値保持部13、フラグメモリ14、動きベクトル

50

メモリ 17 を実現する共に、ソフトウェア実装により比較判定部 12，重心計算部 15，動きベクトル計算部 16 としての演算機能が実現されるようにしてもよい。

#### 【 0 0 3 3 】

代表点情報保持部 11 は、現在の入力画素データ Dg のフレームの前のフレームにおいて設定された代表点 r の情報として、代表点 r の画素値及びアドレスを保持する。上記のように入力画素データ Dg がフレームメモリ 2A に記憶されていくフレーム期間においては、前のフレームの各画素データはフレームメモリ 2B に記憶されている。つまり代表点 r としての画素データは、フレームメモリ 2B に記憶されている。このため代表点 r の情報としての画素値及びアドレスはスイッチ SW3 を介してフレームメモリ 2B から取り込まれ、代表点情報保持部 11 に保持される。

10

閾値保持部 13 は、画素差分値としての閾値 th を保持する。閾値 th は予め設定されている。

#### 【 0 0 3 4 】

比較判定部 12 には、入力画素データ Dg、及びアドレス AD が入力される。また閾値保持部 13 から閾値 th が供給される。さらに代表点情報保持部 11 から代表点 r の画素値 rg と代表点 r のアドレス rAD が入力される。

この比較判定部 12 は、入力されてくる入力画素データ Dg が図 1 ( b ) に示した探索領域 ARs 内であるか否かの判断、及び探索領域 ARs 内の入力画素データ Dg の画素値（輝度レベル）について、代表点 r の画素値（輝度レベル） rg との比較処理を行い、その結果によりフラグメモリ 14 の判定フラグ制御を行う。

20

#### 【 0 0 3 5 】

順次供給されてくる入力画素データ Dg とそのアドレス AD が探索領域 ARs 内であるか否かは、アドレス AD を探索領域 ARs としてのアドレスと比較することで行う。

探索領域 ARs は、図 1 で説明したように前フレームの代表点 r を中心とする注目領域 ARa と同一のアドレス範囲である。例えば図 1 では簡略化のために代表点 r の画素アドレスを中心として  $7 \times 7$  画素の例を示したが、例えば実際には代表点 r としての画素を中心として水平・垂直方向に ( $\pm 3$  2 画素)  $\times$  ( $\pm 3$  2 画素) などとされる。その場合、比較判定部 12 は、代表点 r の画素アドレス rAD を中心とした水平 6 5 画素  $\times$  垂直 6 5 画素の範囲を、探索領域 ARs として決定することができる。

これにより入力画素データ Dg と共に入力されてくるアドレス AD が、この探索領域 ARs の範囲内のアドレスであるか否かを判別することができる。

30

#### 【 0 0 3 6 】

入力画素データ Dg が探索領域 ARs 内のデータであった場合は、その入力画素データ Dg と代表点 r の画素値 rg の比較処理を行う。具体的には輝度レベルとしての画素差分値を算出する。そして算出された画素差分値の絶対値が閾値 th 未満であるか否かを判定する。

閾値 th 未満であった場合は、その入力画素データ Dg は代表点 r の画素に対応する候補点 e (図 1 ( d ) 参照) とする。その場合、フラグメモリ 14 に対して判定フラグのオンを指示する。

#### 【 0 0 3 7 】

フラグメモリ 14 では、1 フレーム分の各画素にそれぞれ対応したフラグ領域が確保されている。例えば 1 フレームの水平方向画素数が Hn、垂直方向画素数が Vn であり、1 フレームの画素数が  $Hn \times Vn$  であるとすれば、フラグメモリ 14 としては、1 画素に付き 1 ビットとして、( $Hn \times Vn$ ) ビットのフラグ領域が確保される。

40

そして比較判定部 12 からフラグオンの指示があった場合は、その際に供給されているアドレス AD に対応するビット位置に、判定フラグが立てられることになる。従ってフラグメモリ 14 上では、図 1 ( d ) の候補点 e と示したように、判定フラグが立てられることになる。

なお、後述する重心計算のためには、フラグオンとされた画素のアドレスが必要となり、フラグに対応してアドレス AD を記憶する領域が確保されるようにしてもよいが、フラグ

50

位置自体がアドレス A D を表現するのであれば、アドレス値自体を記憶する必要はない。

#### 【 0 0 3 8 】

重心計算部 1 5 は、フラグメモリ 1 4 での判定フラグを参照して、フラグオンのアドレス（候補点 e ）が空間的（水平・垂直方向）に連続する候補領域 e A （図 2 ( a ) 参照）を判別する。また候補領域 e A が複数存在する場合は、その中で最大の候補領域 e A を選択する。

そして図 2 ( b ) で説明したように、選択した最大の候補領域 e A について重心点 w を例えれば座標値として算出する。

#### 【 0 0 3 9 】

動きベクトル計算部 1 6 は、重心計算部 1 5 で算出された重心点 w と、代表点 r の画素アドレス r A D に基づいて、動きベクトル V を計算する。つまり図 2 ( c ) で説明したように代表点 r の画素アドレス r A D の座標値を始点とし、重心点 w の座標値を終点とする動きベクトル V が計算される。

算出された動きベクトル V の値、つまりベクトルの方向及び長さを示す値は、動きベクトルメモリ 1 7 に保持され、動きベクトル V を用いて処理を行う所定の処理系に供給される。

#### 【 0 0 4 0 】

また、現在の入力画素データ D g のフレームの次のフレームに対しては、今回のフレーム内で代表点 r が決められる。代表点 r は重心点 w を基準にして設定される。例えば重心点 w に相当する画素位置、例えば図 1 ( c ) の例では画素位置 e y が、代表点 r とされる。重心計算部 1 5 はこのように、算出した重心点 w に基づいて次の代表点 r としての画素アドレス r A D ' を設定し、スイッチ S W 4 を介してフレームメモリ 2 A に供給する。フレームメモリ 2 A では、画素アドレス r A D ' としての画素に関する付加情報として、代表点 r としての情報が記憶される。

#### 【 0 0 4 1 】

そして次のフレームの入力画素データ D g が入力されるフレーム期間に至ると、各スイッチ S W 1 ~ S W 4 は t 2 端子に切り替えられる。

従って、各入力画素データ D g は、今度はフレームメモリ 2 B に記憶されていく。

このフレーム期間においては、フレームメモリ 2 A から代表点 r としての情報が読み出され、代表点情報保持部 1 1 に保持される。

そして上記同様に、判定部 1 内の処理で重心検出方式での動きベクトル V の検出が行われる。

また、そのフレーム期間において重心点 w が計算されたら、重心計算部 1 5 ではそれに基づいて次の代表点 r の画素アドレスが設定される。その場合は、次の代表点 r としての画素アドレス r A D ' は、スイッチ S W 4 を介してフレームメモリ 2 B に供給され、画素アドレス r A D ' としての画素に関する付加情報として、代表点 r としての情報が記憶される。

#### 【 0 0 4 2 】

以降、同様に、入力画素データ D g がフレームメモリ 2 A に記憶されていくと共に、判定部 1 でフレームメモリ 2 B からの代表点 r の情報を用いて今回のフレームについて動きベクトル V が計算される処理と、入力画素データ D g がフレームメモリ 2 B に記憶されていくと共に、判定部 1 でフレームメモリ 2 A からの代表点 r の情報を用いて今回のフレームについて動きベクトル V が計算される処理とが、交互に行われていく。

#### 【 0 0 4 3 】

図 4 に判定部 1 の処理をフローチャートで示す。なお、このフローチャートは説明上の都合により、1 フレーム期間内の処理として示している。つまり、各フレームにおける画素データ D g が入力されてくる期間毎に、図 4 の処理が行われるものである。

#### 【 0 0 4 4 】

まずステップ F 1 0 1 として代表点情報が設定される。上述したように、代表点情報は前フレーム上で設定された代表点 r としての画素値 r g 及びアドレス r A D であり、フレー

10

20

30

40

50

ムメモリ(2B又は2A)から読み込まれて代表点情報保持部11にセットされる。なお、代表点rの画素は、前フレームにおける重心点wに基づいて設定されたとしたが、動きベクトル検出を行おうとするときの、一番最初のフレームでは、当然ながら動きベクトル検出は行われず、従って一番最初のフレーム上では任意に注目領域ARa及びその中心画素としての代表点rが設定されるものとなる。

#### 【0045】

ステップF102として入力画素データDgが比較判定部12に入力される毎に、ステップF103以降の処理が行われる。

まずステップF103では、比較判定部12が入力された画素データDgについてのアドレスADを参照し、その入力画素データDgが探索領域ARS内であるか否かを判断する。探索領域ARS内でなければ、ステップF102に戻って次の入力画素データDgを待機する。

#### 【0046】

探索領域ARS内としての画素データDgが入力された場合、比較判定部12の処理としてステップF104で画素差分計算が行われる。つまり当該入力画素データDgの画素値と、代表点rの画素値の差分絶対値が求められる。

そしてステップF105では、画素差分絶対値と閾値thが比較され、画素差分絶対値が閾値th未満であればステップF106で、フラグメモリ14上で当該画素データDgについて判定フラグがオンとされる。画素差分絶対値が閾値th以上の場合は、判定フラグはオンとされない。

#### 【0047】

ステップF107では、上記比較判定を行った画素データDgが、探索領域ARS内の最後の画素であるか否かを判別する。即ち、探索領域ARS内の全ての画素についてステップF104、F105、F106の処理を終えたか否かを判別する。まだ終えていなければ、ステップF102に戻って次の画素データ入力に備える。そして画素データDgが入力されたら、同様に探索領域ARSであるか否かを判断し、探索領域ARS内であればステップF104、F105、F106の処理を行う。

#### 【0048】

今回のフレーム内で、少なくとも探索領域ARS内の全ての画素データDgについて比較判定処理を行ったら、処理はステップF108に進む。

この時点では、フラグメモリ14上では、例えば図1(d)のように探索領域ARS内の候補点eの情報(アドレス)が判定フラグとして保持されている。そこで、ステップF108では、重心計算部15は、フラグメモリ14上の判定フラグを参照して、重心計算を行う。つまり、まず最大の候補領域eAを選択し、その候補領域eA内の重心点wを算出する。

