

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2010-93820
(P2010-93820A)

(43) 公開日 平成22年4月22日 (2010.4.22)

(51) Int.Cl. F I テーマコード (参考)
HO4N 7/26 (2006.01) HO4N 7/13 Z 5C159

審査請求 未請求 請求項の数 10 O L 外国語出願 (全 57 頁)

<p>(21) 出願番号 特願2009-245887 (P2009-245887)</p> <p>(22) 出願日 平成21年10月2日 (2009.10.2)</p> <p>(31) 優先権主張番号 12/245,661</p> <p>(32) 優先日 平成20年10月3日 (2008.10.3)</p> <p>(33) 優先権主張国 米国 (US)</p>	<p>(71) 出願人 000002185 ソニー株式会社 東京都港区港南1丁目7番1号</p> <p>(71) 出願人 593181638 ソニー エレクトロニクス インク アメリカ合衆国 ニュージャージー州 O 7656 パークリッジ ソニー ドライ ブ 1</p> <p>(74) 代理人 100082005 弁理士 熊倉 禎男</p> <p>(74) 代理人 100067013 弁理士 大塚 文昭</p> <p>(74) 代理人 100109070 弁理士 須田 洋之</p>
---	---

最終頁に続く

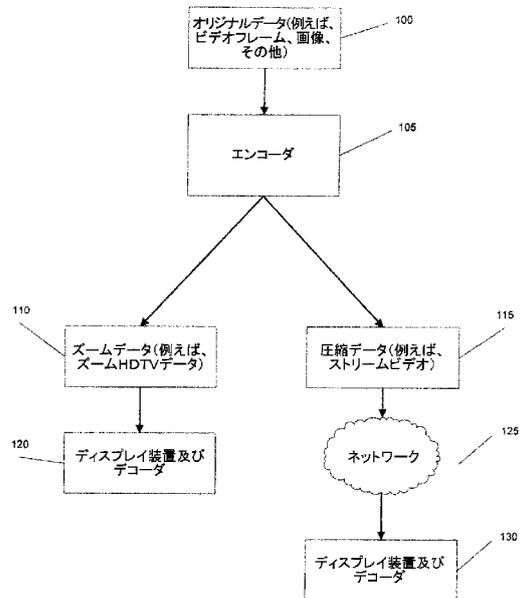
(54) 【発明の名称】 次元縮小法を使用した適応予測

(57) 【要約】 (修正有)

【課題】 次元縮小法を使用した適応予測又はビデオ処理方法を提供する。

【解決手段】 オリジナルデータ100が、次元縮小法を取り入れた適応予測を使用してエンコーダ105で符号化される。デシメートしたフィールドに基づいてマルチフィールドタップが定められる。マルチフィールドタップを使用して相関行列が生成され、相関行列の要素を使用して共分散行列が生成される。相関行列要素を固有ベクトルに投影することにより、主成分空間が求められる。主成分空間がクラスに区分され、各クラス毎に最小二乗フィルタセットが生成される。

【選択図】 図1A



【特許請求の範囲】**【請求項 1】**

デシメートしたフィールドに基づいてマルチフィールドタップを定めるステップと、
前記マルチフィールドタップを使用して相関行列を生成するステップと、
前記相関行列要素に基づいて共分散行列を生成するステップと、
前記相関行列要素を固有ベクトルに投影して主成分空間を求めるステップと、
前記主成分空間をクラスに区分するステップと、
前記クラスの各々に対して最小二乗フィルタセットを生成するステップと、
を含むことを特徴とするコンピュータによる方法。

【請求項 2】

オリジナルの高解像度フィールドをデシメートして、デシメートしたフィールドを生み出すステップをさらに含む、
ことを特徴とする請求項 1 に記載のコンピュータによる方法。

【請求項 3】

前記オリジナルの高解像度フィールドを再構成してディスプレイ装置に表示するステップをさらに含む、
ことを特徴とする請求項 2 に記載のコンピュータによる方法。

【請求項 4】

データ処理システムにより実行された場合、該データ処理システムにデータを処理する動作を実行させる命令を記憶するコンピュータ可読記憶媒体であって、前記動作は、
デシメートしたフィールドに基づいてマルチフィールドタップを定めるステップと、
前記マルチフィールドタップを使用して相関行列を生成するステップと、
前記相関行列要素に基づいて共分散行列を生成するステップと、
前記相関行列要素を固有ベクトルに投影して主成分空間を求めるステップと、
前記主成分空間をクラスに区分するステップと、
前記クラスの各々に対して最小二乗フィルタを生成するステップと、
を含むことを特徴とするコンピュータ可読記憶媒体。

【請求項 5】

前記方法は、オリジナルの高解像度フィールドをデシメートして、デシメートしたフィールドを生み出すステップをさらに含む、
ことを特徴とする請求項 4 に記載のコンピュータ可読記憶媒体。

【請求項 6】

前記方法は、前記オリジナルの高解像度フィールドを再構成してディスプレイ装置に表示するステップをさらに含む、
ことを特徴とする請求項 5 に記載のコンピュータ可読記憶媒体。

【請求項 7】

バスに接続されたプロセッサと、
前記バスを介して前記プロセッサに接続されたメモリと、
前記プロセッサにより前記メモリから実行されて、該プロセッサに、
デシメートしたフィールドに基づいてマルチフィールドタップを定めることと、
前記マルチフィールドタップを使用して相関行列を生成することと、
前記相関行列要素に基づいて共分散行列を生成することと、
前記相関行列要素を固有ベクトルに投影して主成分空間を求めることと、
前記主成分空間をクラスに区分することと、
前記クラスの各々に対して最小二乗フィルタを生成することと、
を行わせる命令と、
を備えることを特徴とするシステム。

【請求項 8】

前記命令は、オリジナルの高解像度フィールドをデシメートして、デシメートしたフィールドを生み出すことをさらに含む、

10

20

30

40

50

ことを特徴とする請求項 7 に記載のシステム。

【請求項 9】

前記命令は、前記オリジナルの高解像度フィールドを再構成してディスプレイ装置に表示することをさらに含む、

ことを特徴とする請求項 8 に記載のシステム。

【請求項 10】

デシメートしたフィールドに基づいてマルチフィールドタップを定めるための手段と、

前記マルチフィールドタップを使用して相関行列を生成するための手段と、

前記相関行列要素に基づいて共分散行列を生成するための手段と、

前記相関行列要素を固有ベクトルに投影して主成分空間を求めるための手段と、

前記主成分空間をクラスに区分するための手段と、

前記クラスの各々に対して最小二乗フィルタを生成するための手段と、

を含むことを特徴とするデータ処理システム。

【発明の詳細な説明】

【技術分野】

【0001】

〔関連出願との相互参照〕

本出願は、______に出願された「適応予測フィルタのための動的クラスタリング」という名称の以下の同一出願人による同時係属出願：出願番号第______号（代理人整理番号 080398、P738）と関連性がある。

【0002】

本発明は、一般的にビデオ処理に関し、より具体的には、次元縮小法を使用した予測に関する。

【0003】

〔著作権表示 / 許諾〕

本特許文書の開示部分は、著作権保護の対象内容を含む。著作権の権利所有者は、合衆国特許商標庁の特許ファイル又は記録内に表されるとおりに他者が特許文書又は特許開示を複製することには異議を唱えないが、それ以外は全ての著作権を留保する。以下の表記が適用される：著作権 (C) 2008, Sony Electronics Inc., 禁無断転載。

【背景技術】

【0004】

ビデオ符号化における根本的な問題は、利用可能な低解像度データを使用した高解像度画像の予測である。通常、これらの場合には最小二乗 (LS) フィルタが使用され、この LS フィルタ係数を使用して、場合によっては異なる時点からの利用可能な低解像度データを使用して高解像度画像を作成する。

【0005】

現時点の高解像度画像を取得するために、ズームングが、場合によっては異なる時点からのデータを利用するステップと、利用可能な情報を組み合わせるステップとを必要とする。デシメートした画像を生成し、最良の LS フィルタの組を発見してオリジナル画像と予測画像との間の差分を最小にしたオリジナル画像を再度取得することにより、LS フィルタのトレーニングが可能となる。

【0006】

フィルタタップとは、画素領域に重ねられるパターンのことである。フィルタタップに重なる画素が、その画素に関連するフィルタ係数の基礎を形成する。

【発明の概要】

【課題を解決するための手段】

【0007】

デシメートしたフィールドに基づいてマルチフィールドタップが定められる。マルチフィールドタップを使用して相関行列が生成され、相関行列の要素を使用して共分散行列が

10

20

30

40

50

生成される。相関行列要素を固有ベクトルに投影することにより、主成分空間が求められる。主成分空間がクラスに区分され、各クラス毎に最小二乗フィルタセットが生成される。

【図面の簡単な説明】

【0008】

【図1A】次元縮小法を使用した適応予測又はビデオ処理の1つの実施形態を示す図である。

【図1B】オリジナルのフィールドと、デシメートしたフィールドと、予測フィールドとの間の関係を示す図である。

【図2】本発明の実施形態による、主成分分析を使用した適応予測の方法を示すフロー図である。

【図3A】本発明の実施形態によるデシメーションの例を示す図である。

【図3B】本発明の実施形態によるデシメーションの例を示す図である。

【図4】本発明の実施形態による複数の時点からのフィールドを使用した予測を示す図である。

【図5】本発明の実施形態による固有ベクトルの分散範囲を示す図である。

【図6】本発明の実施形態による主成分空間を示す図である。

【図7A】本発明の実施に適したコンピュータ環境を示す図である。

【図7B】本発明の実施に適したコンピュータ環境を示す図である。

【図8】本発明の実施形態による分類性能を示す図である。

【図9】本発明の実施形態による点分布を示す図である。

【図10A】本発明の実施形態による別の点分布を示す図である。

【図10B】本発明の実施形態による別の点分布を示す図である。

【図10C】本発明の実施形態による別の点分布を示す図である。

【図10D】本発明の実施形態による別の点分布を示す図である。

【図11】本発明の実施形態による様々な観点から見たさらに別の点分布を示す図である。

【図12】本発明の実施形態による別の分類性能を示す図である。

【図13】本発明の実施形態の様々な構成による性能比較を示す図である。

【図14】本発明の実施形態による画素をクラスタリングする方法を示すフロー図である。

【発明を実施するための形態】

【0009】

以下の本発明の実施形態の詳細な説明では、同じ参照番号が同様の要素を示す添付の図面を参照し、図面には、本発明を実施できる特定の実施形態を例示目的で示す。これらの実施形態について、当業者が本発明を実施できるように十分に詳しく説明し、また、他の実施形態を利用できるとともに、本発明の範囲から逸脱することなく論理的、機械的、電氣的、機能的及びその他の変更を行うことができる。従って、以下の詳細な説明を限定的な意味で捉えるべきではなく、本発明の範囲は添付の特許請求の範囲によってのみ定められる。

【0010】

本発明の動作の概説から始めると、図1Aは、次元縮小法を使用した適応予測又はビデオ処理の1つの実施形態を示している。オリジナルデータ100が、次元縮小法を取り入れた適応予測を使用してエンコーダ105で符号化される。オリジナルデータ100は、ユーザが装置120などの標準画質テレビで見たいと思う、ブルーレイプレーヤーから得たビデオデータであってもよい。これを行うために、エンコーダ105は、データ100をズームしてズームデータ110を生成することができ、これを装置120が適当なアスペクト比で復号し、表示することができる。エンコーダ105から得られる符号化データは圧縮データ115の形を取ることができ、これが(インターネットを介したストリーミングビデオなどのように)復号及び表示するための装置130にネットワーク125を介

10

20

30

40

50

して送信される。

【 0 0 1 1 】

1つの実施形態では、エンコーダ105はビデオデータのフレームを分類し、個々の予測フィルタをフレーム内の各クラスに割り当てることができる。分類による適応予測は、フレームの画素をいくつかのクラスに区分し、そのクラスにフィルタを適応させるステップを含む。フィルタは、その画素を予測又はデシメートするために使用される各画素の係数を含むことができる。フィルタ係数及び対応する画素が、フィルタタップを介して関連付けられる。フィルタタップは、対応する画素及びその画素の周囲の画素に重ね合わせられるパターンである。このパターンが、係数の生成にどの画素を使用するかを定める。1つの実施形態では、主成分分析（PCA）を使用してビデオフレームをクラスに分類する。PCAは、フィルタタップが定める対応する相関行列要素で低解像度フレーム内の各画素を表すことができる。PCAは、各画素の相関情報を使用して、画像内の類似した時空間偏差を有する画素をクラスにグループ分けする。

10

【 0 0 1 2 】

分類フィルタの定義は、デシメートされた画像と、その対応するオリジナル解像度の画像との間の関係に依存する。図1Bは、オリジナルフィールドと、デシメートしたフィールドと、予測フィールドとの間の関係を示している。オリジナル（ターゲット）フィールド150、155、及び160がデシメートされ、対応する低解像度フィールド165、170、及び175が得られる。デシメートしたフィールドから利用できる情報を使用して、オリジナルフィールド150～160と予測フィールド180との間の差分を最小にする予測180及びLSフィルタ係数185を得る。予測エラーは次式のように定義することができる。

20

$$e = \sum_{\vec{x}} (I(\vec{x}) - P(\vec{x}))^2$$

(1)

この場合、 $I(x)$ はオリジナル画像であり、 $P(x)$ は予測画像であり、 $P(x)$ は次式のように定義される。

$$P(\vec{x}) = \sum_j L_j^u I^D(\vec{y}_j^x)$$

30

(2)

この場合、 $I^D(y)$ はデシメートされた画像を表す。

【 0 0 1 3 】

図2は、本発明の実施形態によるPCAの方法200を示す図である。ブロック205において、本方法は、（ハイビジョンなどの）オリジナルの高解像度フィールドをデシメートする。フィールドのデシメーションについては、図3A～図3Bと共に以下でさらに詳細に説明する。

【 0 0 1 4 】

ブロック210において、本方法は、デシメートしたフィールドに対するマルチフィールドタップを定める。マルチフィールドフィルタタップは、対応する画素の係数を生成するために使用される複数のデータフィールドにまたがる。マルチフィールドフィルタタップについては、図3B及び図4と共に以下でさらに詳細に説明する。