#### 【0049】

重心点wは例えば次のように計算する。

図5に判定フラグとして候補点eがみつけられた探索領域ARSを示し、そのx、y座標軸を示しているが、重心のx座標、y座標は、例えば

$$x\text{座標} = \{(\text{判定フラグオンの画素の}x\text{座標})\} / (\text{フラグオンの画素数})$$

$$y\text{座標} = \{(\text{判定フラグオンの画素の}y\text{座標})\} / (\text{フラグオンの画素数})$$

として算出する。算出されたx座標、y座標が、重心点wの座標値となる。

#### 【0050】

次にステップF109では、算出した重心点wに基づいて、次のフレームのための、現フレーム上の代表点rを設定する。即ち上述したように、代表点rの画素アドレスrAD'を設定して、現フレームの画素データDgを記憶しているフレームメモリ(2A又は2B)に記憶させる。

#### 【0051】

ステップF110では、代表点rの画素アドレスrADの座標値と、算出された重心点wを用いて、動きベクトル計算部16で動きベクトルVが算出される。そしてステップF1

10

20

30

40

50

11で、算出された動きベクトルVが動きベクトルメモリ17に記憶され、所要の回路系によって参照される状態となる。

#### 【0052】

この図4の処理が各フレーム期間毎に実行され、各フレーム期間毎に動きベクトルVが検出される。

つまり本例の重心検出による動きベクトル検出は、過去フレームにおける代表点rに対してこの代表点の画素値に近い画素を候補点eとして検出し、その検出された候補点eが空間的に連続する候補領域eAの重心点wを検出する。そして代表点rを始点とし、重心点wを終点として動きベクトルを検出する。

そしてこの重心検出方式によれば、フレーム間の単なる点マッチングではなく、候補領域eAの重心点として、物理的な意味のあるベクトル終点を検出するものであるため、精度の高い動きベクトルを検出できる。10

また、重心点を検出することは、画像の不安定なエッヂ部分やノイズによる影響を受けにくいものともなり、その意味でも精度向上が実現される。

さらにハードウェア規模は従来の代表点マッチング方式とほぼ同等でよく、ブロックマッチング法の場合に比べて小規模化が実現される。

つまり代表点マッチング方式と同程度のハードウェア構成で、より高精度の動きベクトル検出が可能となる。

#### 【0053】

また、重心計算部15は、候補領域eAが複数存在する場合は、最大の候補領域eAを選択し、選択した候補領域eAについて重心点wを算出することで、より正確なベクトル検出が可能となる。20

また、重心点wに基づいて、次のフレームに対する代表点rを設定することで、代表点rとして物理的意味が与えられ、動きベクトル検出の精度を上げることができる。

#### 【0054】

##### 1 - 3 重心検出にかかる第2の装置構成例及び処理

続いて同じく重心検出方式での第2の装置構成例を説明する。

上記例は、重心検出方式での基本的な構成として、代表点rを1つ設定し、代表点rを含む注目領域ARaに対応する探索領域ARS内で画素比較判定を行うものとした。

実際上の適用例としては、代表点rを1つとする場合だけでなく、1フレーム内に複数の代表点rを設定し、各代表点rを含む注目領域ARaに対応して、各探索領域ARS内で画素比較判定を行って重心検出による動きベクトル検出を行うことも考えられる。30

#### 【0055】

例えば図6(a)は、フレームF#1内に4つの代表点r1～r4(注目領域ARa1～ARa4)を設定した状態を示しており、この場合、探索領域ARSは、図6(b)のようにフレームF#2において、探索領域ARS1～ARS4の4つが設定されることになる。

例えばこのように4つの代表点r1～r4を設定する場合の構成例を図7に示す。なお、図3と同一部分には同一符号を付し、説明を省略する。

図7において、構成上、上記図3と異なる点は、判定部1A～1Dとして、判定部1が4つ(代表点の設定数と同数)設けられること、及び判定部選択部4が設けられることである。40

#### 【0056】

判定部1A～1Dは、それぞれ図3の判定部1と同一の構成を有している。つまり、それが代表点情報保持部11、比較判定部12、閾値保持部13、フラグメモリ14、重心計算部15、動きベクトル計算部16、動きベクトルメモリ17を有する。

そして例えば判定部1Aは探索領域ARS1を担当し、判定部1Bは探索領域ARS2を担当し、判定部1Cは探索領域ARS3を担当し、判定部1Dは探索領域ARS4を担当する。従って、各探索領域ARS1～ARS4毎に動きベクトルVが検出されることになる。

**【 0 0 5 7 】**

各判定部 1 A ~ 1 D が、それぞれソフトウェア、ハードウェアのいずれでも実現できることは同様である。

また各判定部 1 A ~ 1 D は、それぞれ独立の回路系（演算処理系）として設けられても良いが、例えば DSP、CPU などの 1 つの演算処理デバイスにより、ここでいう複数の判定部 1 A ~ 1 D を実現することも可能である。

**【 0 0 5 8 】**

このような構成の場合、判定部選択部 4 には、各判定部 1 A ~ 1 D がそれぞれ設定する探索領域 ARs1 ~ ARs4 の情報（アドレス範囲）がセットされる。そして、判定部選択部 4 は、入力画素データ Dg と同期して発生されるアドレス AD について、探索領域 ARs1 ~ ARs4 のうちのどれかに含まれるアドレスであるかを判別する。そして或る探索領域 ARs 内のアドレスと判断された場合は、現在入力されている画素データ Dg について、その探索領域 ARs を担当する判定部（1 A ~ 1 D の内の 1 つ）において処理させる制御を行う。10

**【 0 0 5 9 】**

このような構成例の場合の 1 フレーム期間毎の処理を図 8 に示す。なお、図 4 と同一の処理については同一のステップ番号を付している。

この図 8 の処理は、各判定部 1 A ~ 1 D がそれぞれ実行するものである。

**【 0 0 6 0 】**

各判定部 1 A ~ 1 D は、それぞれステップ F101A で前フレーム上での担当する代表点 r について代表点情報を設定する。即ち各判定部 1 A ~ 1 D はそれぞれ、代表点 r としての画素値 rg 及びアドレス rAD を、フレームメモリ（2B 又は 2A）から読み込んで代表点情報保持部 11 にセットする。20

従って例えば、判定部 1 A の代表点情報保持部 11 には、図 6 (a) の代表点 r1 の情報が、判定部 1 B の代表点情報保持部 11 には、代表点 r2 の情報が、判定部 1 C の代表点情報保持部 11 には、代表点 r3 の情報が、判定部 1 D の代表点情報保持部 11 には代表点 r4 の情報が、それぞれセットされる。

またこのとき、各判定部 1 A ~ 1 D はそれぞれ、代表点 r1 ~ r4 に応じて設定される探索領域 ARs1 ~ ARs4 の情報を、判定部選択部 4 に通知する。

**【 0 0 6 1 】**

各判定部 1 A ~ 1 D は、それぞれステップ F102 として入力画素データ Dg が比較判定部 12 に入力されると、ステップ F103A で、その際に判定部選択部 4 によって選択されたか否かを確認する。30

**【 0 0 6 2 】**

判定部選択部 4 は、入力画素データ Dg のアドレス AD に基づいて判定部 1 A ~ 1 D の 1 つを選択（処理実行指示）する（又はいずれも選択しない）。

つまり、例えば入力画素データ Dg のアドレス AD が探索領域 ARs1 内であった場合は、その入力画素データ Dg に対応して判定部 1 A を選択する。

同様に、入力画素データ Dg のアドレス AD が探索領域 ARs2 内であった場合は、その入力画素データ Dg に対応して判定部 1 B を選択する。40

入力画素データ Dg のアドレス AD が探索領域 ARs3 内であった場合は、その入力画素データ Dg に対応して判定部 1 C を選択する。

入力画素データ Dg のアドレス AD が探索領域 ARs4 内であった場合は、その入力画素データ Dg に対応して判定部 1 D を選択する。

また、入力画素データ Dg のアドレス AD が探索領域 ARs1 ~ ARs4 のいずれにも相当しなかった場合は、その入力画素データ Dg に対応しては、判定部 1 A ~ 1 D のどれも選択しない。

**【 0 0 6 3 】**

つまり、各判定部 1 A ~ 1 D では、それぞれステップ F103A で、判定部選択部 4 によって選択されているか否かにより、その際に入力された画素データ Dg について処理を行50

うか否かを決定する。

選択されていなければ、ステップ F 102 に戻って次の入力画素データ Dg を待機する。判定部選択部 4 によって選択されていれば、当該入力画素データ Dg についてステップ F 104 ~ F 107 の処理、つまり画素差分の算出と閾値 TH との比較による判定フラグ制御を行うことになる。

このステップ F 104 以降の処理は、図 3 と同様であるため重複説明を避けるが、結果として、各判定部 1A ~ 1D では、それぞれが担当する探索領域 ARs1 ~ ARs4 において候補点 e を求め、その後、最大の候補領域 eA から重心点 w を算出し、代表点 r の座標から重心点 w の座標としての動きベクトル V を検出することになる。

#### 【0064】

このようにすることで、1 画面上で、その画像内容の各部の動きベクトル V を検出することができ、これによって多様且つ利用範囲の広い動きベクトル検出が可能となる。

なお、図 7 では判定部選択部 4 を備える構成としたが、各判定部 1A ~ 1D 内に、それぞれ入力画素データ Dg について担当する探索領域 ARs 内であるか否かを判別する機能を備えるようにしてもよい。

また、この例では代表点 r 、探索領域 ARs を 4 つとしたが、これらを 2 つ、3 つ、或いは 5 以上とする例も当然考えられ、そのためには判定部 1 が代表点の数に応じて構成されればよい。

#### 【0065】

##### 2. 線分検出による動きベクトル検出

###### 2 - 1 線分検出による動きベクトル検出方式

次に、図 9 により、線分検出による動きベクトル検出方式を説明する。

図 9 (a) (b) は、時間的に連続するフレームとして、過去フレーム F # 1 と現在フレーム F # 2 を示している。上記重心検出の場合と同様に、過去フレーム F # 1 に注目領域 ARa を設定し、現在フレーム F # 2 には探索領域 ARs を設定する。注目領域 ARa と探索領域 ARs は大きさと位置が同じものとする。また注目領域 ARa の中心にある画素を代表点 r とする。注目領域 ARa と代表点 r の関係を拡大して図 9 (c) に示している。