40

【 0 0 1 5 】

ブロック215において、本方法は、マルチフィールドフィルタタップにおける画素間の相互関係を含む相関行列を生成する。相関行列の要素は、高次元空間におけるデシメートした画像内の画素を表す。この方法は、PCAを使用して低次元空間に問題を引き継ぎ、画素をより効率的に分類する。

【 0 0 1 6 】

ブロック220において、本方法は、式3のように相関行列要素の共分散行列を生成す

50

る。本方法は、個々のデシメートした画素の相関行列要素を列形式で配置し、各列から平均ベクトルを除去した後でm列を有する観測行列Xを生成する。平均ベクトルの除去により、画素の周囲の局所的変化が強調される。本方法は、Xの共分散行列に固有分解を適用する。この分解はn個の固有ベクトル及びn個の固有値を生み出し、この場合nは相関行列における要素の数である。共分散行列は、画素の数であるmとnとの小さい方である階数nを有する。Xの共分散行列は、次式のように定義することができる。

$$\hat{C}_{xx} = \frac{1}{m-1} X X^T = \frac{1}{m-1} \sum_{i=1}^m (\underline{v}_i - \hat{\underline{\mu}})(\underline{v}_i - \hat{\underline{\mu}})^T \quad (3)$$

この場合、

$$\hat{\underline{\mu}} = \frac{1}{m} \sum_{i=1}^m \underline{v}_i \quad (4)$$

は平均ベクトルである。

【0017】

固有分解動作は、式5のように要約することができる。Qは、列における固有ベクトルを含み、 Λ は対応する固有値：

$$\hat{C}_{xx} = Q \Lambda Q^T = Q \Lambda Q^{-1} \quad (5)$$

を有する対角行列である。

固有ベクトルは正規直交基底を構成するので、 Q^T を Q^{-1} に置き換えることができる。

【0018】

ブロック225において、本方法は、Qの列にXの列を投影することにより固有ベクトルに行列Xを投影して主成分を求め、式6のように主成分の分布を含む空間を生成する。主成分は、Xの個々の列が個々の固有ベクトルにどれほど類似しているかを示す。行列PはPCを含む。個々の画素は、所定の固有ベクトルの数により表される。

$$P = Q^T X$$

(6)

フィルタ係数は、中心画素の周囲のデシメートした画素に対応する相関行列の関数である。フィルタタップはフィルタ係数を識別する。本方法は、対応する相関行列の類似性に基づいてデシメートした画像の画素を分類し、各クラス毎に異なるフィルタ係数の組を生成する。本方法は、個々のデシメートした画像の画素を表すためにフィルタタップの相関情報を使用して固有ベクトルを求め、その固有値に従って固有ベクトルをソートする。

【0019】

方法200は、PCAを使用して、固有ベクトルが定める軸を含む空間に分類問題を変換することができる。この方法は、縮小した次元空間におけるデシメートした画素の座標としてXの列（各列は異なる画素を表す）と固有ベクトルとの間の類似性を使用する。

【0020】

ブロック230において、方法200はPC空間をクラスに区分する。PC空間は、よ

10

20

30

40

50

り低次元の空間におけるデシメートした画像の画素を表し、類似の相関行列を含む領域が P C 空間に集められる。本方法は、各領域に等しい数の画素が存在するように最初に軸の各々を 3 つの領域に区分し、この結果、三次元の中に総数 27 の領域が生じ、画素が 27 のグループに分類される。他の実施形態では、クラスの数は静的に 27 とは定められない。異なる数のクラスを選択するステップについては、図 14 と共に以下で詳細に説明する。

【 0 0 2 1 】

ブロック 235 において、本方法は、各クラス毎に最小二乗 (LS) フィルタを生成する。LS フィルタは、オリジナル (ターゲット) フレームにおける全画素に渡ってオリジナル画像と予測画像との間の差分の二乗和を最小にする。本方法は、個々のフィルタ要素に関して偏導関数を取り、これを各々についてゼロにすることによりフィルタ係数を求める。

10

$$e = \sum_{\vec{x}} (I(\vec{x}) - P(\vec{x}))^2$$

(7)

【 0 0 2 2 】

方法 200 は、図 2 に示すように、フィルタタップが特定した画素の相関関係を使用してフィルタタップにおける係数の組 L を決定する。

20

$$L = (A)^{-1}b$$

(8)

この場合、

$$A_{j,k} = \sum_{\vec{x}} I^D(\vec{y}_j^{\vec{x}}) I^D(\vec{y}_k^{\vec{x}})$$

$$b_k = \sum_{\vec{x}} I^D(\vec{y}_k^{\vec{x}}) I(\vec{x})$$

30

(9、10)

この場合、 $A_{j,k}$ は相関行列であり (添字 j 及び k は 2 つのフィルタタップ T の範囲を動く)、 b_k は観測ベクトルである。 I^D はデシメートした画像を表す。

【 0 0 2 3 】

方法 200 は、観測ベクトル及び相関行列の逆数を使用して LS フィルタを生成する。この方法は、各領域に固有の異なる LS フィルタを使用する。デシメートした画像に m 個の画素と c 個のクラス (領域) が存在すると仮定すると、

40

$$m = m_1 + m_2 + \dots + m_c$$

(11)

が、各クラスにおける画素の数を示す。与えられたクラス u に関して、

$$L^u = (A^u)^{-1}b^u$$

(12)

この場合、

$$A_{j,k}^u = \sum_{\vec{x}|u=S(\vec{x})} I^D(\vec{y}_j^{\vec{x}}) I^D(\vec{y}_k^{\vec{x}})$$

$$b_k^u = \sum_{\vec{x}|u=S(\vec{x})} I^D(\vec{y}_k^{\vec{x}}) I(\vec{x})$$

(1 3 、 1 4)

この場合、

10

$$u = S(\vec{x})$$

(1 5)

が、ターゲット画素 x の予測フィルタ分類マップである。

【 0 0 2 4 】

図 3 A は、1つの期間を使用した空間アルゴリズムのためのデシメーションを示す図である。フィルタタップ 3 0 5 が識別した個々のソース画素が、その値の $1/4$ をデシメートした画素 3 1 0 の値に与える。

【 0 0 2 5 】

図 3 B は、2つの期間を使用した本発明の実施形態による時空アルゴリズムのためのデシメーションを示す図である。図 3 B に示すデシメーション動作は、予測アルゴリズムへの入力として2又はそれ以上の時間フィールドを使用する。マルチフィールドフィルタタップ 3 5 0 が、その時間フィールドから引き出される画素を決定する。デシメーション動作は、デシメーションを加重係数で均一に分散させてデシメートした画素 3 5 5 を生成する。他の実施形態では、より複雑かつ適応的なデシメーションアルゴリズムが使用される。

20

【 0 0 2 6 】

図 4 は、3つの異なる時間フィールドから得た3つのフィールド 4 0 5 を使用する時空フィルタタップ例を示す図である。このマルチフィールドフィルタタップは、予測フィルタへの入力として各フィールドからの9つのデシメート画素 4 1 0 を使用して中心画素 4 2 0 の周囲の画素 4 1 5 を予測する。マルチフィールドフィルタタップは、デシメートした画像上を移動して、より高解像度のビデオフィールドを予測する。この方法は、最小二乗法を通じて最適化したフィルタ要素を求めることができる。

30

【 0 0 2 7 】

図 5 は、本方法が行列形式に戻した後の、減少分散を含む最初の9つの固有ベクトル 5 0 5 を示す図である。パーセンテージ 5 1 0 は、個々の固有ベクトルがオリジナルフィールドの総分散のうちどの程度を運ぶかを示している。

【 0 0 2 8 】

1つの実施形態では、この方法は、デシメートした画素の各々を3つの座標で表す、三次元空間における最初の3つの固有ベクトルを使用して向上した結果を得る。図 6 は、区分動作に基づくこれらの画素のサブセットの分布を示す図である。軸 6 0 5、6 1 0、及び 6 1 5 は最初の3つの固有ベクトルに対応し、個々の画素の座標は対応する主成分である。

40

【 0 0 2 9 】

別の実施形態では、ブロック 2 3 0 において、方法 2 0 0 がクラスタリングを使用して、動的有効性指標を使用した縮小次元空間におけるより細かいグループに画素を分類する。動的有効性指標は、画素クラスの緊密度及びクラス自体の間の距離を測定する費用関数である。クラスタリングの目的は、同じクラスの画素間の距離を最小にすることによりクラスの緊密度を増加させるとともにクラス間の距離を最大にすることである。実験的データが、反復クラスタリングが画素を効率良く分類し、SNRの向上を実現することを示唆

50

している。

【0030】

図14は、本発明の実施形態による画素クラスをクラスタリングする方法1400を示す図である。クラスタの数 c は、PC空間における画素の分布に関わらず分かっているものとして処理される。ブロック1405において、本方法は、使用するクラスの数を決

【0031】

ブロック1410において、本方法は、個々の画素に数字を付け、この範囲における c の数の均一な分布を選択する。本方法は、クラスタされる c クラスの重心（重み付き中心）の初期値としてこの均一な選択を使用する。

【0032】

ブロック1415において、本方法は、重心の各々までの画素の距離に基づいて、個々の画素を最も近い重心（及びクラスタ）に割り当てる。

【0033】

ブロック1420において、本方法は、画素の割り当てを使用して個々のクラスタの新しいクラスタ重心を決定する。

【0034】

ブロック1425において、本方法は、現在のクラスタリング構成の動的有効性指標を評価する。他の実施形態では、他の費用関数を使用することができる。動的有効性指標は次式のように定義することができる。

$$\text{動的有効性指標} = \frac{\text{Intra}}{\text{Inter}} = \frac{\frac{1}{N} \sum_{i=1}^k \sum_{x \in C_i} \|x - z_i\|^2}{\min_{i,j} (\|z_i - z_j\|)^2} \quad (16)$$

この場合、 N は点（データオブジェクト）の数、 k はクラスタの数、 x はデータ点の位置、 z_i は i 番目のクラスタのクラスタ中心の位置（重心）、及び C_i は i 番目のクラスタにおける点の組である。動的有効性指標は、クラスタ内の項（分子）とクラスタ間の項（分母）とを含む。クラスタ内の項は、各点の対応する重心までの平均距離であり、全クラスタの緊密度の平均レベルを測る。クラスタ間の項は、全部のクラスタのペアの中で互いに最も近いクラスタのペアの間の距離である。

【0035】

ブロック1430において、本方法は、最も近いクラスタ重心に基づいてPC空間における個々の点（画素）に新しいクラスメンバーシップを割り当てる。

【0036】

ブロック1435において、本方法は、新しいクラスの割り当てに基づいて個々のクラスタの新しいクラスタ重心を決定する。

【0037】

ブロック1440において、本方法は、新しい構成の有効性指標を評価し、この有効性指標をブロック1425において生成された有効性指標と比較する。指標間の差がしきい値を超える場合、本方法はブロック1430へ戻り、クラスタリングをさらに繰り返す。しきい値を超えなければ、本方法は現在のクラスタリング方式を使用する。最初の繰り返し中、本方法は、ブロック1440において評価された指標をブロック1425において評価された指標と比較する。次の繰り返し中、本方法は、ブロック1440において評価された指標を比較する。

【0038】

本方法は、各繰り返しにおいて動的有効性指標を評価し、繰り返し間の変化がしきい値よりも下がるまでこれを続行する。これを行っているときには、さらなる割り当ては行われず、方法1400は、現在のクラスタリング方式を使用してPC空間を区分する。実験的データが、結果が飽和し、しきい値よりも下がるまでに20回から30回の間の繰り返

10

20

30

40

50

しが必要であることを示唆している。

【0039】

方法1400は、動的有効性指標を使用してクラスタリング構成のための比較点を提供する。本方法は、現在の重心の組に基づいてクラスの割り当てを再帰的に調整し、現在のクラスの割り当てに基づいて重心を調整する。このようにして、本方法は、向上が飽和状態になるまでクラスタリングを繰り返し向上させる。

【0040】

方法1400が図2のブロック230においてクラスタリング動作を終了した後、方法200は、各クラスタの個々のメンバーを使用してLSフィルタ係数を生成し、これが動的にクラスタされたLSフィルタを生成する。

10

【0041】

図8は、静的クラスタリング805と動的クラスタリング810との信号対雑音比(SNR)性能の比較800を示す図である。実績の赤色の部分は、方法1400などの動的クラスタリング法で得られたクラス固有のLSフィルタを使用した結果生じた利得を示す。

【0042】

クラスタリングを繰り返すことにより、デシメートした画像の画素が分類され、各クラス毎に異なるLS係数の組が生成される。図9は、三次元空間におけるデシメートした画像の画素の分布例900を示す図である。個々の画素を主成分1、2、及び3で表している。

20

【0043】

図10A~図10Dは、PC1-2 1005、PC2-3 1010、PC3-4 1015、及びPC4-5 1020のペアワイズ分布を表している。これらのペアは、様々な二次元平面への五次元分布の投影を表す。

【0044】

図11は、32個のクラスタに分割された分布の様々な投影1105~1120を示す図である。各クラスタを異なる色調のグレーで表している。

【0045】

図12は、本発明の実施形態を使用して得られたSNRの結果1205を示す図である。最初の5つの固有ベクトルを使用して五次元空間により全てのフィールドに関して(ビデオシーンの間に60回)分析を繰り返した。動的クラスタリングLSフィルタのSNR性能1215を、静的クラスタリングLSフィルタのSNRパフォーマンス1210と比較する。全てのシーンに渡って約0.6dBのNSR性能の上昇が得られた。

30

【0046】

図13は、クラスタリング方法のパラメータを変更した同様の分析の結果を示す図である。7つのPCまでは、より多くのPC1315を使用したときにSNR向上1305が得られる。分散の99パーセントが、最初の7つのPCに確実に組み入れられる。予想されるように、クラス1310の数を増やした場合に性能が上昇し続けた。

【0047】

図2及び図14に示す一連のフロー図を参照しながら、本発明の特定の方法についてコンピュータソフトウェアの観点から説明した。本方法は、ブロックの形で示す機械実行可能命令で作られたコンピュータプログラムを構成する。フロー図を参照して本方法について説明することにより、当業者は、好適に構成された機械(メモリを含む機械可読媒体からの命令を実行する機械のプロセッサ)において本方法を実行するためのこのような命令を含むこのようなプログラムを開発できるようになる。機械実行可能命令は、コンピュータプログラミング言語で書かれたものであってもよく、或いはファームウェアの論理に統合したものであってもよい。認められている規格に準拠するプログラミング言語で書かれている場合、様々なハードウェアプラットフォームにおいて、及び様々なオペレーティングシステムへのインターフェイスのためにこのような命令を実行することができる。また、ある特定のプログラミング言語に関連して本発明を説明してはいない。様々なプログラ

40

50

ミング言語を使用して、本明細書で説明するような本発明の教示を実現できることが理解できよう。さらに、1つの動作を行う場合、或いは1つの結果をもたらす場合に、(プログラム、手順、処理、アプリケーション、モジュール、論理...などの)1つの形又は別の形のソフトウェアについて語るのは当業では一般的なことである。このような表現は、コンピュータがソフトウェアを実行することにより、コンピュータのプロセッサが動作を実行し、或いは結果を生み出すようになるという言い方を簡単にしたものにはすぎない。本発明の範囲から逸脱することなく、図2及び図14に示す方法により多くの又はより少ない処理を組み込むことができること、及び図示するとともに本明細書で説明したブロックの構成により特定の順序を暗示するわけではないことが理解されよう。

【0048】

1つの実施形態では、図7Aに示すように、サーバコンピュータ701がインターネット705に接続され、このインターネット705を介してデータを提供する。クライアントコンピュータ703が、ISP(インターネットサービスプロバイダ)707を介してインターネット705に接続され、従来のインターネットブラウジングアプリケーションを実行してサーバ701とデータを交換する。サーバコンピュータ701は、図2及び図14に示す方法を実行することができる。任意的に、サーバ701は、クライアントシステムにインターネットへのアクセスを提供するISPの一部であってもよい。本明細書で使用する「インターネット」という用語は、TCP/IPプロトコル、及び場合によってはワールドワイドウェブ(web)を構成するハイパーテキストマークアップ言語(HTML)文書のためのハイパーテキスト転送プロトコル(HTTP)のようなその他のプロトコルなどのいくつかのプロトコルを使用するネットワークのネットワークを意味する。インターネットの物理的接続及びインターネットのプロトコル及び通信手順は当業者にとって周知である。インターネットにアクセスすることにより、クライアントコンピュータシステムのユーザは、情報を交換し、電子メールを送受信し、HTMLフォーマットで作成された文書などの文書を閲覧し、コンテンツを受信できるようになる。本発明が、インターネットアクセス及びインターネットウェブベースのサイトに限定されるものではなく、直接接続されたプライベートネットワークも企図されることが容易に明らかである。

【0049】

サーバ701としての使用に適したコンピュータシステムの1つの実施形態を図7Bに示す。コンピュータシステム710は、システムバス735に接続されたプロセッサ720、メモリ725、及び入力/出力機能730を含む。メモリ725は、プロセッサ720により実行された場合、本明細書で説明した方法を実行する命令を記憶するように構成される。メモリ725はまた、次元縮小を使用した適応予測のためのノについてのデータを記憶することもできる。入力/出力730は、次元縮小、或いはその一部又は代表部分を使用した適応予測のためのノについてのデータの配信及び表示を行い、また記憶、処理又は表示のための様々な種類のデータの入力も行う。入力/出力730はまた、プロセッサ720によるアクセスが可能なあらゆる種類の記憶装置を含む様々な種類の機械可読媒体も含む。当業者であれば、サーバ701が、メモリ725で実行するオペレーティングシステムソフトウェアにより制御されることを直ちに認識するであろう。入力/出力730及び関連する媒体は、オペレーティングシステムのための機械実行可能命令及び本発明の方法、並びに次元縮小を使用した適応予測のためのノについてのデータを記憶する。

【0050】

図7A~図7Bの説明は、本発明の実施に適したコンピュータハードウェアおよびその他の動作構成要素の概観を示すことを意図したものであり、利用可能な環境を限定することを意図したのではない。コンピュータシステム740が、異なるアーキテクチャを有する多くの考えられるコンピュータシステムのうちの1つの例であることを理解できよう。典型的なコンピュータシステムは、通常、少なくともプロセッサ、メモリ、及びメモリをプロセッサに接続するバスを含む。当業者であれば、マルチプロセッサシステム、ミニコンピュータ、及びメインフレームコンピュータなどを含むその他のコンピュータシステム構成により本発明を実施できることを直ちに理解するであろう。通信ネットワークを介

10

20

30

40

50

してリンクされた遠隔処理装置によってタスクを実行する分散コンピューティング環境において本発明を実施することもできる。

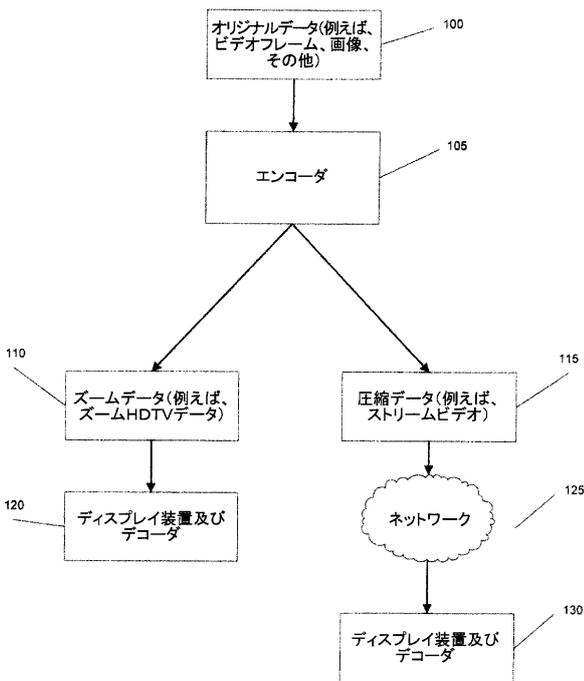
【0051】

次元縮小を使用した適応予測について説明した。本明細書では特定の実施形態について図示し説明したが、当業者であれば、同じ目的を実現するために計算されたあらゆる構成を、図示の特定の実施形態と置き換えることができることを理解するであろう。本出願は、本発明のあらゆる適応又は変形を対象とすることが意図される。

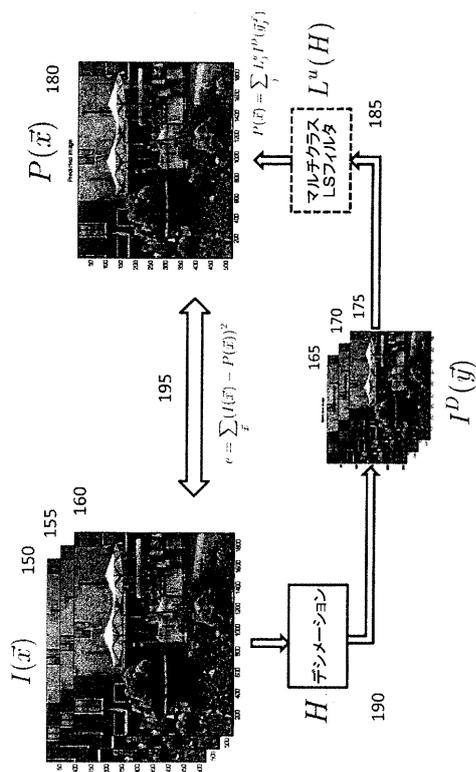
【0052】

次元縮小を使用した適応予測に関して本出願で使用した専門用語は、これらの環境の全てを含むことを意図したものである。従って、本発明は、以下の特許請求の範囲及びこの同等物によってのみ限定されることがはっきりと意図される。