#### 【0066】

このように代表点 r が決められた後、現在フレーム F # 2 において、探索領域 ARs にある全ての画素と代表点 r の画素との、輝度値同士の差分絶対値を計算する。そして求めた差分絶対値がある一定の閾値よりも小さければ、その画素を候補点 e とする比較判定処理を行う。

図 9 (d) には、探索領域 ARs 内の全ての画素について、代表点 r の値との比較判定を行い、候補点 e を調べた結果を示す。ここでは の画素として示すように候補点 e が検出されたとしている。

なお、k、k + 1 ··· はフレーム上での画素の走査方向（水平方向）を示している。フレームを構成する画素データは、走査方向のストリームデータとして入力されてくるため、例えば 1 フレーム内での k 番目の走査線期間内における一部期間に、図示する探索領域 ARs 内の最初の行の画素が入力されてくる。

#### 【0067】

図 9 (d) のように候補点 e が判別されるが、この結果については、候補点 r が走査方向（水平方向）に連続した領域を候補線分 eL とする。図 9 (d) のように候補点 e が検出された場合、探索領域 ARs 内には、図 9 (c) のように各走査ライン毎にみて候補点 e が存在する部分に、候補線分 eL1、eL2 ··· eL7 が検出されることになる。

そして、このような各候補線分の中で最大（最長）の候補線分が選択される。

例えば図 9 (e) の場合、候補線分 eL2 が選択される。なお、候補線分 eL が 1 つであった場合は、当然、その候補線分が選択される。

そして、選択された候補線分 eL2 について、図 9 (f) に示すように中点 c を求める。。

10

20

30

40

50

中点 c を求めたら、その中点 c を動きベクトルの終点とする。すなわち、求める動きベクトルは代表点 r を始点とし中点 c を終点とするベクトルである。図 9 ( f ) に示すように、代表点 r に相当する画素位置を e x とすると、この画素位置 e x 中点 c のベクトルが、動きベクトル v となる。

#### 【 0 0 6 8 】

本実施の形態の線分検出による動きベクトル検出は、以上のように候補線分 e L 内の中点をベクトル終点とするようにして行われる。

なお、図 9 のフレーム F # 2 に続く、図示していないフレーム F # 3 の画素データが入力される期間は、フレーム F # 2 が過去フレームとなる。その場合、代表点 r は、例えは求めた中点 c に基づいて設定される。例えは中点 c に最も近い画素として、図 9 ( f ) の画素位置 e y 1 又は e y 2 の一方を代表点 r とし、その代表点 r を中心として n × n の領域が注目領域 A R a とされる。  
10

そしてフレーム F # 3 においては、注目領域 A R a と同一位置が探索領域 A R s とされ、上記同様に動きベクトル検出が行われる。

#### 【 0 0 6 9 】

##### 2 - 2 線分検出にかかる第 1 の装置構成例及び処理

上記図 9 で説明した線分検出方式を用いる動きベクトル検出装置の構成を図 10 に示す。図示するように、動きベクトル検出装置は、判定部 2 0 、フレームメモリ 2 A 、 2 B 、アドレス生成部、スイッチ SW 1 ~ SW 4 を備える。

#### 【 0 0 7 0 】

フレームメモリ 2 A 、 2 B は、動画映像の各フレームを形成する画素データ D g を記憶するメモリであり、1 フレーム毎に交互に順番に用いられる。例えは偶数番目のフレームはフレームメモリ 2 A に記憶され、奇数番目のフレームはフレームメモリ 2 B に記憶される。  
20

スイッチ SW 1 ~ SW 4 は、例えは入力画素データ D g として偶数番目のフレームメモリ 2 A が入力される期間は、それぞれ図示するように t 1 端子が選択される。そして次のフレーム期間では t 2 端子が接続されるというように、フレーム期間単位で切り替えられる。  
。

#### 【 0 0 7 1 】

スイッチ SW 1 ~ SW 4 が t 1 端子に接続されているフレーム期間においては、入力画素データ D g は、スイッチ SW 1 を介してフレームメモリ 2 A に供給される。このとき、クロック C K に基づいてアドレス生成部 3 でアドレス A D が生成される。即ち水平、垂直方向の画素位置を示す row ( ロウ ) / column ( カラム ) の値による画素アドレスである。  
30

このアドレス A D は、スイッチ SW 2 を介して入力画素データ D g に対応するアドレスとしてフレームメモリ 2 A に供給される。フレームメモリ 2 A ではアドレス A D に基づいて入力画素データ D g を記憶する。

#### 【 0 0 7 2 】

また、入力画素データ D g 及びアドレス A D は、判定部 2 0 にも入力される。判定部 2 0 は、フレームを構成する入力画素データ D g について上述した線分検出の処理を行って、前フレームの代表点を基に動きベクトルを検出する。  
40

#### 【 0 0 7 3 】

判定部 2 0 は、図示するように代表点情報保持部 2 1 、比較判定部 2 2 、閾値保持部 2 3 、線分判定部 2 4 、動きベクトル計算部 2 5 、動きベクトルメモリ 2 6 を有する。

なお、これらの各部は、ソフトウェア、ハードウェアのいずれでも実現できる。例えはハードウェアで構成する場合、代表点情報保持部 2 1 、閾値保持部 2 3 、後述する線分判定部 2 4 内の記憶部、及び動きベクトルメモリ 2 6 として、必要なレジスタやメモリを備え、また比較判定部 2 2 、線分判定部 2 4 、動きベクトル計算部 2 5 としての演算回路が形成されるようにすればよい。

また判定部 2 0 を D S P や C P U などの演算処理デバイスにより構成し、内部 R A M やレ  
50

ジスタとして、代表点情報保持部 21、閾値保持部 23、後述する線分判定部 24 内の記憶部、及び動きベクトルメモリ 26 として記憶領域を実現する共に、ソフトウェア実装により比較判定部 22、線分判定部 24、動きベクトル計算部 25 としての演算機能が実現されるようにしてもよい。

#### 【0074】

代表点情報保持部 21 は、現在の入力画素データ  $D_g$  のフレームの前のフレームにおいて設定された代表点  $r$  の情報として、代表点  $r$  の画素値及びアドレスを保持する。上記のように入力画素データ  $D_g$  がフレームメモリ 2A に記憶されていくフレーム期間においては、前のフレームの各画素データはフレームメモリ 2B に記憶されている。つまり代表点  $r$  としての画素データは、フレームメモリ 2B に記憶されている。このため代表点  $r$  の情報としての画素値及びアドレスはスイッチ SW3 を介してフレームメモリ 2B から取り込まれ、代表点情報保持部 21 に保持される。  
10

閾値保持部 23 は、画素差分値としての閾値  $t_h$  を保持する。閾値  $t_h$  は予め設定されている。

#### 【0075】

比較判定部 22 には、入力画素データ  $D_g$ 、及びアドレス  $A_D$  が入力される。また閾値保持部 23 から閾値  $t_h$  が供給される。さらに代表点情報保持部 21 から代表点  $r$  の画素値  $r_g$  と代表点  $r$  のアドレス  $r_A D$  が入力される。

この比較判定部 22 は、入力されてくる入力画素データ  $D_g$  が図 9 (b) に示した探索領域  $A_R s$  内であるか否かの判断、及び探索領域  $A_R s$  内の入力画素データ  $D_g$  の画素値（輝度レベル）について、代表点  $r$  の画素値（輝度レベル） $r_g$  との比較処理を行い、その結果を線分判定部 24 に通知する。  
20

#### 【0076】

順次供給されてくる入力画素データ  $D_g$  とそのアドレス  $A_D$  が探索領域  $A_R s$  内であるか否かは、アドレス  $A_D$  を探索領域  $A_R s$  としてのアドレスと比較することで行う。

探索領域  $A_R s$  は、図 9 で説明したように前フレームの代表点  $r$  を中心とする注目領域  $A_R a$  と同一のアドレス範囲である。例えば図 9 では簡略化のために代表点  $r$  の画素アドレスを中心として  $7 \times 7$  画素の例を示したが、例えば実際には代表点  $r$  としての画素を中心として水平・垂直方向に ( $\pm 3$  2 画素)  $\times$  ( $\pm 3$  2 画素) などとされる。従ってこの場合、比較判定部 22 は、代表点  $r$  の画素アドレス  $r_A D$  を中心とした水平 6 5 画素  $\times$  垂直 6 5 画素の範囲を、探索領域  $A_R s$  として決定することができる。  
30

これにより入力画素データ  $D_g$  と共に入力されてくるアドレス  $A_D$  が、この探索領域  $A_R s$  の範囲内のアドレスであるか否かを判別することができる。

#### 【0077】

入力画素データ  $D_g$  が探索領域  $A_R s$  内のデータであった場合は、その入力画素データ  $D_g$  と代表点  $r$  の画素値  $r_g$  の比較処理を行う。具体的には輝度レベルとしての画素差分値を算出する。そして算出された画素差分値の絶対値が閾値  $t_h$  未満であるか否かを判定する。

閾値  $t_h$  未満であった場合は、その入力画素データ  $D_g$  は代表点  $r$  の画素に対応する候補点  $e$  (図 9 (d) 参照) とする。その場合、線分判定部 24 に対して計測フラグのオンを指示する。  
40

#### 【0078】

線分判定部 24 は、計測レジスタ、アドレスレジスタ、最大線分レジスタ、中点座標レジスタ、としての記憶領域を備え、また所定の演算処理を行う機能を有する。

計測レジスタは、計測フラグとして 1 ビットの情報が保持できればよい。

アドレスレジスタは、線分の始点アドレスを記憶するもので、1 つのアドレス  $A_D$  の記憶に必要なビット数であればよい。

最大線分レジスタは、最大長の候補線分  $e_L$  の長さを記憶するレジスタであり、線分長に必要なビット数の記憶領域でよい。線分長は単に画素数やアドレス長の情報でもよい。なお最大の候補線分の始点アドレスと終点アドレスを、線分長の意味で記憶しても良い。  
50