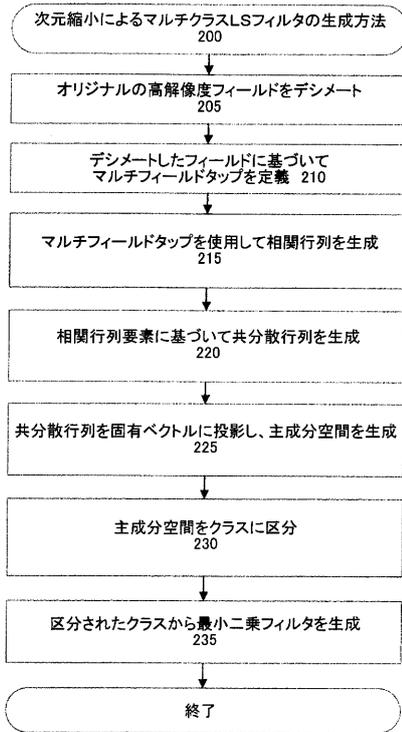
【図1A】



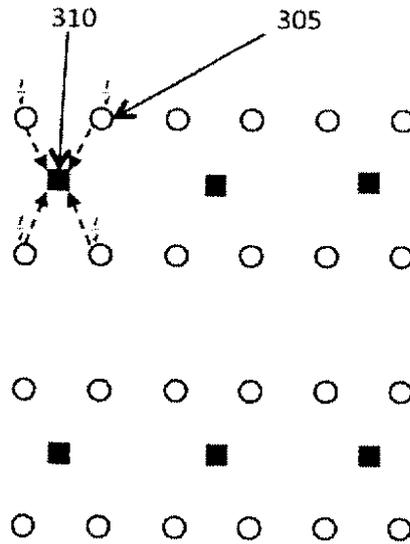
【図1B】



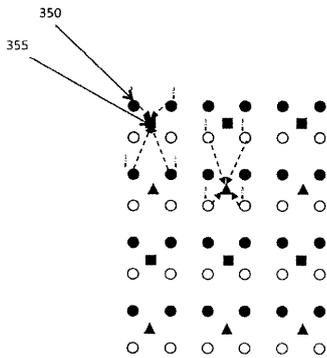
【 図 2 】



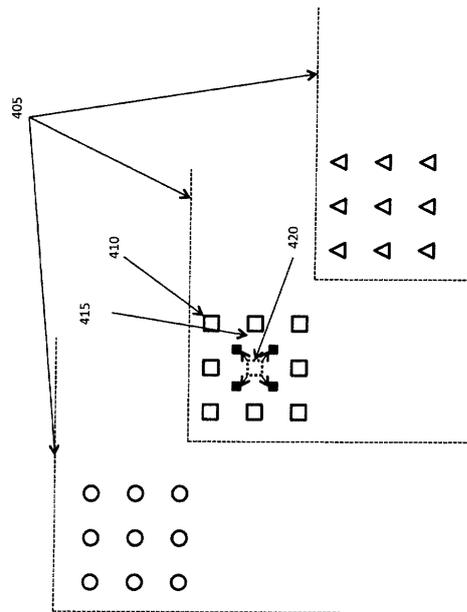
【 図 3 A 】



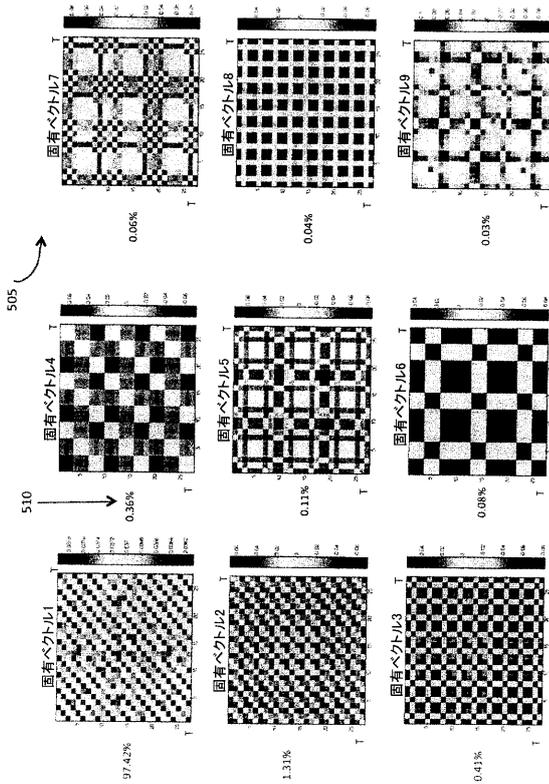
【 図 3 B 】



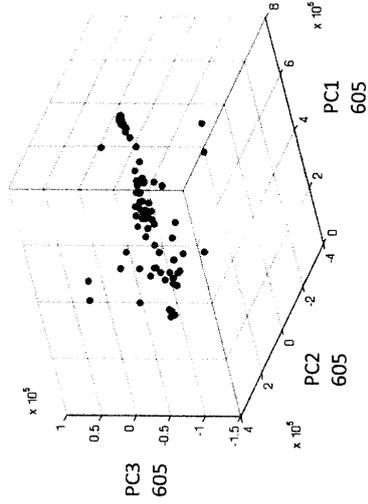
【 図 4 】



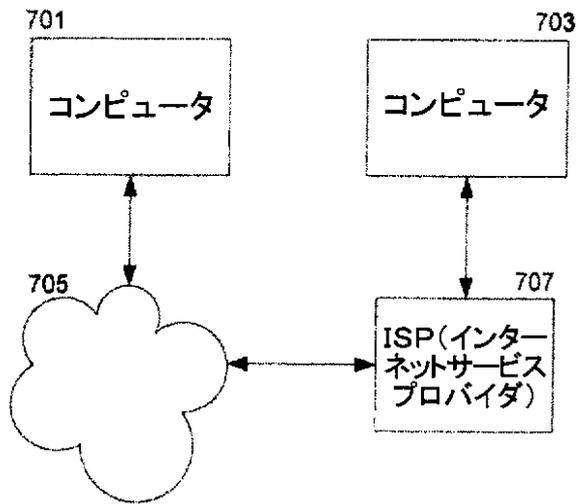
【図5】



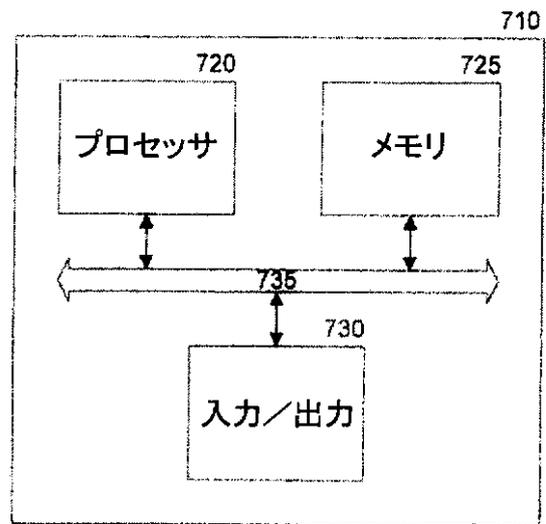
【図6】



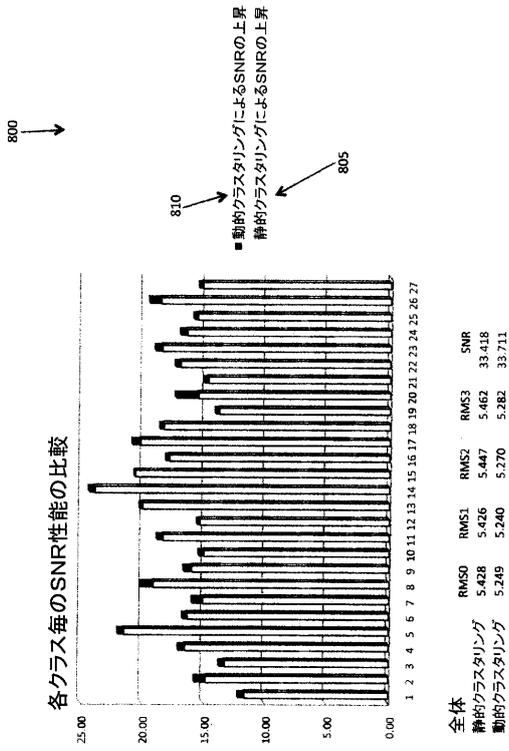
【図7A】



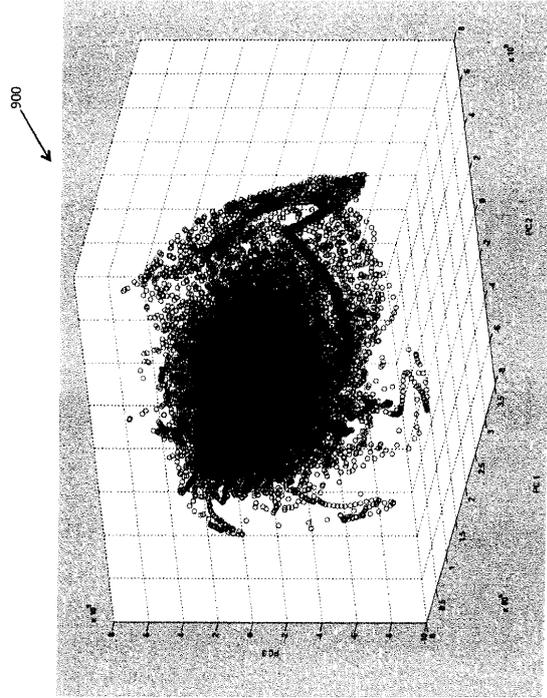
【図7B】



【 図 8 】

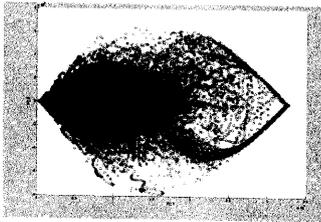


【 図 9 】



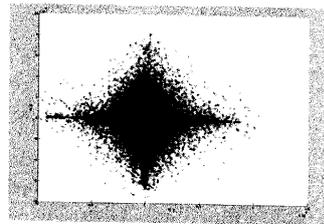
【 図 10 A 】

1005



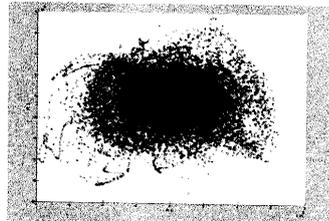
【 図 10 D 】

1020



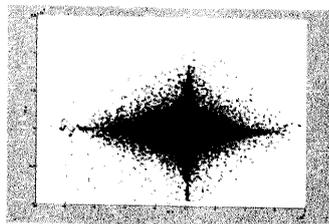
【 図 10 B 】

1010

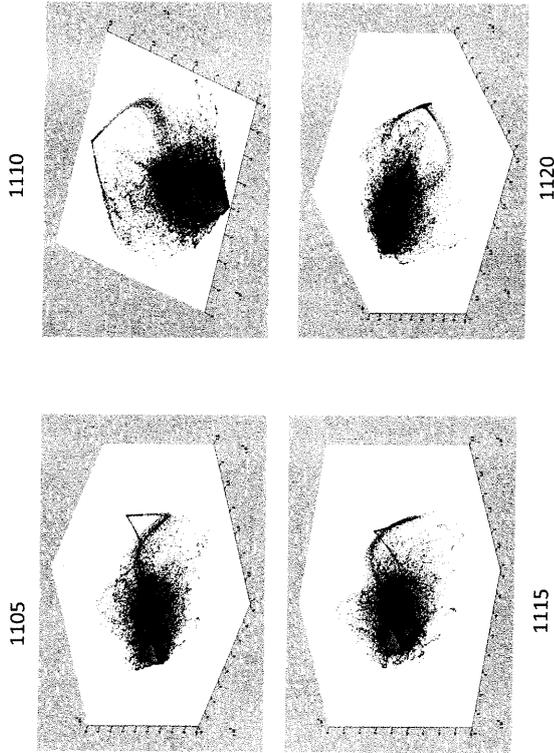


【 図 10 C 】

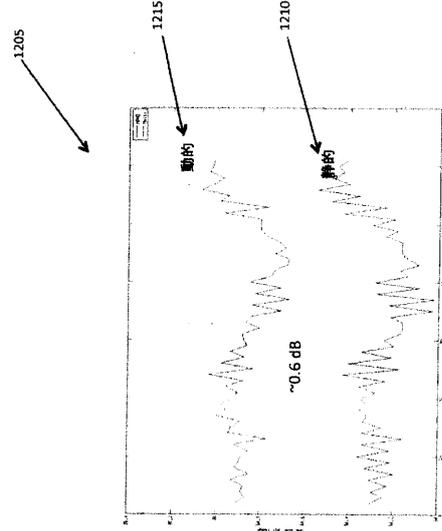
1015



【図 1 1】



【図 1 2】



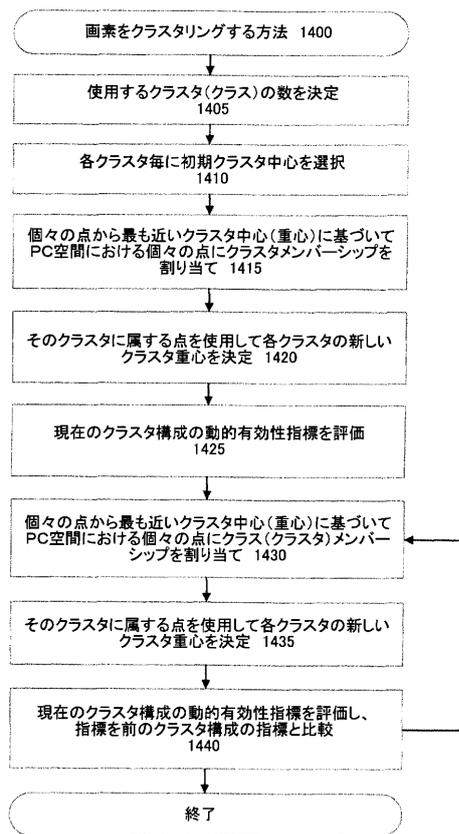
【図 1 3】

SNR性能の比較

ワールド03のSNR

固有ベクトルの数	クラスの数	ランダム初期化による均一なクラスタリング
3	27	34.55
5	27	34.62
7	32	34.63
9	64	34.79
	128	35.03
		34.92
		35.02
		35.30
		35.36
		35.11
		35.35

【図 1 4】



【手続補正書】

【提出日】平成21年12月7日(2009.12.7)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】全文

【補正方法】変更

【補正の内容】

【発明の詳細な説明】

【技術分野】

【0001】

〔関連出願との相互参照〕

本出願は、2008年10月3日に出願された「適応予測フィルタのための動的クラスタリング」という名称の以下の同一出願人による同時係属出願：出願番号第12/245664号（代理人整理番号080398.P738）と関連性がある。

【0002】

本発明は、一般的にビデオ処理に関し、より具体的には、次元縮小法を使用した予測に関する。

【0003】

〔著作権表示/許諾〕

本特許文書の開示部分は、著作権保護の対象内容を含む。著作権の権利所有者は、合衆国特許商標庁の特許ファイル又は記録内に表されるとおりに他者が特許文書又は特許開示を複製することには異議を唱えないが、それ以外は全ての著作権を留保する。以下の表記が適用される：著作権(C)2008, Sony Electronics Inc., 禁無断転載。

【背景技術】

【0004】

ビデオ符号化における根本的な問題は、利用可能な低解像度データを使用した高解像度画像の予測である。通常、これらの場合には最小二乗(LS)フィルタが使用され、このLSフィルタ係数を使用して、場合によっては異なる時点からの利用可能な低解像度データを使用して高解像度画像を作成する。

【0005】

現時点の高解像度画像を取得するために、ズームングが、場合によっては異なる時点からのデータを利用するステップと、利用可能な情報を組み合わせるステップとを必要とする。デシメートした画像を生成し、最良のLSフィルタの組を発見してオリジナル画像と予測画像との間の差分を最小にしたオリジナル画像を再度取得することにより、LSフィルタのトレーニングが可能となる。

【0006】

フィルタタップとは、画素領域に重ねられるパターンのことである。フィルタタップに重なる画素が、その画素に関連するフィルタ係数の基礎を形成する。

【発明の概要】

【課題を解決するための手段】

【0007】

デシメートしたフィールドに基づいてマルチフィールドタップが定められる。マルチフィールドタップを使用して相関行列が生成され、相関行列の要素を使用して共分散行列が生成される。相関行列要素を固有ベクトルに投影することにより、主成分空間が求められる。主成分空間がクラスに区分され、各クラス毎に最小二乗フィルタセットが生成される。

【図面の簡単な説明】

【0008】

【図1A】次元縮小法を使用した適応予測又はビデオ処理の1つの実施形態を示す図であ

る。

【図 1 B】オリジナルのフィールドと、デシメートしたフィールドと、予測フィールドとの間の関係を示す図である。

【図 2】本発明の実施形態による、主成分分析を使用した適応予測の方法を示すフロー図である。

【図 3 A】本発明の実施形態によるデシメーションの例を示す図である。

【図 3 B】本発明の実施形態によるデシメーションの例を示す図である。

【図 4】本発明の実施形態による複数の時点からのフィールドを使用した予測を示す図である。

【図 5】本発明の実施形態による固有ベクトルの分散範囲を示す図である。

【図 6】本発明の実施形態による主成分空間を示す図である。

【図 7 A】本発明の実施に適したコンピュータ環境を示す図である。

【図 7 B】本発明の実施に適したコンピュータ環境を示す図である。

【図 8】本発明の実施形態による分類性能を示す図である。

【図 9】本発明の実施形態による点分布を示す図である。

【図 10 A】本発明の実施形態による別の点分布を示す図である。

【図 10 B】本発明の実施形態による別の点分布を示す図である。

【図 10 C】本発明の実施形態による別の点分布を示す図である。

【図 10 D】本発明の実施形態による別の点分布を示す図である。

【図 11】本発明の実施形態による様々な観点から見たさらに別の点分布を示す図である。

【図 12】本発明の実施形態による別の分類性能を示す図である。

【図 13】本発明の実施形態の様々な構成による性能比較を示す図である。

【図 14】本発明の実施形態による画素をクラスタリングする方法を示すフロー図である。

【発明を実施するための形態】

【0009】

以下の本発明の実施形態の詳細な説明では、同じ参照番号が同様の要素を示す添付の図面を参照し、図面には、本発明を実施できる特定の実施形態を例示目的で示す。これらの実施形態について、当業者が本発明を実施できるように十分に詳しく説明し、また、他の実施形態を利用できるとともに、本発明の範囲から逸脱することなく論理的、機械的、電氣的、機能的及びその他の変更を行うことができる。従って、以下の詳細な説明を限定的な意味で捉えるべきではなく、本発明の範囲は添付の特許請求の範囲によってのみ定められる。

【0010】

本発明の動作の概説から始めると、図 1 A は、次元縮小法を使用した適応予測又はビデオ処理の 1 つの実施形態を示している。オリジナルデータ 100 が、次元縮小法を取り入れた適応予測を使用してエンコーダ 105 で符号化される。オリジナルデータ 100 は、ユーザが装置 120 などの標準画質テレビで見たいと思う、ブルーレイプレーヤーから得たビデオデータであってもよい。これを行うために、エンコーダ 105 は、データ 100 をズームしてズームデータ 110 を生成することができ、これを装置 120 が適当なアスペクト比で復号し、表示することができる。エンコーダ 105 から得られる符号化データは圧縮データ 115 の形を取ることができ、これが（インターネットを介したストリーミングビデオなどのように）復号及び表示するための装置 130 にネットワーク 125 を介して送信される。

【0011】

1 つの実施形態では、エンコーダ 105 はビデオデータのフレームを分類し、個々の予測フィルタをフレーム内の各クラスに割り当てることができる。分類による適応予測は、フレームの画素をいくつかのクラスに区分し、そのクラスにフィルタを適応させるステップを含む。フィルタは、その画素を予測又はデシメートするために使用される各画素の係

数を含むことができる。フィルタ係数及び対応する画素が、フィルタタップを介して関連付けられる。フィルタタップは、対応する画素及びその画素の周囲の画素に重ね合わせられるパターンである。このパターンが、係数の生成にどの画素を使用するかを定める。1つの実施形態では、主成分分析（PCA）を使用してビデオフレームをクラスに分類する。PCAは、フィルタタップが定める対応する相関行列要素で低解像度フレーム内の各画素を表すことができる。PCAは、各画素の相関情報を使用して、画像内の類似した時空間偏差を有する画素をクラスにグループ分けする。

【0012】

分類フィルタの定義は、デシメートされた画像と、その対応するオリジナル解像度の画像との間の関係に依存する。図1Bは、オリジナルフィールドと、デシメートしたフィールドと、予測フィールドとの間の関係を示している。オリジナル（ターゲット）フィールド150、155、及び160がデシメートされ（190）、対応する低解像度フィールド165、170、及び175が得られる。デシメートしたフィールドから利用できる情報を使用して、オリジナルフィールド150～160と予測フィールド180との間の差分を最小にする予測180及びLSフィルタ係数185を得る。予測エラー195は次式のように定義することができる。

$$e = \sum_{\vec{x}} (I(\vec{x}) - P(\vec{x}))^2 \quad (1)$$

この場合、 $I(x)$ はオリジナル画像であり、 $P(x)$ は予測画像であり、 $P(x)$ は次式のように定義される。

$$P(\vec{x}) = \sum_j L_j^u I^D(\vec{y}_j^x) \quad (2)$$

この場合、 $I^D(y)$ はデシメートされた画像を表す。

【0013】

図2は、本発明の実施形態によるPCAの方法200を示す図である。ブロック205において、本方法は、（ハイビジョンなどの）オリジナルの高解像度フィールドをデシメートする。フィールドのデシメーションについては、図3A～図3Bと共に以下でさらに詳細に説明する。

【0014】

ブロック210において、本方法は、デシメートしたフィールドに対するマルチフィールドタップを定める。マルチフィールドフィルタタップは、対応する画素の係数を生成するために使用される複数のデータフィールドにまたがる。マルチフィールドフィルタタップについては、図3B及び図4と共に以下でさらに詳細に説明する。

【0015】

ブロック215において、本方法は、マルチフィールドフィルタタップにおける画素間の相互関係を含む相関行列を生成する。相関行列の要素は、高次元空間におけるデシメートした画像内の画素を表す。この方法は、PCAを使用して低次元空間に問題を引き継ぎ、画素をより効率的に分類する。

【0016】

ブロック220において、本方法は、式3のように相関行列要素の共分散行列を生成する。本方法は、個々のデシメートした画素の相関行列要素を列形式で配置し、各列から平均ベクトルを除去した後でm列を有する観測行列Xを生成する。平均ベクトルの除去により、画素の周囲の局所的変化が強調される。本方法は、Xの共分散行列に固有分解を適用する。この分解はn個の固有ベクトル及びn個の固有値を生み出し、この場合nは相関行列における要素の数である。共分散行列は、画素の数であるmとnとの小さい方である階数nを有する。Xの共分散行列は、次式のように定義することができる。

$$\hat{C}_{xx} = \frac{1}{m-1} X X^T = \frac{1}{m-1} \sum_{i=1}^m (\underline{v}_i - \underline{\hat{\mu}})(\underline{v}_i - \underline{\hat{\mu}})^T \quad (3)$$

この場合、

$$\underline{\hat{\mu}} = \frac{1}{m} \sum_{i=1}^m \underline{v}_i \quad (4)$$

は平均ベクトルである。

【0017】

固有分解動作は、式5のように要約することができる。Qは、列における固有ベクトルを含み、 Λ は対応する固有値：

$$\hat{C}_{xx} = Q \Lambda Q^T = Q \Lambda Q^{-1} \quad (5)$$

を有する対角行列である。

固有ベクトルは正規直交基底を構成するので、 Q^T を Q^{-1} に置き換えることができる。

【0018】

ブロック225において、本方法は、Qの列にXの列を投影することにより固有ベクトルに行列Xを投影して主成分を求め、式6のように主成分の分布を含む空間を生成する。主成分は、Xの個々の列が個々の固有ベクトルにどれほど類似しているかを示す。行列PはPCを含む。個々の画素は、所定の固有ベクトルの数により表される。

$$P = Q^T X$$

(6)

フィルタ係数は、中心画素の周囲のデシメートした画素に対応する相関行列の関数である。フィルタタップはフィルタ係数を識別する。本方法は、対応する相関行列の類似性に基づいてデシメートした画像の画素を分類し、各クラス毎に異なるフィルタ係数の組を生成する。本方法は、個々のデシメートした画像の画素を表すためにフィルタタップの相関情報を使用して固有ベクトルを求め、その固有値に従って固有ベクトルをソートする。

【0019】

方法200は、PCAを使用して、固有ベクトルが定める軸を含む空間に分類問題を変換することができる。この方法は、縮小した次元空間におけるデシメートした画素の座標としてXの列（各列は異なる画素を表す）と固有ベクトルとの間の類似性を使用する。

【0020】

ブロック230において、方法200はPC空間をクラスに区分する。PC空間は、より低次元の空間におけるデシメートした画像の画素を表し、類似の相関行列を含む領域がPC空間に集められる。本方法は、各領域に等しい数の画素が存在するように最初に軸の各々を3つの領域に区分し、この結果、三次元の中に総数27の領域が生じ、画素が27のグループに分類される。他の実施形態では、クラスの数は静的に27とは定められない。異なる数のクラスを選択するステップについては、図14と共に以下で詳細に説明する。

【 0 0 2 1 】

ブロック 2 3 5 において、本方法は、各クラス毎に最小二乗 (L S) フィルタを生成する。L S フィルタは、オリジナル (ターゲット) フレームにおける全画素に渡ってオリジナル画像と予測画像との間の差分の二乗和を最小にする。本方法は、個々のフィルタ要素に関して偏導関数を取り、これを各々についてゼロにすることによりフィルタ係数を求める。

$$e = \sum_{\vec{x}} (I(\vec{x}) - P(\vec{x}))^2$$

(7)

【 0 0 2 2 】

方法 2 0 0 は、図 2 に示すように、フィルタタップが特定した画素の相関関係を使用してフィルタタップにおける係数の組 L を決定する。

$$L = (A)^{-1}b$$

(8)

この場合、

$$A_{j,k} = \sum_{\vec{x}} I^D(\vec{y}_j^{\vec{x}}) I^D(\vec{y}_k^{\vec{x}})$$

$$b_k = \sum_{\vec{x}} I^D(\vec{y}_k^{\vec{x}}) I(\vec{x})$$

(9、10)

この場合、 $A_{j,k}$ は相関行列であり (添字 j 及び k は 2 つのフィルタタップ T の範囲を動く)、 b_k は観測ベクトルである。 I^D はデシメートした画像を表す。

【 0 0 2 3 】

方法 2 0 0 は、観測ベクトル及び相関行列の逆数を使用して L S フィルタを生成する。この方法は、各領域に固有の異なる L S フィルタを使用する。デシメートした画像に m 個の画素と c 個のクラス (領域) が存在すると仮定すると、

$$m = m_1 + m_2 + \dots + m_c$$

(1 1)

が、各クラスにおける画素の数を示す。与えられたクラス u に関して、

$$L^u = (A^u)^{-1}b^u$$

(1 2)

この場合、

$$A_{j,k}^u = \sum_{\vec{x}|u=S(\vec{x})} I^D(\vec{y}_j^{\vec{x}}) I^D(\vec{y}_k^{\vec{x}})$$

$$b_k^u = \sum_{\vec{x}|u=S(\vec{x})} I^D(\vec{y}_k^{\vec{x}}) I(\vec{x})$$

(1 3 、 1 4)

この場合、

$$u = S(\vec{x})$$

(1 5)

が、ターゲット画素 x の予測フィルタ分類マップである。

【 0 0 2 4 】

図 3 A は、1つの期間を使用した空間アルゴリズムのためのデシメーションを示す図である。フィルタタップ 3 0 5 が識別した個々のソース画素が、その値の 1 / 4 をデシメートした画素 3 1 0 の値に与える。

【 0 0 2 5 】

図 3 B は、2つの期間を使用した本発明の実施形態による時空アルゴリズムのためのデシメーションを示す図である。図 3 B に示すデシメーション動作は、予測アルゴリズムへの入力として2又はそれ以上の時間フィールドを使用する。マルチフィールドフィルタタップ 3 5 0 が、その時間フィールドから引き出される画素を決定する。デシメーション動作は、デシメーションを加重係数で均一に分散させてデシメートした画素 3 5 5 を生成する。他の実施形態では、より複雑かつ適応的なデシメーションアルゴリズムが使用される。

【 0 0 2 6 】

図 4 は、3つの異なる時間フィールドから得た3つのフィールド 4 0 5 を使用する時空フィルタタップ例を示す図である。このマルチフィールドフィルタタップは、予測フィルタへの入力として各フィールドからの9つのデシメート画素 4 1 0 を使用して中心画素 4 2 0 の周囲の画素 4 1 5 を予測する。マルチフィールドフィルタタップは、デシメートした画像上を移動して、より高解像度のビデオフィールドを予測する。この方法は、最小二乗法を通じて最適化したフィルタ要素を求めることができる。

【 0 0 2 7 】

図 5 は、本方法が行列形式に戻した後の、減少分散を含む最初の9つの固有ベクトル 5 0 5 を示す図である。パーセンテージ 5 1 0 は、個々の固有ベクトルがオリジナルフィールドの総分散のうちのどの程度を運ぶかを示している。

【 0 0 2 8 】

1つの実施形態では、この方法は、デシメートした画素の各々を3つの座標で表す、三次元空間における最初の3つの固有ベクトルを使用して向上した結果を得る。図 6 は、区分動作に基づくこれらの画素のサブセットの分布を示す図である。軸 PC 1 (6 0 5)、PC 2 (6 0 5)、及び PC 3 (6 0 5) は最初の3つの固有ベクトルに対応し、個々の画素の座標は対応する主成分である。

【 0 0 2 9 】

別の実施形態では、ブロック 2 3 0 において、方法 2 0 0 がクラスタリングを使用して、動的有効性指標を使用した縮小次元空間におけるより細かいグループに画素を分類する。動的有効性指標は、画素クラスの緊密度及びクラス自体の間の距離を測定する費用関数である。クラスタリングの目的は、同じクラスの画素間の距離を最小にすることによりクラスの緊密度を増加させるとともにクラス間の距離を最大にすることである。実験的データが、反復クラスタリングが画素を効率良く分類し、SNRの向上を実現することを示唆

している。

【0030】

図14は、本発明の実施形態による画素クラスをクラスタリングする方法1400を示す図である。クラスタの数 c は、PC空間における画素の分布に関わらず分かっているものとして処理される。ブロック1405において、本方法は、使用するクラスの数を決

【0031】

ブロック1410において、本方法は、個々の画素に数字を付け、この範囲における c の数の均一な分布を選択する。本方法は、クラスタされる c クラスの重心（重み付き中心）の初期値としてこの均一な選択を使用する。

【0032】

ブロック1415において、本方法は、重心の各々までの画素の距離に基づいて、個々の画素を最も近い重心（及びクラスタ）に割り当てる。

【0033】

ブロック1420において、本方法は、画素の割り当てを使用して個々のクラスタの新しいクラスタ重心を決定する。

【0034】

ブロック1425において、本方法は、現在のクラスタリング構成の動的有効性指標を評価する。他の実施形態では、他の費用関数を使用することができる。動的有効性指標は次式のように定義することができる。

$$\text{動的有効性指標} = \frac{\text{Intra}}{\text{Inter}} = \frac{\frac{1}{N} \sum_{i=1}^k \sum_{x \in C_i} \|x - z_i\|^2}{\min_{i,j} (\|z_i - z_j\|)^2} \quad (16)$$

この場合、 N は点（データオブジェクト）の数、 k はクラスタの数、 x はデータ点の位置、 z_i は i 番目のクラスタのクラスタ中心の位置（重心）、及び C_i は i 番目のクラスタにおける点の組である。動的有効性指標は、クラスタ内の項（分子）とクラスタ間の項（分母）とを含む。クラスタ内の項は、各点の対応する重心までの平均距離であり、全クラスタの緊密度の平均レベルを測る。クラスタ間の項は、全部のクラスタのペアの中で互いに最も近いクラスタのペアの間の距離である。

【0035】

ブロック1430において、本方法は、最も近いクラスタ重心に基づいてPC空間における個々の点（画素）に新しいクラスメンバーシップを割り当てる。

【0036】

ブロック1435において、本方法は、新しいクラスの割り当てに基づいて個々のクラスタの新しいクラスタ重心を決定する。

【0037】

ブロック1440において、本方法は、新しい構成の有効性指標を評価し、この有効性指標をブロック1425において生成された有効性指標と比較する。指標間の差がしきい値を超える場合、本方法はブロック1430へ戻り、クラスタリングをさらに繰り返す。しきい値を超えなければ、本方法は現在のクラスタリング方式を使用する。最初の繰り返し中、本方法は、ブロック1440において評価された指標をブロック1425において評価された指標と比較する。次の繰り返し中、本方法は、ブロック1440において評価された指標を比較する。

【0038】

本方法は、各繰り返しにおいて動的有効性指標を評価し、繰り返し間の変化がしきい値よりも下がるまでこれを続行する。これを行っているときには、さらなる割り当ては行われず、方法1400は、現在のクラスタリング方式を使用してPC空間を区分する。実験的データが、結果が飽和し、しきい値よりも下がるまでに20回から30回の間の繰り返

しが必要であることを示唆している。

【0039】

方法1400は、動的有効性指標を使用してクラスタリング構成のための比較点を提供する。本方法は、現在の重心の組に基づいてクラスの割り当てを再帰的に調整し、現在のクラスの割り当てに基づいて重心を調整する。このようにして、本方法は、向上が飽和状態になるまでクラスタリングを繰り返し向上させる。

【0040】

方法1400が図2のブロック230においてクラスタリング動作を終了した後、方法200は、各クラスタの個々のメンバーを使用してLSフィルタ係数を生成し、これが動的にクラスタされたLSフィルタを生成する。

【0041】

図8は、静的クラスタリング805と動的クラスタリング810との信号対雑音比(SNR)性能の比較800を示す図である。実績の赤色の部分は、方法1400などの動的クラスタリング法で得られたクラス固有のLSフィルタを使用した結果生じた利得を示す。

【0042】

クラスタリングを繰り返すことにより、デシメートした画像の画素が分類され、各クラス毎に異なるLS係数の組が生成される。図9は、三次元空間におけるデシメートした画像の画素の分布例900を示す図である。個々の画素を主成分1、2、及び3で表している。

【0043】

図10A~図10Dは、PC1-2 1005、PC2-3 1010、PC3-4 1015、及びPC4-5 1020のペアワイズ分布を表している。これらのペアは、様々な二次元平面への五次元分布の投影を表す。