中点座標レジスタは、線分の中点座標を記憶する。座標値を記憶できる記憶領域でよい。

#### 【0079】

この線分判定部24は、比較判定部12からフラグオンの指示があった場合は、計測フラグをオンとする。またその際には、その時点で供給されているアドレスADを、線分の先頭アドレスとしてアドレスレジスタに記憶する。

また、候補線分eLとされた線分のサイズ（線分長）の演算を行い、その値が最大であれば、最大線分レジスタに記憶する。

また最大線分レジスタに記憶された候補線分eLについて中点座標を計算し、中点座標レジスタに記憶する。

#### 【0080】

詳しい説明は図11において述べるが、この線分判定部24からは、探索領域ARS内の全画素データDgについて比較判定部22及び線分判定部24の処理が終えた時点で、最大の候補線分eLについての中点c（中点座標）が中点座標レジスタに記憶されている。動きベクトル計算部25は、線分判定部24において算出された中点cと、代表点rの画素アドレスrADに基づいて、動きベクトルVを計算する。つまり図9(f)で説明したように代表点rの画素アドレスrADの座標値を始点とし、重心点wの座標値を終点とする動きベクトルVが計算される。

算出された動きベクトルVの値、つまりベクトルの方向及び長さを示す値は、動きベクトルメモリ17に保持され、動きベクトルVを用いて処理を行う所定の処理系に供給される。

#### 【0081】

また、現在の入力画素データDgのフレームの次のフレームに対しては、今回のフレーム内で代表点rが決められる。代表点rは中点cを基準にして設定される。例えば中点cに相当する画素位置、例えば図9(f)の例では画素位置ey1又はey2どちらかが、次の代表点rとされる。

線分判定部24はこのように、算出した中点cに基づいて次の代表点rとしての画素アドレスrAD'を設定し、スイッチSW4を介してフレームメモリ2Aに供給する。フレームメモリ2Aでは、画素アドレスrAD'としての画素に関する付加情報として、代表点rとしての情報が記憶される。

#### 【0082】

そして次のフレームの入力画素データDgが入力されるフレーム期間に至ると、各スイッチSW1～SW4はt2端子に切り替えられる。

従って、各入力画素データDgは、今度はフレームメモリ2Bに記憶されていく。

このフレーム期間においては、フレームメモリ2Aから代表点rとしての情報が読み出され、代表点情報保持部21に保持される。

そして上記同様に、判定部20内の処理で中点検出方式での動きベクトルVの検出が行われる。

また、そのフレーム期間において中点cが計算されたら、線分判定部24ではそれに基づいて次の代表点rの画素アドレスが設定される。その場合は、次の代表点rとしての画素アドレスrAD'は、スイッチSW4を介してフレームメモリ2Bに供給され、画素アドレスrAD'としての画素に関する付加情報として、代表点rとしての情報が記憶される。

#### 【0083】

以降、同様に、入力画素データDgがフレームメモリ2Aに記憶されていくと共に、判定部20で、フレームメモリ2Bからの代表点rの情報を用いて、今回のフレームについて動きベクトルVが計算される処理と、入力画素データDgがフレームメモリ2Bに記憶されていくと共に、判定部20で、フレームメモリ2Aからの代表点rの情報を用いて、今回のフレームについて動きベクトルVが計算される処理とが、交互に行われていく。

#### 【0084】

図11に判定部20の処理をフローチャートで示す。なお、このフローチャートは説明上

10

20

30

40

50

の都合により、1フレーム期間内の処理として示している。つまり、各フレームにおける画素データDgが入力されてくる期間毎に、図11の処理が行われる。

#### 【0085】

まずステップF201として代表点情報が設定される。上述したように、代表点情報は前フレーム上で設定された代表点rとしての画素値rg及びアドレスrADであり、フレームメモリ(2B又は2A)から読み込まれて代表点情報保持部21にセットされる。

なお、代表点rの画素は、前フレームにおける中点cに基づいて設定されたとしたが、動きベクトル検出を行おうとするときの、一番最初のフレームでは、当然ながら動きベクトル検出は行われず、従って一番最初のフレーム上では任意に注目領域ARa及びその中心画素としての代表点rが設定されるものとなる。

10

#### 【0086】

ステップF202として入力画素データDgが比較判定部22に入力される毎に、ステップF203以降の処理が行われる。

まずステップF203では、比較判定部22は、入力された画素データDgについてのアドレスADを参照し、その入力画素データDgが探索領域ARS内であるか否かを判断する。探索領域ARS内でなければ、ステップF202に戻って次の入力画素データDgを待機する。

#### 【0087】

探索領域ARS内としての画素データDgが入力された場合、比較判定部22の処理としてステップF204で画素差分計算が行われる。つまり当該入力画素データDgの画素値と、代表点rの画素値の差分絶対値が求められる。

20

そしてステップF205では、画素差分絶対値と閾値thが比較され、画素差分絶対値が閾値th未満であればステップF207に進む。

そして線分判定部24における計測フラグがオンとされているか否かを判別し、オフであったらステップF208で計測フラグをオンとさせる。

線分判定部24は、計測フラグレジスタにおいて計測フラグをオンとするとともに、ステップF209では現在のアドレスADを線分の始点アドレスとしてアドレスレジスタに記憶する。そしてステップF210に進む。

なお、ステップF207の時点で、既に計測フラグがオンとされている場合は、ステップF208,F209の処理を行わずにステップF210に進む。

30

#### 【0088】

ステップF210では、線分判定部24は候補線分eLとしての線分長を算出する。即ち、現在の画素データDgについてのアドレスADを線分終点とし、アドレスレジスタに記憶してあるアドレスを線分始点として、線分終点から線分始点を減算することで線分長を求める。

そしてステップF211では、算出した線分長が、その時点で最大線分レジスタに記憶されている線分長以上であるか否かを判断する。

#### 【0089】

算出した線分長が、その時点での最大線分レジスタに記憶されている線分長以上であった場合は、ステップF212に進んで最大線分レジスタの値を今回算出した値に更新する。そしてステップF213で、最大線分レジスタに記憶した線分の中点座標を、その線分始点アドレスと線分終点アドレスから計算し、ステップF214で中点座標レジスタの値を、計算した値に更新する。そしてステップF215に進む。

40

#### 【0090】

上記ステップF205で、画素差分絶対値と閾値thの比較結果として、画素差分絶対値が閾値th以上である場合、つまり今回入力された画素データDgが候補点eとはならない場合は、ステップF206で計測フラグのリセット処理を行ったうえでステップF215に進む。

また上記ステップF211で、算出された線分長が、最大線分レジスタに記憶されている線分長より短い場合は、最大線分レジスタや中点座標レジスタの更新を行わずにステップ

50

F 2 1 5 に進む。

**【 0 0 9 1 】**

これら各場合においてステップ F 2 1 5 に進んだときは、現在の処理対象となった画素データ D g が、探索領域 A R s 内の最後の画素であるか否かを判別する。即ち、探索領域 A R s 内の全ての画素についてステップ F 2 0 4 以降の処理を終えたか否かを判別する。まだ終えていなければ、ステップ F 2 0 2 に戻って次の画素データ入力に備える。そして画素データ D g が入力されたら、同様に探索領域 A R s であるか否かを判断し、探索領域 A R s 内であればステップ F 2 0 4 以降の処理を行う。

**【 0 0 9 2 】**

そして今回のフレーム内で、少なくとも探索領域 A R s 内の全ての画素データ D g についてステップ F 2 0 4 以降の処理を行ったら、処理はステップ F 2 1 5 から F 2 1 6 に進む。  
10

この時点で、上記ステップ F 2 0 4 ~ F 2 1 4 で説明した処理により、線分判定部 2 4 の中点座標レジスタには、最大長の候補線分 e L についての中点座標が記憶されているものとなる。

**【 0 0 9 3 】**

上記ステップ F 2 0 4 ~ F 2 1 4 の処理の様子を図 1 2 に模式的に示す。

図 1 2 は、図 9 ( d ) のように候補点 e が検出される場合を、拡大して示している。及び はフレーム内の画素(入力画素データ D g )を示し、破線の範囲内が探索領域 A R s 内である。 で示す画素 g 1 ~ g 1 4 は図 9 ( d ) で候補点 e として示した画素としている。  
20

**【 0 0 9 4 】**

例えば走査 k の水平期間において画素データ g 1 が入力された際に、この画素は最初の候補点 e として検出されるが、このとき計測フラグがオンとされ、また画素データ g 1 のアドレス A D が線分始点としてアドレスレジスタに記憶される。

このとき、検出される線分長は、線分始点 = 線分終点となるため画素数でいえば「 1 」である。それ以前に候補線分 e L は検出されていないため、線分長「 1 」が最大線分となつて最大線分レジスタに記憶される。また中点座標が記憶される。

**【 0 0 9 5 】**

次に、候補点 e となる画素データ g 2 が入力された際には、計測フラグは既にオンであるため線分始点としてアドレスレジスタは更新されない。このとき、検出される線分長は、線分始点が画素データ g 1 のアドレス、線分終点が画素データ g 2 のアドレスとなるため、画素数でいえば「 2 」である。その時点で最大線分レジスタの値は「 1 」であり、今回はそれを越えるため最大線分レジスタの値は線分長「 2 」に更新される。また中点座標レジスタの値が画素 g 1 , g 2 による線分の中点座標に更新される。  
30

画素データ g 2 の次の画素データが入力された際に、候補点 e とはならなかったとする、その際に計測フラグはリセットされる。

**【 0 0 9 6 】**

その後、画素データ D g の入力が進み、例えば走査 k + 1 の水平期間において画素データ g 3 が入力された際に、この画素が候補点 e として検出されたとすると、このとき計測フラグがオンとされ、また画素データ g 3 のアドレスが線分始点としてアドレスレジスタに記憶される。  
40

このとき、検出される線分長は、線分始点 = 線分終点となるため画素数でいえば「 1 」である。その時点の最大線分レジスタの値は「 2 」であるため、最大線分レジスタ及び中点座標レジスタは更新されない。