【0044】

図11は、32個のクラスタに分割された分布の様々な投影1105~1120を示す図である。各クラスタを異なる色調のグレーで表している。

【0045】

図12は、本発明の実施形態を使用して得られたSNRの結果1205を示す図である。最初の5つの固有ベクトルを使用して五次元空間により全てのフィールドに関して(ビデオシーンの間に60回)分析を繰り返した。動的クラスタリングLSフィルタのSNR性能1215を、静的クラスタリングLSフィルタのSNRパフォーマンス1210と比較する。全てのシーンに渡って約0.6dBのNSR性能の上昇が得られた。

【0046】

図13は、クラスタリング方法のパラメータを変更した同様の分析の結果を示す図である。7つのPCまでは、より多くのPC1315を使用したときにSNR向上1305が得られる。分散の99パーセントが、最初の7つのPCに確実に組み入れられる。予想されるように、クラス1310の数を増やした場合に性能が上昇し続けた。

【0047】

図2及び図14に示す一連のフロー図を参照しながら、本発明の特定の方法についてコンピュータソフトウェアの観点から説明した。本方法は、ブロックの形で示す機械実行可能命令で作られたコンピュータプログラムを構成する。フロー図を参照して本方法について説明することにより、当業者は、好適に構成された機械(メモリを含む機械可読媒体からの命令を実行する機械のプロセッサ)において本方法を実行するためのこのような命令を含むこのようなプログラムを開発できるようになる。機械実行可能命令は、コンピュータプログラミング言語で書かれたものであってもよく、或いはファームウェアの論理に統合したものであってもよい。認められている規格に準拠するプログラミング言語で書かれている場合、様々なハードウェアプラットフォームにおいて、及び様々なオペレーティングシステムへのインターフェイスのためにこのような命令を実行することができる。また、ある特定のプログラミング言語に関連して本発明を説明してはいない。様々なプログラ

ミング言語を使用して、本明細書で説明するような本発明の教示を実現できることが理解できよう。さらに、1つの動作を行う場合、或いは1つの結果をもたらす場合に、(プログラム、手順、処理、アプリケーション、モジュール、論理...などの)1つの形又は別の形のソフトウェアについて語るのは当業では一般的なことである。このような表現は、コンピュータがソフトウェアを実行することにより、コンピュータのプロセッサが動作を実行し、或いは結果を生み出すようになるという言い方を簡単にしたものにはすぎない。本発明の範囲から逸脱することなく、図2及び図14に示す方法により多くの又はより少ない処理を組み込むことができること、及び図示するとともに本明細書で説明したブロックの構成により特定の順序を暗示するわけではないことが理解されよう。

【0048】

1つの実施形態では、図7Aに示すように、サーバコンピュータ701がインターネット705に接続され、このインターネット705を介してデータを提供する。クライアントコンピュータ703が、ISP(インターネットサービスプロバイダ)707を介してインターネット705に接続され、従来のインターネットブラウジングアプリケーションを実行してサーバ701とデータを交換する。サーバコンピュータ701は、図2及び図14に示す方法を実行することができる。任意的に、サーバ701は、クライアントシステムにインターネットへのアクセスを提供するISPの一部であってもよい。本明細書で使用する「インターネット」という用語は、TCP/IPプロトコル、及び場合によってはワールドワイドウェブ(web)を構成するハイパーテキストマークアップ言語(HTML)文書のためのハイパーテキスト転送プロトコル(HTTP)のようなその他のプロトコルなどのいくつかのプロトコルを使用するネットワークのネットワークを意味する。インターネットの物理的接続及びインターネットのプロトコル及び通信手順は当業者にとって周知である。インターネットにアクセスすることにより、クライアントコンピュータシステムのユーザは、情報を交換し、電子メールを送受信し、HTMLフォーマットで作成された文書などの文書を閲覧し、コンテンツを受信できるようになる。本発明が、インターネットアクセス及びインターネットウェブベースのサイトに限定されるものではなく、直接接続されたプライベートネットワークも企図されることが容易に明らかである。

【0049】

サーバ701としての使用に適したコンピュータシステムの1つの実施形態を図7Bに示す。コンピュータシステム710は、システムバス735に接続されたプロセッサ720、メモリ725、及び入力/出力機能730を含む。メモリ725は、プロセッサ720により実行された場合、本明細書で説明した方法を実行する命令を記憶するように構成される。メモリ725はまた、次元縮小を使用した適応予測のためのノについてのデータを記憶することもできる。入力/出力730は、次元縮小、或いはその一部又は代表部分を使用した適応予測のためのノについてのデータの配信及び表示を行い、また記憶、処理又は表示のための様々な種類のデータの入力も行う。入力/出力730はまた、プロセッサ720によるアクセスが可能なあらゆる種類の記憶装置を含む様々な種類の機械可読媒体も含む。当業者であれば、サーバ701が、メモリ725で実行するオペレーティングシステムソフトウェアにより制御されることを直ちに認識するであろう。入力/出力730及び関連する媒体は、オペレーティングシステムのための機械実行可能命令及び本発明の方法、並びに次元縮小を使用した適応予測のためのノについてのデータを記憶する。

【0050】

図7A~図7Bの説明は、本発明の実施に適したコンピュータハードウェアおよびその他の動作構成要素の概観を示すことを意図したものであり、利用可能な環境を限定することを意図したものではない。コンピュータシステム740が、異なるアーキテクチャを有する多くの考えられるコンピュータシステムのうちの1つの例であることを理解できよう。典型的なコンピュータシステムは、通常、少なくともプロセッサ、メモリ、及びメモリをプロセッサに接続するバスを含む。当業者であれば、マルチプロセッサシステム、ミニコンピュータ、及びメインフレームコンピュータなどを含むその他のコンピュータシステム構成により本発明を実施できることを直ちに理解するであろう。通信ネットワークを介

してリンクされた遠隔処理装置によってタスクを実行する分散コンピューティング環境において本発明を実施することもできる。

【 0 0 5 1 】

次元縮小を使用した適応予測について説明した。本明細書では特定の実施形態について図示し説明したが、当業者であれば、同じ目的を実現するために計算されたあらゆる構成を、図示の特定の実施形態と置き換えることができることを理解するであろう。本出願は、本発明のあらゆる適応又は変形を対象とすることが意図される。

【 0 0 5 2 】

次元縮小を使用した適応予測に関して本出願で使用した専門用語は、これらの環境の全てを含むことを意図したものである。従って、本発明は、以下の特許請求の範囲及びこの同等物によってのみ限定されることがはっきりと意図される。

フロントページの続き

(74)代理人 100109335

弁理士 上杉 浩

(74)代理人 100151987

弁理士 谷口 信行

(72)発明者 オグス デマーシ

アメリカ合衆国 カリフォルニア州 95125 - 2137 サン ホセ カノアス ガーデン
アベニュー 2118 アpartment #247

(72)発明者 マルコ パニコニ

アメリカ合衆国 カリフォルニア州 95008 キャンベル ユニオン アベニュー 402
ユニット シー

Fターム(参考) 5C159 MA01 SS03 SS08 TA08 TA68 TC02 TC03 TC35 TC42 TD02
TD03 TD04 TD12 TD13 TD16 TD17 UA02 UA11 UA29

【 外国語明細書 】

Adaptive Prediction Using a Dimensionality Reduction Process

RELATED APPLICATIONS

The present application is related to the following commonly-owned, concurrently-filed applications: application serial no. _____ (Attorney Docket No. 080398.P738), filed _____, entitled “Dynamic Clustering for Adaptive Prediction Filters”.

FIELD OF THE INVENTION

This invention relates generally to video processing, and more particularly to predicting using a dimensionality reduction process.

COPYRIGHT NOTICE/PERMISSION

A portion of the disclosure of this patent document contains material which is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever. The following notice applies: Copyright © 2008, Sony Electronics Inc., All Rights Reserved.

BACKGROUND OF THE INVENTION

A fundamental problem in video coding is prediction of high resolution images using available low resolution data. Least square (LS) filters are commonly used in these cases, and the LS filter coefficients are used to produce higher resolution images using available low resolution data, possibly from different times.

Zooming requires utilizing data possibly from different times and combining the available information to obtain a high resolution image for the current time. The training of the LS filters is possible by generating decimated images and finding the best set of LS filters to obtain back the original images minimizing the different between the original and predicted images.

A filter tap is a pattern overlaid on a region of pixels. The pixels overlapping the filter tap form the basis of filter coefficients associated with that pixel.

SUMMARY OF THE INVENTION

Multi-field taps are defined based on a decimated field. The multi-field taps are used to generate correlation matrices, the elements of which are used to generate covariance matrices. A principal component space is obtained by projecting the correlation matrix elements on to eigenvectors. The principal

component space is partitioned into classes and a least square filter set is generated for each class.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1A illustrates one embodiment of adaptive prediction or video processing using a dimensionality reduction process;

Figure 1B illustrates a relationship between original, decimated, and predicted fields;

Figure 2 is a flow diagram illustrating a method of adaptive prediction using principal components analysis according to an embodiment of the invention;

Figures 3A and 3B are diagrams illustrating examples of decimation according to an embodiment of the invention;

Figure 4 is a diagram illustrating prediction using fields from multiple times according to an embodiment of the invention;

Figure 5 is a diagram illustrating eigenvector variance coverage according to an embodiment of the invention;

Figure 6 is a diagram illustrating a principal component space according to an embodiment of the invention;

Figures 7A and 7B are diagrams of a computer environment suitable for practicing the invention;

Figure 8 is a diagram illustrating classification performance according to an embodiment of the invention;

Figure 9 is a diagram illustrating point distribution according to an embodiment of the invention;

Figures 10A-10D illustrate another point distribution according to an embodiment of the invention;

Figure 11 illustrates yet another point distribution from various perspectives according to an embodiment of the invention;

Figure 12 is a diagram illustrating another classification performance according to an embodiment of the invention;

Figure 13 is a diagram illustrating a performance comparison according to various configurations of an embodiment of the invention; and

Figure 14 is a flow diagram illustrating a method of clustering pixels according to an embodiment of the invention.

DETAILED DESCRIPTION OF THE INVENTION

In the following detailed description of embodiments of the invention, reference is made to the accompanying drawings in which like references indicate similar elements, and in which is shown by way of illustration specific embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the

invention, and it is to be understood that other embodiments may be utilized and that logical, mechanical, electrical, functional and other changes may be made without departing from the scope of the present invention. The following detailed description is, therefore, not to be taken in a limiting sense, and the scope of the present invention is defined only by the appended claims.

Beginning with an overview of the operation of the invention, Figure 1A illustrates one embodiment of adaptive prediction or video processing using a dimensionality reduction process. Original data 100 is encoded at encoder 105 using adaptive prediction that incorporates a dimensionality reduction process. The original data 100 may be video data from a BluRay player, which a user wishes to view on a standard definition television such as device 120. To accomplish this, encoder 105 may zoom data 100 to produce zoomed data 110, which device 120 is able to decode and display at the appropriate aspect ratio. The encoded data from encoder 105 may take the form of compressed data 115, which is transmitted through network 125 to device 130 to decode and display (e.g., streaming video over the internet.)

In one embodiment, encoder 105 may classify frames of video data and assign individual prediction filters to each class in the frame. Adaptive prediction with classification involves partitioning the pixels of a frame into a number of classes and adapting a filter to that class. A filter may include coefficients for each pixel which may be used to predict or decimate that pixel. The filter

coefficients and the corresponding pixel are related through a filter tap. A filter tap is a pattern overlaid on the corresponding pixel and pixels around that pixel. The pattern defines which pixels will be used to generate coefficients. In one embodiment, Principal Component Analysis (PCA) is used to classify video frames into classes. The PCA may represent each pixel in a low resolution frame with the corresponding correlation matrix elements defined by the filter taps. PCA uses the correlation information for each pixel to group pixels with similar spatiotemporal variations in the images into classes.

Defining classification filters depends on the relationship between decimated images and their corresponding original resolution images. Figure 1B illustrates a relationship between original, decimated, and predicted fields. Original (target) fields 150, 155, and 160 are decimated and corresponding low resolution fields 165, 170, and 175 are obtained. Information available from the decimated fields is used to obtain prediction 180 and LS filter coefficients 185 that minimize the difference between the original fields 150-160 and the predicted field 180. The prediction error may be defined as:

$$e = \sum_{\vec{x}} (I(\vec{x}) - P(\vec{x}))^2 \quad (1)$$

where $\mathbf{I}(\mathbf{x})$ is the original image and $\mathbf{P}(\mathbf{x})$ is the predicted image, where $\mathbf{P}(\mathbf{x})$ is defined as:

$$P(\vec{x}) = \sum_j L_j^u I^D(\vec{y}_j^{\vec{x}}) \quad (2)$$

where $\mathbf{I}^D(\mathbf{y})$ represents the decimated image.

Figure 2 illustrates a method 200 of PCA according to an embodiment of the invention. At block 205, the method decimates the original high resolution (e.g., high definition) fields. Decimation of fields is described in greater detail below in conjunction with Figures 3A-3B.

At block 210, the method defines multi-field taps over the decimated fields. Multi-field filter taps straddle more than one field of data to be used to generate coefficients for the corresponding pixel. Multi-field filter taps are described in greater detail below in conjunction with Figures 3B and 4.

At block 215, the method generates correlation matrices that include the cross relationships between the pixels in the multi-field filter taps. Elements of the correlation matrices represent the pixels in the decimated image in a high dimensional space. The method carries the problem over to a lower dimensional space using PCA to classify the pixels more efficiently.

At block 220, the method generates a covariance matrix of the correlation matrix elements as in equation 3. The method places the correlation matrix elements for each decimated pixel into a column format and generates an observation matrix \mathbf{X} with \mathbf{m} columns after removing the mean vector from each column. Mean vector removal emphasizes the local changes around the pixels.

The method applies eigendecomposition to the covariance matrix of \mathbf{X} . The decomposition produces \mathbf{n} eigenvectors and \mathbf{n} eigenvalues where \mathbf{n} is the number of elements in the correlation matrices. The covariance matrix has a rank \mathbf{n} which is the smaller of \mathbf{m} and \mathbf{n} , \mathbf{m} being the number of pixels. The covariance matrix of \mathbf{X} may be defined

as:

$$\hat{\mathbf{C}}_{xx} = \frac{1}{m-1} \mathbf{X} \mathbf{X}^T = \frac{1}{m-1} \sum_{i=1}^m (\underline{v}_i - \underline{\hat{\mu}})(\underline{v}_i - \underline{\hat{\mu}})^T \quad (3)$$

where

$$\underline{\hat{\mu}} = \frac{1}{m} \sum_{i=1}^m \underline{v}_i \quad (4)$$

is the mean vector.

Eigendecomposition operation can be summarized as in equation 5. \mathbf{Q} includes the eigenvectors in the columns and Λ is a diagonal matrix with the

$$\hat{\mathbf{C}}_{xx} = \mathbf{Q} \Lambda \mathbf{Q}^T = \mathbf{Q} \Lambda \mathbf{Q}^{-1} \quad (5)$$

corresponding eigenvalues:

\mathbf{Q}^T can be replaced with \mathbf{Q}^{-1} because the eigenvectors constitute an orthonormal basis.

At block 225, the method projects the matrix \mathbf{X} on to eigenvectors by projecting columns of \mathbf{X} on to the columns of \mathbf{Q} to obtain the principal components and generate a space with the distribution of principal components as in equation 6. The principal components indicate how similar each column of \mathbf{X} is to each eigenvector. The matrix \mathbf{P} includes the PCs. Each pixel is represented by a predetermined number of eigenvectors.

$$\mathbf{P} = \mathbf{Q}^T \mathbf{X} \quad (6)$$

Filter coefficients are functions of the correlation matrices corresponding to the decimated pixels around the center pixel. The filter tap identifies the filter coefficients. The method classifies the decimated image pixels based on the similarities of the corresponding correlation matrices and generates a different set of filter coefficients for each class. The method obtains eigenvectors using correlation information of a filter tap to represent each decimated image pixel and sorts the eigenvectors according to their eigenvalues.

The method 200 may use PCA to transform the classification problem into a space with axes defined by the eigenvectors. The method uses the similarity between columns of \mathbf{X} (each column represents a different pixel) and the eigenvectors as the coordinates of the decimated pixels in the reduced dimensional space.

At block 230, the method 200 partitions the PC space into classes. PC space represents the decimated image pixels in the lower dimensional space and regions with similar correlation matrices are grouped together in the PC space. The method initially partitions each of the axes into three regions such that there will be equal number of pixels in each region, which results in a total of twenty-seven regions in three dimensions, with the pixels classified into twenty-seven groups. In other embodiments, the number of classes is not statically defined as 27. Selecting different numbers of classes is described in greater detail below in conjunction with Figure 14.

At block 235, the method generates a least squares (LS) filter for each class. The LS filter minimizes the sum of the squared difference between the original and predicted images over all pixels in the original (target) frame. The method obtains the filter coefficients by taking the partial derivatives with respect to each filter element and forcing it to be zero for each:

$$e = \sum_{\vec{x}} (I(\vec{x}) - P(\vec{x}))^2 \quad (7)$$

The method 200, as illustrated in Figure 2, determines the set of coefficients in the filter taps \mathbf{L} using the correlations of the pixels determined by the filter taps:

$$L = (A)^{-1}b \quad (8)$$

where

$$\begin{aligned} A_{j,k} &= \sum_{\vec{x}} I^D(\vec{y}_j^{\vec{x}}) I^D(\vec{y}_k^{\vec{x}}) \\ b_k &= \sum_{\vec{x}} I^D(\vec{y}_k^{\vec{x}}) I(\vec{x}) \end{aligned} \quad (9, 10)$$

where $A_{j,k}$ is the correlation matrix (indices j and k run over two filter taps T) and b_k is the observation vector. I^D represents the decimated image.

The method 200 generates the LS filter using an observation vector and an inverse of the correlation matrices. The method uses different LS filters specific to each region. Assuming that there are m pixels and c classes (regions) in the decimated image,

$$m = m_1 + m_2 + \cdots + m_c \quad (11)$$

shows the number of pixels in each class. For a given class u ,

$$L^u = (A^u)^{-1}b^u \quad (12)$$

where

$$A_{j,k}^u = \sum_{\vec{x}|u=S(\vec{x})} I^D(\vec{y}_j^{\vec{x}}) I^D(\vec{y}_k^{\vec{x}})$$

$$b_k^u = \sum_{\vec{x}|u=S(\vec{x})} I^D(\vec{y}_k^{\vec{x}}) I(\vec{x})$$
(13, 14)

where

$$u = S(\vec{x})$$
(15)

is the prediction filter classification map for the target pixel \mathbf{x} .

Figure 3A is a diagram illustrating decimation for spatial algorithms using a single time period. Each source pixel identified by filter tap 305 contributes one-quarter of its value to the value of decimated pixel 310.

Figure 3B is a diagram illustrating decimation for a spatiotemporal algorithm according to an embodiment of the invention using two time periods. The decimation operation illustrated in Figure 3B uses two or more time fields as input to a prediction algorithm. A multi-field filter tap 350 determines which pixels are drawn from which time fields. The decimation operation uniformly distributes the decimation with weighted coefficients to generate decimated pixel 355. In other embodiments, more complex and adaptive decimation algorithms are used.

Figure 4 is a diagram illustrating a sample spatiotemporal filter tap that uses three fields 405 from three different time fields. The multi-field filter tap

uses nine decimated pixels 410 from each field as input to the prediction filter to predict pixels 415 around center pixel 420. The multi-field filter tap moves over the decimated image to predict a higher resolution video field. The method may obtain optimized filter elements through a least squares process.

Figure 5 illustrates the first nine eigenvectors 505 with the decreasing variance after the method places them back into matrix form. The percentages 510 indicate how much of the total variance of the original field each eigenvectors carries.

In one embodiment, the method obtains improved results using the first three eigenvectors in a three-dimensional space, representing each of the decimated pixels with three coordinates. Figure 6 illustrates the distribution of a subset of these pixels based on a partition operation. The axes 605, 610, and 615 correspond to the first three eigenvectors and coordinates of each pixel are the corresponding principal components.

In another embodiment, at block 230 the method 200 uses clustering to classify pixels into finer groups in the reduced dimensional space using a dynamic validity index. The dynamic validity index is a cost function that measures the compactness of a pixel class and the distance between the classes themselves. The goal of clustering is to minimize the distance between pixels in the same class, which increases the compactness of the class, and maximizing the distance

between classes. Empirical data suggests that iterative clustering classifies the pixels efficiently and provides an SNR improvement.

Figure 14 illustrates a method 1400 of clustering pixel classes according to an embodiment of the invention. The number of clusters, c , is treated as known regardless of the distribution of the pixels in the PC space. At block 1405, the method determines a number of classes to use.

At block 1410, the method labels each pixel with a number and selects a uniform distribution of c numbers in this range. The method uses this uniform selection as initial values for the centroids (weighted centers) of the c classes to be clustered.

At block 1415, the method assigns each pixel to the closet centroids (and cluster) based on the distance of the pixel to each of the centroids.

At block 1420, the method determines new cluster centroids for each cluster using the pixel assignments.

At block 1425, the method evaluates the dynamic validity index of the current clustering configuration. In other embodiments, other cost functions may be used. The dynamic validity index may be defined as:

$$\text{Dynamic Validity Index} = \frac{\text{Intra}}{\text{Inter}} = \frac{\frac{1}{N} \sum_{i=1}^k \sum_{x \in C_i} \|x - z_i\|^2}{\min_{i,j} (\|z_i - z_j\|)^2}$$

(16)

where N is the number of points (data objects), k is the number of clusters, x is the location of a data point, z_i is the cluster center location (centroid) of the i^{th} cluster and C_i is the set of points in the i^{th} cluster. The dynamic validity index includes an intra-cluster term (numerator) and an inter-cluster term (denominator). The intra-cluster term is the average distance of each point to their corresponding centroid and measures the average level of compactness of all clusters. The inter-cluster term is the distance between the pair of clusters that are closest to each other among all pairs of clusters.

At block 1430, the method assigns new class membership to each point (pixel) in the PC space based on the closest cluster centroid.

At block 1435, the method determines new cluster centroids of each cluster based on the new class assignments.

At block 1440, the method evaluates the validity index of the new configuration, and compares this validity index with the validity index generated at block 1425. If the difference between the indexes exceeds a threshold, the method returns to block 1430 and goes through an additional iteration of clustering. Otherwise, the method uses the current clustering scheme. During the first iteration, the method compares the index evaluated at block 1440 with the index evaluated at block 1425. During subsequent iterations, the method compares the index evaluated at block 1440

The method evaluates the dynamic validity index at each iteration and continues until the change between iterations drops below a threshold. When this occurs, no further assignments are made and the method 1400 uses the current clustering scheme to partition the PC space. Empirical data suggests that between twenty and thirty iterations are required before the results saturate and drop below the threshold.

The method 1400 uses the dynamic validity index to provide a point of comparison for clustering configurations. The method recursively adjusts the class assignments based on the current set of centroids and adjusts the centroids based on the current class assignments. In this way, the method iteratively improves the clustering until the improvement saturates.

After the method 1400 completes the clustering operation at block 230 of Figure 2, the method 200 uses each member of each cluster to generate the LS filter coefficients, which produces a dynamically clustered LS filter.

Figure 8 illustrates a comparison 800 of the signal-to-noise ratio (SNR) performance of static clustering 805 and dynamic clustering 810. Red portions of the performance indicate the gain resulting from use of class specific LS filters derived with a dynamic clustering method such as method 1400.

Iterative clustering classifies pixels in the decimated images and generates a different set of LS coefficients for each class. Figure 9 illustrates a sample

distribution 900 of pixels of a decimated image in three-dimensional space. Each pixel is represented with principal components one, two, and three.

Figures 10A-10D represent a pair-wise distribution of PC1-2 1005, PC2-3 1010, PC3-4 1015, and PC4-5 1020. These pairs represent projections of a five dimensional distribution on to various two dimensional planes.

Figure 11 illustrates various perspectives 1105-1120 of a distribution divided into thirty-two clusters. Each cluster is represented by a different tone of gray.

Figure 12 illustrates the SNR results 1205 obtained using an embodiment of the invention. Analyses were repeated for every field (sixty times during the video scene) with five dimensional spaces using the first five eigenvectors. SNR performance 1215 of the dynamic clustering LS filter is compared with the SNR performance 1210 of static clustering LS filters. Approximately 0.6dB NSR performance increase was obtained over all scenes.

Figure 13 illustrates the results of similar analyses varying parameters of the clustering method. SNR improvements 1305 are obtained as more PCs 1315 are used up until seven PCs. Ninety-nine percent of variance is demonstrably incorporated into the first seven PCs. Performance continued to increase when increasing the number of classes 1310, as would be expected.

The particular methods of the invention are described in terms of computer software with reference to a series of flow diagrams illustrated in

Figures 2 and 14. The methods constitute computer programs made up of machine-executable instructions illustrated as blocks. Describing the methods by reference to a flow diagram enables one skilled in the art to develop such programs including such instructions to carry out the methods on suitably configured machines (the processor of the machine executing the instructions from machine-readable media, including memory.) The machine-executable instructions may be written in a computer programming language or may be embodied in firmware logic. If written in a programming language conforming to a recognized standard, such instructions can be executed on a variety of hardware platforms and for interface to a variety of operating systems. In addition, the present invention is not described with reference to any particular programming language. It will be appreciated that a variety of programming languages may be used to implement the teachings of the invention as described herein.

Furthermore, it is common in the art to speak of software, in one form or another (e.g., program, procedure, process, application, module, logic...), as taking an action or causing a result. Such expressions are merely a shorthand way of saying that execution of the software by a computer causes the processor of the computer to perform an action or produce a result. It will be appreciated that more or fewer processes may be incorporated into the methods illustrated in Figures 2 and 14 without departing from the scope of the invention and that no particular order is implied by the arrangement of blocks shown and described herein.

In one embodiment, as shown in Figure 7A, a server computer 701 is coupled to, and provides data through, the Internet 705. A client computer 703 is coupled to the Internet 705 through an ISP (Internet Service Provider) 705 and executes a conventional Internet browsing application to exchange data with the server 701. A server computer 701 may execute the methods illustrated in Figures 2 and 14. Optionally, the server 701 can be part of an ISP which provides access to the Internet for client systems. The term "Internet" as used herein refers to a network of networks which uses certain protocols, such as the TCP/IP protocol, and possibly other protocols such as the hypertext transfer protocol (HTTP) for hypertext markup language (HTML) documents that make up the World Wide Web (web). The physical connections of the Internet and the protocols and communication procedures of the Internet are well known to those of skill in the art. Access to the Internet allows users of client computer systems to exchange information, receive and send e-mails, view documents, such as documents which have been prepared in the HTML format, and receive content. It is readily apparent that the present invention is not limited to Internet access and Internet web-based sites; directly coupled and private networks are also contemplated.

One embodiment of a computer system suitable for use as server 701 is illustrated in Figure 7B. The computer system 710, includes a processor 720, memory 725 and input/output capability 730 coupled to a system bus 735. The

memory 725 is configured to store instructions which, when executed by the processor 720, perform the methods described herein. The memory 725 may also store data for/of adaptive prediction using dimensionality reduction. Input/output 730 provides for the delivery and display of the data for/of adaptive prediction using dimensionality reduction or portions or representations thereof, and also the input of data of various types for storage, processing or display. Input/output 730 also encompasses various types of machine-readable media, including any type of storage device that is accessible by the processor 720. One of skill in the art will immediately recognize that the server 701 is controlled by operating system software executing in memory 725. Input/output 730 and related media store the machine-executable instructions for the operating system and methods of the present invention as well as the data for/of adaptive prediction using dimensionality reduction.

The description of Figures 7A-B is intended to provide an overview of computer hardware and other operating components suitable for implementing the invention, but is not intended to limit the applicable environments. It will be appreciated that the computer system 740 is one example of many possible computer systems which have different architectures. A typical computer system will usually include at least a processor, memory, and a bus coupling the memory to the processor. One of skill in the art will immediately appreciate that the invention can be practiced with other computer system configurations, including

multiprocessor systems, minicomputers, mainframe computers, and the like. The invention can also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network.

Adaptive prediction using dimensionality reduction has been described. Although specific embodiments have been illustrated and described herein, it will be appreciated by those of ordinary skill in the art that any arrangement which is calculated to achieve the same purpose may be substituted for the specific embodiments shown. This application is intended to cover any adaptations or variations of the present invention.

The terminology used in this application with respect to adaptive prediction using dimensionality reduction is meant to include all of these environments. Therefore, it is manifestly intended that this invention be limited only by the following claims and equivalents thereof.