**【 0 0 9 7 】**

次にも候補点 e となる画素データ g 4 が入力された際には、計測フラグは既にオンであるため線分始点としてアドレスレジスタは更新されない。このとき、検出される線分長は、線分始点が画素データ g 3 のアドレス、線分終点が画素データ g 4 のアドレスとなるため、画素数でいえば「 2 」である。その時点で最大線分レジスタの値は「 2 」であり、今回  
50

は最大線分レジスタの値以上となるため最大線分レジスタの値は、画素  $g_3, g_4$  による候補線分  $e_L$  の線分長「2」に更新され、また中点座標レジスタの値が画素  $g_3, g_4$  による線分の中点座標に更新される。

次の画素データ  $g_5$  の入力の際も、またさらに次の画素データ  $g_6$  の入力の際も、同様に線分長が最大値を更新していくため、最大線分レジスタ及び中点座標レジスタの値が更新されていく。画素データ  $g_6$  の入力後の時点で、最大線分レジスタの値は「4」となり、中点座標レジスタの値は画素  $g_3, g_4, g_5, g_6$  の線分の中点座標とされることになる。

画素データ  $g_6$  の次の画素データが入力された際に、これが候補点  $e$  とはならなかったとすると、その際に計測フラグはリセットされる。

10

#### 【0098】

この図12の例の場合は、以降においては、候補点  $e$  となる画素データ ( $g_7 \sim g_{14}$ ) がそれぞれ入力される時点でも、算出される線分長が「4」以上となることはない。従つて処理が図11のステップ F212 ~ F214 に進むことはなく、探索領域 ARs 内の全画素についてステップ F204 以降の処理を終えた時点では、結果的に、最大線分レジスタの値は「4」とされており、中点座標レジスタの値は画素  $g_3, g_4, g_5, g_6$  の線分の中点座標が保持されていることになる。

つまり、図11のステップ F215 で、探索領域 ARs 内の全画素について処理を終えた時点では、図9(e)(f) に示した候補線分  $e_L$  の中点  $c$  の座標が中点座標レジスタに保持されていることになる。

20

#### 【0099】

図11のステップ F216 に進んだら、線分判定部24は、算出した最大の候補線分  $e_L$  の中点  $c$ 、つまり中点座標レジスタの値に基づいて、次のフレームのための、現フレーム上での代表点  $r$  を設定する。即ち上述したように、代表点  $r$  の画素アドレス  $rAD'$  を設定して、現フレームの画素データ  $Dg$  を記憶しているフレームメモリ (2A 又は 2B) に記憶させる。

#### 【0100】

ステップ F217 では、代表点  $r$  の画素アドレス  $rAD$  の座標値と、最大の候補線分  $e_L$  の中点  $c$ 、つまり中点座標レジスタの値を用いて、動きベクトル計算部25で動きベクトル  $V$  が算出される。そしてステップ F218 で、算出された動きベクトル  $V$  が動きベクトルメモリ17に記憶され、所要の回路系によって参照される状態となる。

30

#### 【0101】

この図11の処理が各フレーム期間毎に実行され、各フレーム期間毎に動きベクトル  $V$  が検出される。

つまり本例の線分検出による動きベクトル検出は、過去フレームにおける代表点  $r$  に対してこの代表点の画素値に近い画素を候補点  $e$  として検出し、その検出された候補点  $e$  が走査方向に連続する候補線分  $e_L$  のうちで最大長の候補線分  $e_L$  の中点  $c$  を検出する。そして代表点  $r$  を始点とし、中点  $c$  を終点として動きベクトルを検出する。

そしてこの線分検出方式によれば、フレーム間の単なる点マッチングではなく、最大の候補線分  $e_L$  の中点として、物理的な意味のあるベクトル終点を検出するものであるため、精度の高い動きベクトルを検出できる。

40

また、線分中点を検出することは、画像の不安定なエッヂ部分やノイズによる影響を受けにくいものとなり、その意味でも精度向上が実現される。

また、中点に基づいて、次のフレームに対する代表点を設定することで、代表点として物理的意味が与えられ、動きベクトル検出の精度を上げることができる。

#### 【0102】

また線分判定部24は、候補点  $e$  が走査方向に連続する候補線分  $e_L$  として、最大長となる候補線分を判定し、記憶しておけばよいため、必要な記憶容量が少なく、ハードウェア規模の小型化に好適である。

即ちハードウェア規模、特に処理に必要な記憶領域は従来の代表点マッチング方式よりも

50

小さくでき、その上で上記理由から単なる点同士のマッチングである代表点マッチング方式よりも高精度の検出が可能となる。

#### 【0103】

また、線分検出方式は、上述した重心検出方式よりもハードウエアの軽減を意図したものとも言える。

重心検出方式の場合は、探索領域 A R s 内の全ての画素について差分絶対値を計算しそれが決められた閾値よりも大きいか小さいかを判断した後でないと重心 w の決定ができない。つまり重心 w を決定するまで探索領域 A R s 内のデータ、もしくは候補点 e となりうるか否かというフラグ情報を保持しておく必要がある（図 3 のフラグメモリ）。

一方、線分検出方式においては、求める候補線分 e L の中点を決定するまでにおいて探索領域 A R s 内の全ての画素データ D g のアドレスもしくは候補点であるか否かといった情報を持ち続ける必要がない。

即ち上記図 12 を用いて説明したように、最大線分が検出される毎に中点座標が更新されていくような処理形態をとることで、過去の走査に係る候補点 e や候補線分 e L の情報は、より長い候補線分が検出された時点で破棄される。また、最大線分を越えなかった候補点 e や候補線分 e L の情報は記憶されない。従って、中点座標の検出までに必要な記憶領域は、非常に少なくて済み、それによってハードウエアの軽減が可能になるものである。

#### 【0104】

##### 2 - 3 線分検出にかかる第 2 の装置構成例及び処理

続いて同じく線分検出方式での第 2 の装置構成例を説明する。

上記図 10 の例は、線分検出方式での基本的な構成として、代表点 r を 1 つ設定し、代表点 r を含む注目領域 A R a に対応する探索領域 A R s 内で画素比較判定を行うものとした。

この線分検出方式においても実際上の適用例としては、代表点 r を 1 つとする場合だけでなく、1 フレーム内に複数の代表点 r を設定し、各代表点 r を含む注目領域 A R a に対応して、各探索領域 A R s 内で画素比較判定を行って動きベクトル検出を行うことも考えられる。

#### 【0105】

即ち上述した重心検出方式において図 6 (a) (b) で説明したように、フレーム F # 1 内に 4 つの代表点 r 1 ~ r 4 (注目領域 A R a 1 ~ A R a 4 ) を設定し、これに対応して 4 つの探索領域 A R s 1 ~ A R s 4 を設定するような場合である。例えばこのように 4 つの代表点 r 1 ~ r 4 を設定する場合の構成例を図 13 に示す。なお、図 10 と同一部分には同一符号を付し、説明を省略する。

図 13において、構成上、上記図 10 と異なる点は、判定部 20A ~ 20D として、判定部 20 が 4 つ（代表点の設定数と同数）設けられること、及び判定部選択部 4 が設けされることである。

#### 【0106】

判定部 20A ~ 20D は、それぞれ図 3 の判定部 20 と同一の構成を有している。つまり、それぞれが代表点情報保持部 21、比較判定部 22、閾値保持部 23、線分判定部 24、動きベクトル計算部 25、動きベクトルメモリ 26 を有する。

そして例えば判定部 20A は図 6 (b) の探索領域 A R s 1 を担当し、判定部 20B は探索領域 A R s 2 を担当し、判定部 20C は探索領域 A R s 3 を担当し、判定部 20D は探索領域 A R s 4 を担当する。従って、各探索領域 A R s 1 ~ A R s 4 每に動きベクトル V が検出されることになる。

#### 【0107】

各判定部 20A ~ 20D が、それぞれソフトウェア、ハードウェアのいずれでも実現できることは同様である。

また各判定部 20A ~ 20D は、それぞれ独立の回路系（演算処理系）として設けられても良いが、例えば DSP、CPU などの 1 つの演算処理デバイスにより、ここでいう複数

10

20

30

40

50

の判定部 20A～20D を実現することも可能である。

#### 【0108】

判定部選択部 4 には、各判定部 20A～20D がそれぞれ設定する探索領域 ARs1～ARs4 の情報（アドレス範囲）がセットされる。そして、判定部選択部 4 は、入力画素データ Dg と同期して発生されるアドレス AD について、探索領域 ARs1～ARs4 のうちのどれかに含まれるアドレスであるかを判別し、或る探索領域 ARs 内のアドレスと判断された場合は、現在入力されている画素データ Dg について、その探索領域 ARs を担当する判定部（20A～20D の内の 1 つ）において処理させる制御を行う。

#### 【0109】

このような構成例の場合の 1 フレーム期間毎の処理を図 14 に示す。なお、図 11 と同一の処理については同一のステップ番号を付している。10

この図 14 の処理は、各判定部 20A～20D がそれぞれ実行する。

#### 【0110】

各判定部 20A～20D は、それぞれステップ F201A で前フレーム上での担当する代表点 r について代表点情報を設定する。即ち各判定部 20A～20D はそれぞれ、代表点 r としての画素値 rg 及びアドレス rAD を、フレームメモリ（2B 又は 2A）から読み込んで代表点情報保持部 21 にセットする。

従って例えば、判定部 20A の代表点情報保持部 21 には、図 6（a）の代表点 r1 の情報が、判定部 20B の代表点情報保持部 21 には、代表点 r2 の情報が、判定部 20C の代表点情報保持部 21 には、代表点 r3 の情報が、判定部 20D の代表点情報保持部 21 には代表点 r4 の情報が、それぞれセットされる。またこのとき、各判定部 20A～20D はそれぞれ、代表点 r1～r4 に応じて設定される探索領域 ARs1～ARs4 の情報を、判定部選択部 4 に通知する。20

#### 【0111】

各判定部 20A～20D は、それぞれステップ F202 として入力画素データ Dg が比較判定部 22 に入力されると、ステップ F203A で、その際に判定部選択部 4 によって選択されたか否かを確認する。

#### 【0112】

判定部選択部 4 は、入力画素データ Dg のアドレス AD に基づいて判定部 20A～20D の 1 つを選択（処理実行指示）する（又はいずれも選択しない）。30

つまり、例えば入力画素データ Dg のアドレス AD が探索領域 ARs1 内であった場合は、その入力画素データ Dg に対応して判定部 20A を選択する。

同様に、入力画素データ Dg のアドレス AD が探索領域 ARs2 内であった場合は、その入力画素データ Dg に対応して判定部 20B を選択する。

入力画素データ Dg のアドレス AD が探索領域 ARs3 内であった場合は、その入力画素データ Dg に対応して判定部 20C を選択する。

入力画素データ Dg のアドレス AD が探索領域 ARs4 内であった場合は、その入力画素データ Dg に対応して判定部 20D を選択する。

また、入力画素データ Dg のアドレス AD が探索領域 ARs1～ARs4 のいずれにも相当しなかった場合は、その入力画素データ Dg に対応しては、判定部 20A～20D のどれも選択しない。40

#### 【0113】

つまり、各判定部 20A～20D では、それぞれステップ F203A で、判定部選択部 4 によって選択されているか否かにより、その際に入力された画素データ Dg について処理を行うか否かを決定する。

選択されていなければ、ステップ F202 に戻って次の入力画素データ Dg を待機する。判定部選択部 4 によって選択されていれば、当該入力画素データ Dg について比較判定部 22 と線分判定部 24 によりステップ F204 以降の処理、つまり画素差分の算出と閾値 th との比較による計測フラグ制御、最大線分判定及び中点計算を行うことになる。

このステップ F204 以降の処理は、図 11 と同様であるため重複説明を避けるが、結果50

として、各判定部 20A～20D では、それぞれが担当する探索領域 ARs1～ARs4 において最大の候補線分 eL の中点 c を求める。その後ステップ F216～F218 の処理で、代表点 r の座標から中点 c の座標としての動きベクトル V を検出することになる。

#### 【0114】

このようにすることで、1画面上で、その画像内容の各部の動きベクトル V を検出することができ、これによって多様且つ利用範囲の広い動きベクトル検出が可能となる。

なお、図13では判定部選択部4を備える構成としたが、各判定部20A～20D内に、それぞれ入力画素データ Dg について担当する探索領域 ARs 内であるか否かを判別する機能を備えるようにしてもよい。

また、この例では代表点 r、探索領域 ARs を4つとしたが、これらを2つ、3つ、或いは5以上とする例も当然考えられ、そのためには判定部20が代表点の数に応じて構成されればよい。

#### 【0115】

##### 3. 代表点設定方式例

上述してきた重心検出方式、線分検出方式では、代表点 r は、直前のフレームの候補領域 eA の重心 w、もしくは直前のフレームの候補線分 eL の中点 c に基づいて設定されるものとした。しかしながら動きベクトル検出を開始する際の最初のフレームにおいては、直前のフレームが存在しないため、代表点 r を任意に設定するものとした。

例えばフレーム画面上で注目領域 ARa を設定し、その中心画素を代表点 r とする。これを代表点 r の第1の代表点設定方法とする。

#### 【0116】

このような第1の代表点設定方法は、設定処理が簡単であるという利点がある一方、代表点 r は単に注目領域 ARs の中心であるというだけであるため、選ばれた代表点 r に物理的な意味に乏しく、引き続き決定される動きベクトルの精度が低くなってしまう懸念がないこともない。

そこで、最初の代表点 r の設定を、次の第2、第3の設定方法のように行うことも考えられる。

#### 【0117】

第2の代表点設定方法としては、例えば注目領域 ARa 内で最も広い平坦部の重心を代表点とする。

これは、上記「重心検出」の考え方を適用するもので、つまり、注目領域 ARs において画素値が似ている画素群を検出し、その画素群の重心を代表点とする。画素値の似ている画素群が複数ある場合は、最大面積の画素群を選択して、その重心点を代表点 r とすればよい。

例えば上記重心検出方式の場合にこれを適用すれば、最初から重心点から重心点への動きベクトル検出ができる。また、最初の代表点が「重心点」として意味のあるものとして、3フレーム目以降も、最初の「重心点」の位置変化を追っていくことになる。これらのことから、動きベクトル検出の検出精度を向上させることができる。

#### 【0118】

第3の代表点設定方法としては、注目領域 ARa 内を水平方向に切り分け、各水平部分において最も長く連続する平坦部を検出し、全ての水平部分の中で最も長く連続する線分の中点を代表点とする。

これは上記「線分検出」の考え方を適用するもので、つまり、注目領域 AR において、水平方向(走査線方向)に見て画素差分が小さい画素の並び(線分)を検出し、その中最も長い線分の中点を代表点 r とする。

例えば上記線分検出方式の場合にこれを適用すれば、最初から、中点から中点への動きベクトル検出ができる。また、最初の代表点が「最大線分の中点」として意味のあるものとして、3フレーム目以降も、最初の「最大線分の中点」の位置変化を追っていくことになる。これらのことから、動きベクトル検出の検出精度を向上させることができる。

なお、走査線方向に線分長を検査することは、ハードウェア負担が小さいことは先に述べた通りである。

【0119】

4. 動きベクトル検出情報の用途

以上、本発明の実施の形態の動きベクトル検出について述べてきたが、ここでは、上記のようにして得られた動きベクトル検出結果を使用する用途、アプリケーションの例を示しておく。

例えば以下の用途が考えられる。もちろん例示する以外にも用途はある。

【0120】

・データ圧縮

10

動きベクトル先の画像信号を参照して最適な圧縮を行う。

・オブジェクトの追跡／トラッキング

例えばビデオカメラにおいて被写体の動きを検出し、自動追尾処理を行う。

・オブジェクトのトラッキング

例えば画像処理等のために画像データ内の特定対象をトラッキングする。

・監視カメラ

監視用カメラにおいて動きのある被写体を追尾する。

・動き物体抽出

カメラ撮影や画像処理のために動きのある物体の画像を抽出する。

・テロップ抽出

20

画像上に付加されたテロップを抽出する。

・手振れ補正

ビデオカメラにおける手ブレ補正

・フレーム倍速

フレーム数を倍にする

・スローーション画像生成

フレーム間に新しいフレームを作る

・I / P (インターレース / プログレッシブ) 変換

・ノイズ低減

動き成分とノイズ成分を分離する

30

・高画質化

・動きボケ補正

移動物体の動きボケを補正する

・画素ずらしサンプリング

オブジェクトが動くことで新しい情報が得られる

・カメラ動きの検出

・車載カメラにおける速度検出

・動き予測

・2 - 3 p u l l - d o w n 検出

【0121】

40

5. プログラム及び記録媒体

本発明のプログラムは、上述した重心検出方式又は線分検出方式による動きベクトル検出処理を演算装置、例えばCPUやDSPによって実現させるプログラムである。

即ち、ビデオカメラ、画像処理装置、パーソナルコンピュータなどにおいて、上述した判定部1の動作を、CPU等によって実現するために、上記図4、図8、図11、図14で説明した処理を実行させるプログラムが本発明のプログラムである。

【0122】

さらに、そのような本発明のプログラムが記録された本発明のプログラム記録媒体によれば、プログラムの提供が容易となり、本発明を広く提供できる。例えばビデオカメラ、画像処理装置の設計に好適となったり、汎用のコンピュータ装置により本発明を実施できる

50

。

### 【 0 1 2 3 】

本発明のプログラムは、機器に内蔵されている記録媒体としてのHDD(ハードディスクドライブ)や、ROM等に予め記録しておくことができる。

あるいはまた、フレキシブルディスク、CD-ROM(Compact Disc Read Only Memory)、MO(Magnet optical)ディスク、DVD(Digital Versatile Disc)、磁気ディスク、半導体メモリなどのリムーバブル記録媒体に、一時的あるいは永続的に格納(記録)しておくことができる。このようなリムーバブル記録媒体は、いわゆるパッケージソフトウェアとして提供することができる。

また本発明のプログラムは、リムーバブル記録媒体からインストールする他、ダウンロードサイトから、LAN(Local Area Network)、インターネットなどのネットワークを介してダウンロードすることもできる。

10

### 【 0 1 2 4 】

#### 【発明の効果】

以上の説明から理解されるように、本発明では「重心検出」及び「線分検出」として、比較的小規模のハードウェアで精度の良い動きベクトル検出が可能となるという効果がある。

重心検出は、過去フレームにおける代表点に対してこの代表点の画素値に近い画素を候補点として検出し、その検出された候補点が空間的に連続する候補領域の重心点を検出する。そして代表点を始点とし、重心点を終点として動きベクトルを検出する。

20

線分検出は、過去フレームにおける代表点に対してこの代表点の画素値に近い画素を候補点として検出し、候補点が走査方向(水平方向)に最も長く連続した候補線分の中点を検出する。そして代表点を始点とし、中点を終点として動きベクトルを検出する。

### 【 0 1 2 5 】

いずれの場合も、フレーム間の単なる点マッチングではなく、候補領域の重心点、或いは候補線分の中点として、物理的な意味のあるベクトル終点を検出するものであるため、精度の高い動きベクトルを検出できる。

また、重心点又は線分中点を検出することは、画像の不安定なエッヂ部分やノイズによる影響を受けにくいものとなり、その意味でも精度向上が実現される。

ハードウェア規模は従来の代表点マッチング方式とほぼ同等でよく、ブロックマッチング法の場合に比べて小規模化が実現される。もちろん上記理由から単なる点同士のマッチングである代表点マッチング方式よりも高精度の検出が可能となる。

30

### 【 0 1 2 6 】

また重心検出の場合、重心算出手段(重心算出ステップ)は、候補領域が複数存在する場合は、最大の候補領域を選択して、選択した候補領域について重心点を算出することで、より正確なベクトル検出が可能となる。

線分検出の場合、線分判定手段は、候補点が走査方向に連続する候補線分として、最大長となる候補線分を判定し、記憶しておけばよいため、必要な記憶容量が少なく、ハードウェア規模の小型化に好適である。

また、重心検出の場合の重心点、或いは線分検出の場合の中点に基づいて、次のフレームに対する代表点を設定することで、代表点として物理的意味が与えられ、動きベクトル検出の精度を上げることができる。

40

### 【 0 1 2 7 】

また動画映像データの過去フレーム内での複数の代表点を設定し、上記各代表点を用いて動きベクトルを検出するためには、重心検出の場合は比較判定手段、記憶手段、重心算出手段、及びベクトル検出手段を、線分検出の場合は比較判定手段、線分判定手段、中点算出手段、及びベクトル検出手段を、複数の代表点のそれぞれに対応するように設ければよく、これによって多様且つ利用範囲の広い動きベクトル検出が可能となる。

### 【 0 1 2 8 】

また本発明のプログラムによれば、上記効果を奏する動きベクトル検出方法、動きベクト

50

ル検出装置を実現でき、本発明の記録媒体によれば、上記の効果を実現する動きベクトル検出装置の実現及びプログラムの広範な提供に適している。

## 【図面の簡単な説明】

【図1】本発明の実施の形態の重心検出による動きベクトル検出の説明図である。

【図2】実施の形態の重心検出による動きベクトル検出の説明図である。

【図3】実施の形態の重心検出にかかる第1の装置構成例のブロック図である。

【図4】実施の形態の重心検出による動きベクトル検出処理のフローチャートである。

【図5】実施の形態の重心検出動作の説明図である。

【図6】実施の形態の複数の代表点を設定する場合の説明図である。

【図7】実施の形態の重心検出にかかる第2の装置構成例のブロック図である。

【図 8】実施の形態の重心検出による動きベクトル検出処理のフローチャートである

【図9】本発明の実施の形態の線分検出による動きベクトル検出の説明図である。

【図1.9】実施の形態の縦分検出にかかる第1の装置構成例のブロック図である。

【図1-1】実施の形態の線分検出による動きベクトル検出処理のフローチャートである。

【図12】実施の形態の線分判定処理の説明図である。

【図1-3】実施の形態の線分検出にかかる第2の装置構成例のブロック図である。

【図 1-4】実施の形態の線分検出による動画ベクトル検出処理のフロー手順である

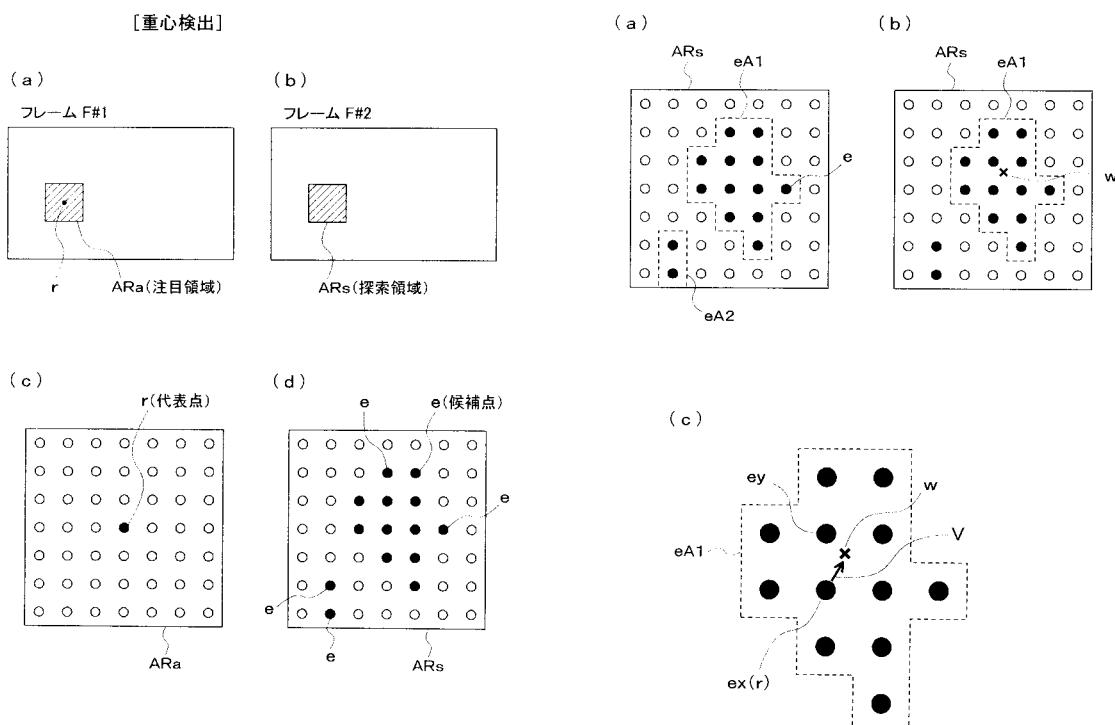
【図1.5】従来の代表点マッチング方法の説明図である

## 【符号の説明】

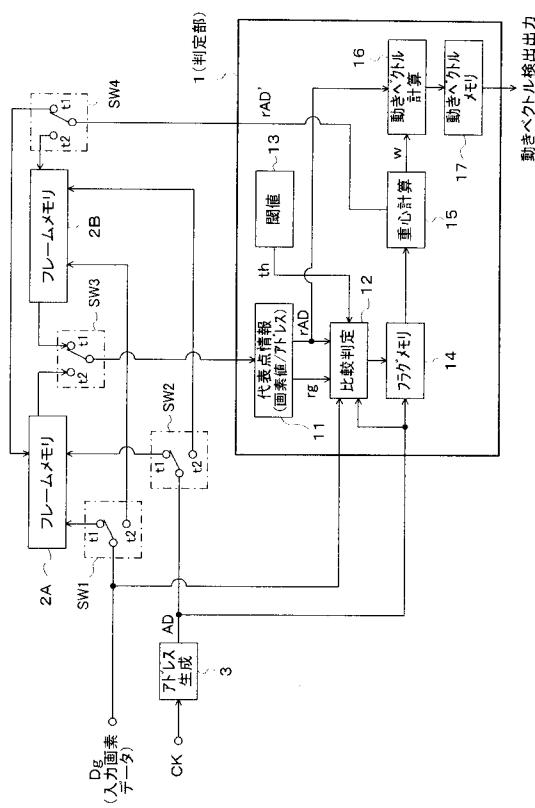
【内部構成】  
1 , 1 A ~ 1 D , 2 0 , 2 0 A ~ 2 0 D 判定部、2 A , 2 B フレームメモリ、3 アドレス生成部、4 判定部選択部、1 1 , 2 1 代表点情報保持部、1 2 , 2 2 比較判定部、1 3 , 2 3 閾値保持部、1 4 フラグメモリ、1 5 重心計算部、1 6 , 2 5 動きベクトル計算部、1 7 , 2 6 動きベクトルメモリ、2 4 線分判定部

( 1 )

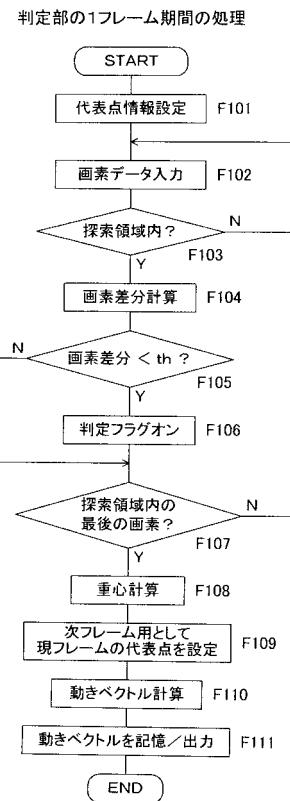
( 2 )



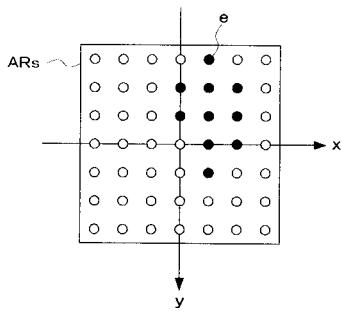
【図3】



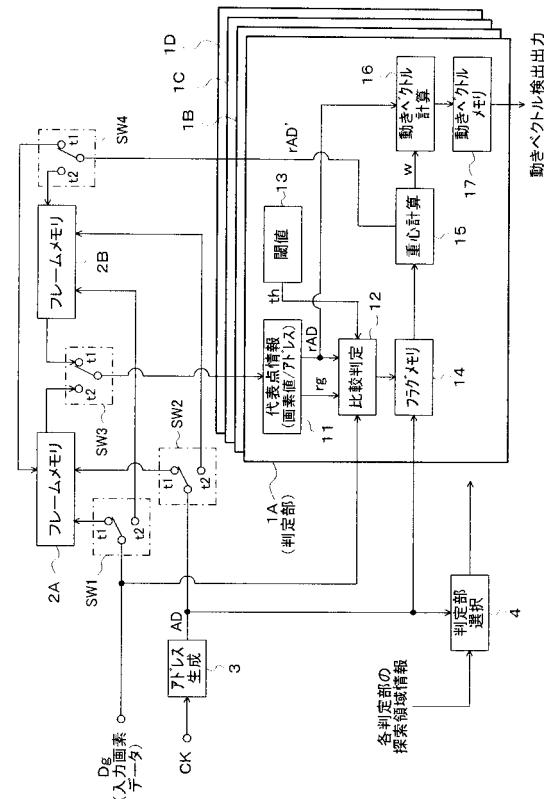
【図4】



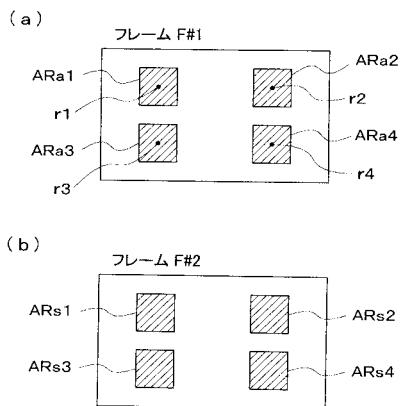
【図5】



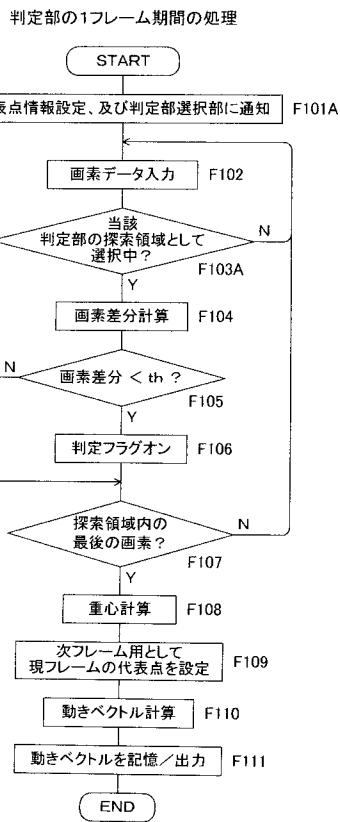
【図7】



【図6】

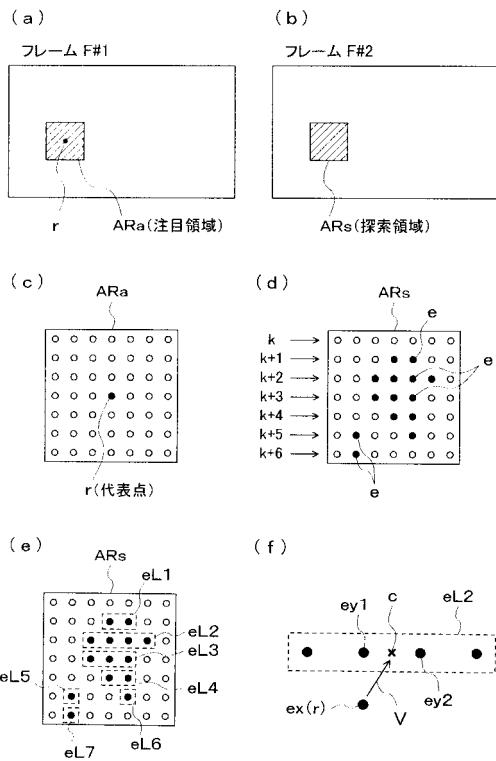


【図 8】

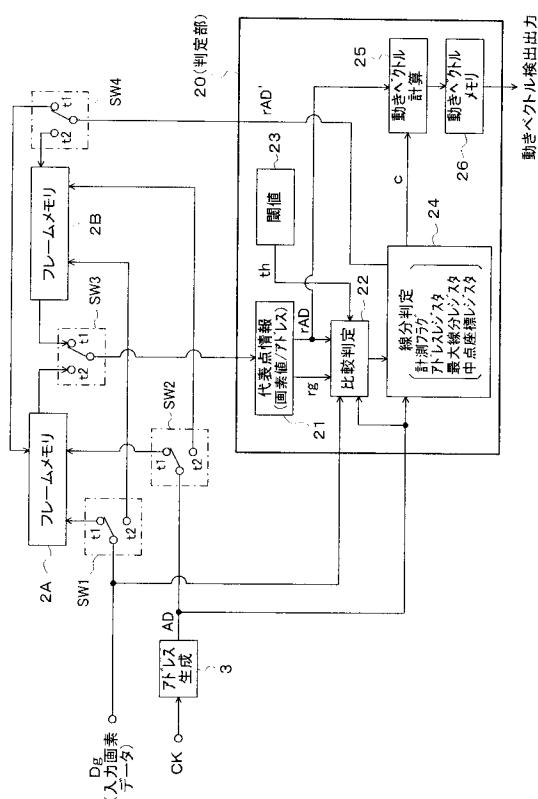


【図 9】

[線分検出]

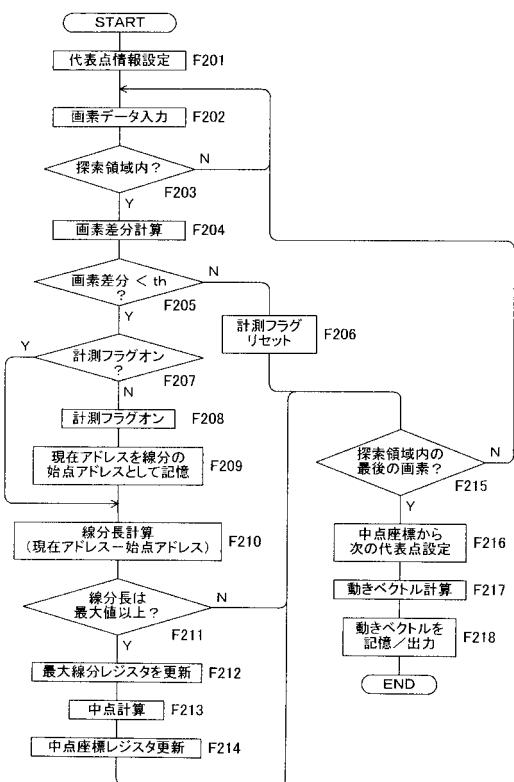


【図 10】

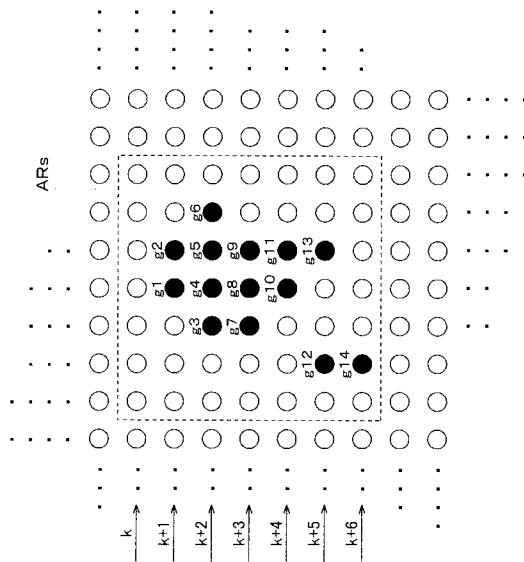


【図 11】

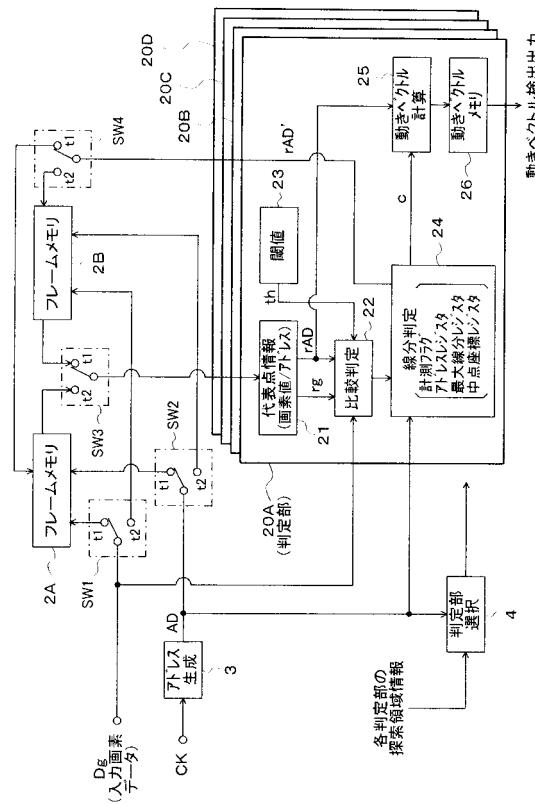
判定部の1フレーム期間の処理



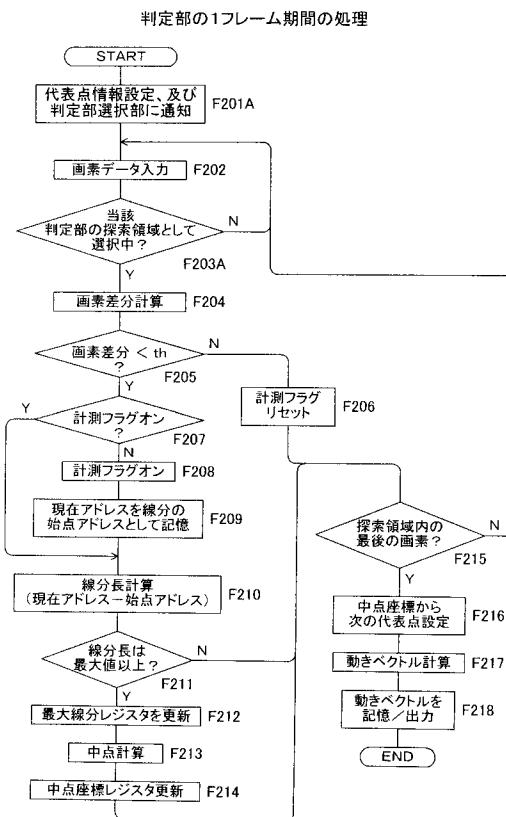
【図12】



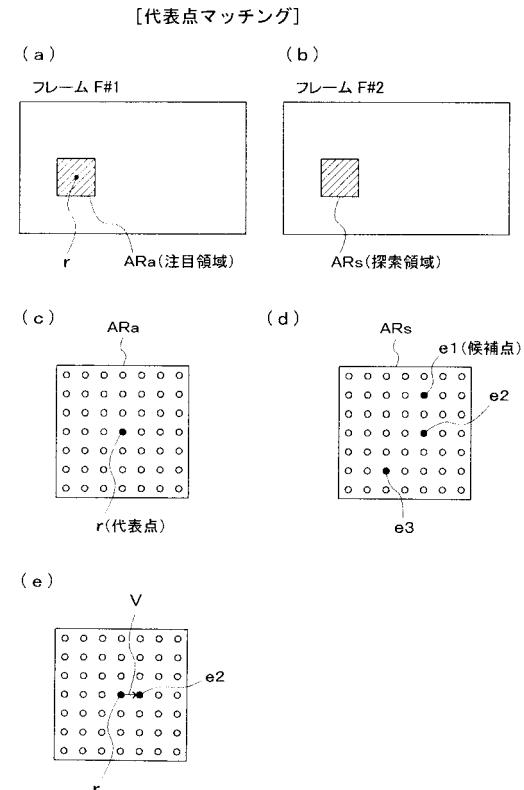
【図13】



【図14】



【図15】



---

フロントページの続き

(56)参考文献 特開平07-274178(JP,A)  
特開平08-063554(JP,A)  
特開2002-329196(JP,A)  
特開平07-274180(JP,A)  
特開平08-030495(JP,A)  
特開平08-237536(JP,A)  
特開平06-086149(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06T 7/00~7/60

H04N 5/232

H04N 7/32