CLAIMS

What is claimed is:

1. A computerized method comprising:
 - defining multi-field taps based on a decimated field;
 - generating correlation matrixes using the multi-field taps;
 - generating covariance matrixes based on the correlation matrix elements;
 - projecting the correlation matrix elements onto eigenvectors to obtain a principal component space;
 - partitioning the principal component space into classes; and
 - generating a least square filter set for each of the classes.
2. The computerized method of claim 1, further comprising:
 - decimating an original high resolution field to produce a decimated field.
3. The computerized method of claim 2, further comprising:
 - reconstructing the original high resolution field for display on a display device.

4. A computer-readable storage medium storing instructions which when executed by a data processing system cause the data processing system to perform operations that process data, the operations comprising:

defining multi-field taps based on a decimated field;

generating correlation matrixes using the multi-field taps;

generating covariance matrixes based on the correlation matrix elements;

projecting the correlation matrix elements onto eigenvectors to obtain a principal component space;

partitioning the principal component space into classes; and

generating a least square filter for each of the classes.

5. The computer-readable storage medium of claim 4, the method further comprising:

decimating an original high resolution field to produce a decimated field.

6. The computer-readable storage medium of claim 5, the method further comprising:

reconstructing the original high resolution field for display on a display device.

7. A system comprising:

a processor coupled to a bus;
a memory coupled to the processor through the bus; and
instructions executed by the processor from the memory to cause the
processor to

- define multi-field taps based on a decimated field;
- generate correlation matrixes using the multi-field taps;
- generate covariance matrixes based on the correlation matrix
elements;
- project the correlation matrix elements onto eigenvectors to obtain
a principal component space;
- partition the principal component space into classes; and
- generate a least square filter for each of the classes.

7. The system of claim 7, the instructions further comprising:
decimate an original high resolution field to produce a decimated field.
8. The system of claim 8, the instructions further comprising:
reconstruct the original high resolution field for display on a display
device.
9. A data processing system comprising:

means for defining multi-field taps based on a decimated field;

means for generating correlation matrixes using the multi-field taps;

means for generating covariance matrixes based on the correlation matrix elements;

means for projecting the correlation matrix elements onto eigenvectors to obtain a principal component space;

means for partitioning the principal component space into classes; and

means for generating a least square filter for each of the classes.

ABSTRACT OF THE DISCLOSURE

Multi-field taps are defined based on a decimated field. The multi-field taps are used to generate correlation matrices, the elements of which are used to generate covariance matrices. A principal component space is obtained by projecting the correlation matrix elements on to eigenvectors. The principal component space is partitioned into classes and a least square filter set is generated for each class.

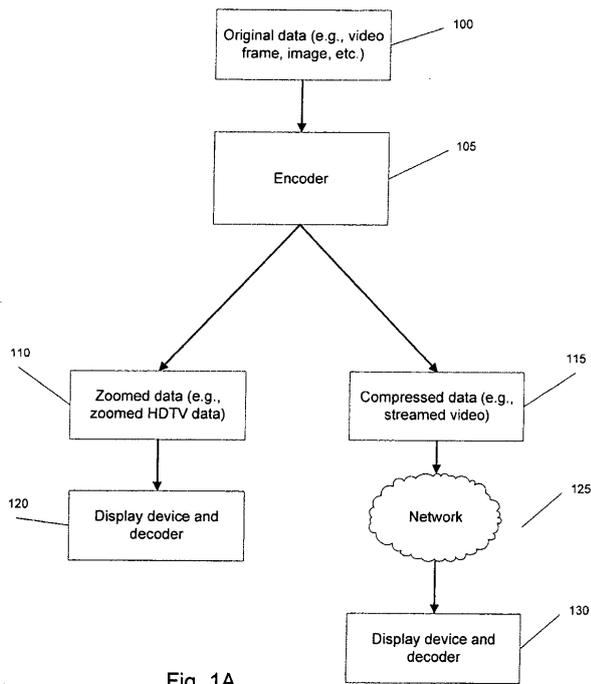
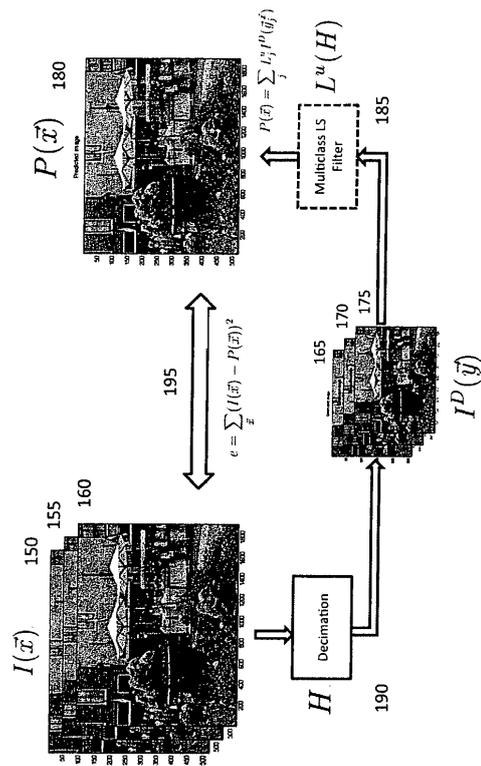


Fig. 1A

Figure 1B



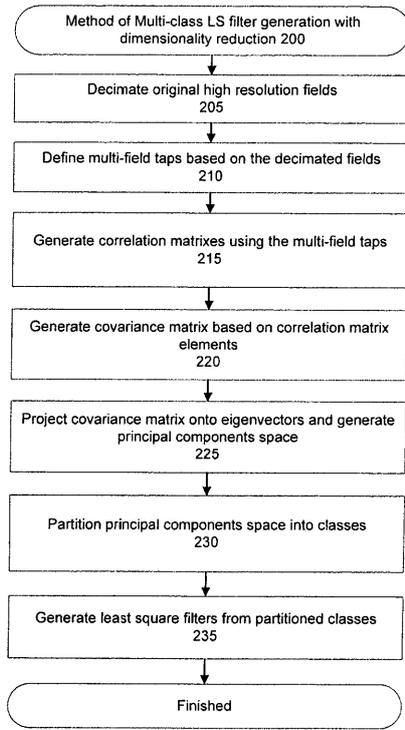


Fig. 2

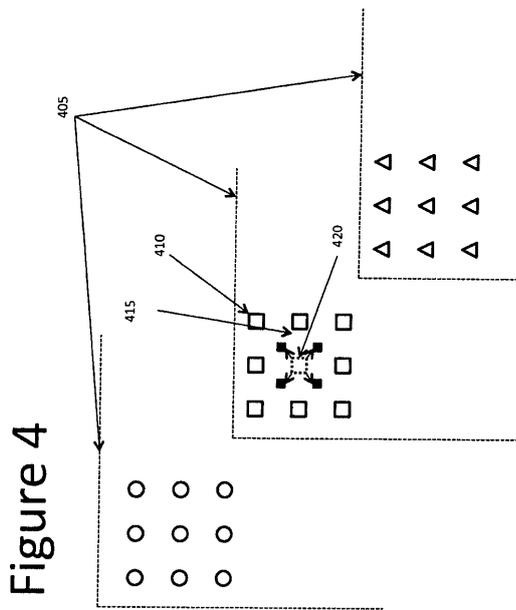


Figure 4

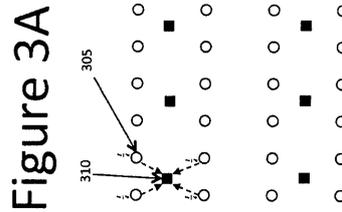
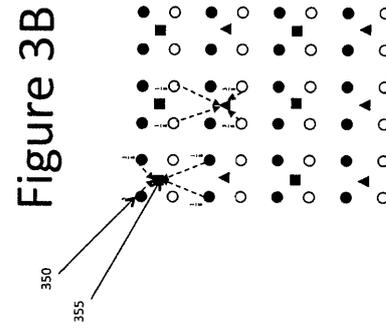


Figure 3A

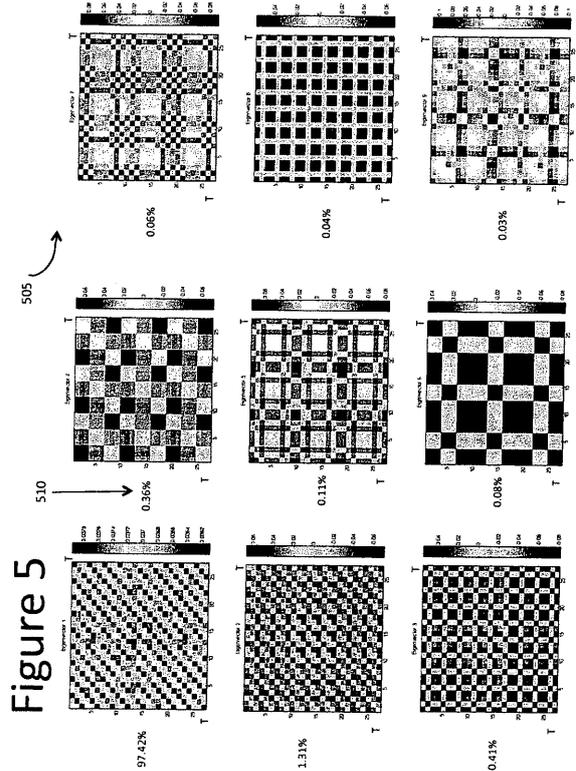


Figure 5

Figure 6

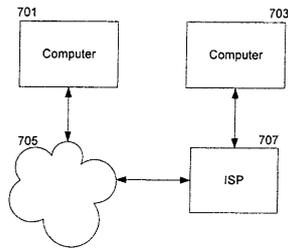
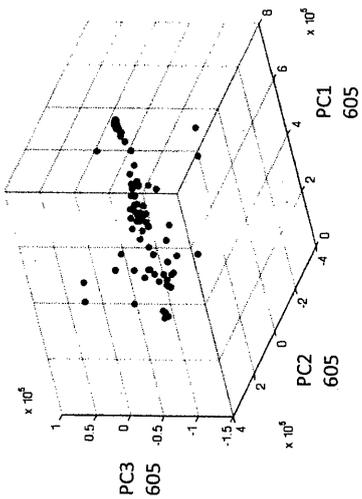


Fig. 7A

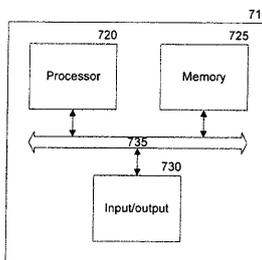
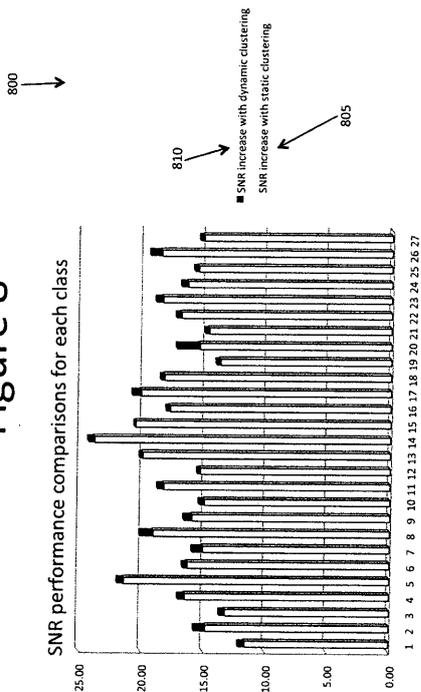


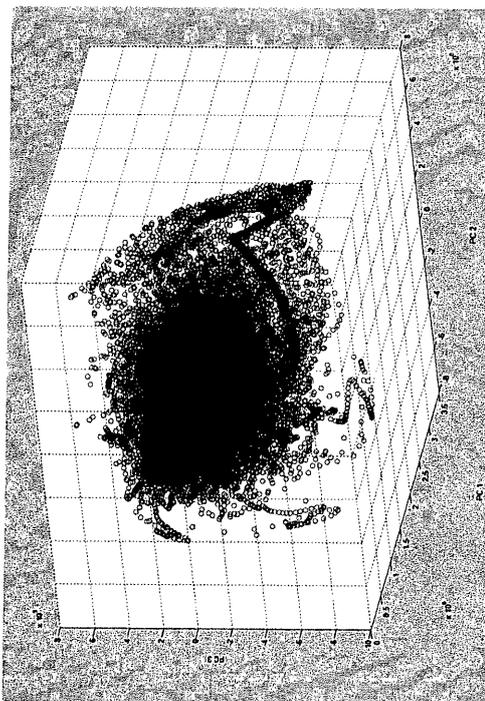
Fig. 7B

Figure 8



Overall	RMS0	RMS1	RMS2	RMS3	SNR
Static Clustering	5.428	5.426	5.447	5.462	33.418
Dynamic Clustering	5.249	5.240	5.270	5.282	33.711

Figure 9



Figures 10A-10D

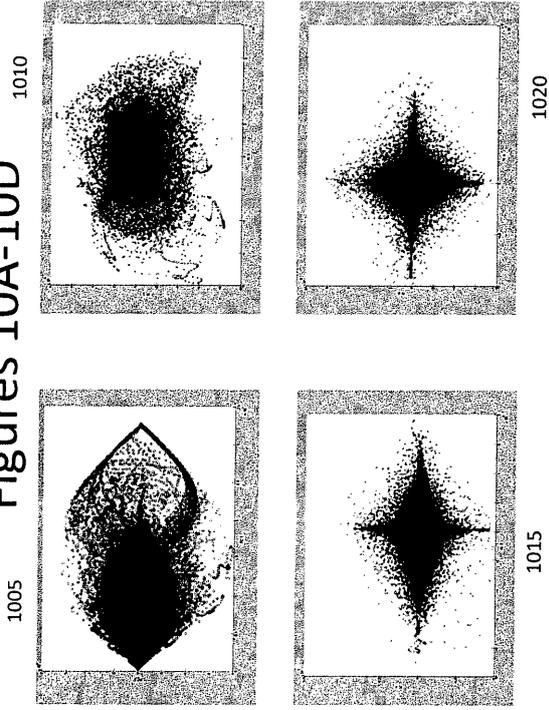


Figure 11

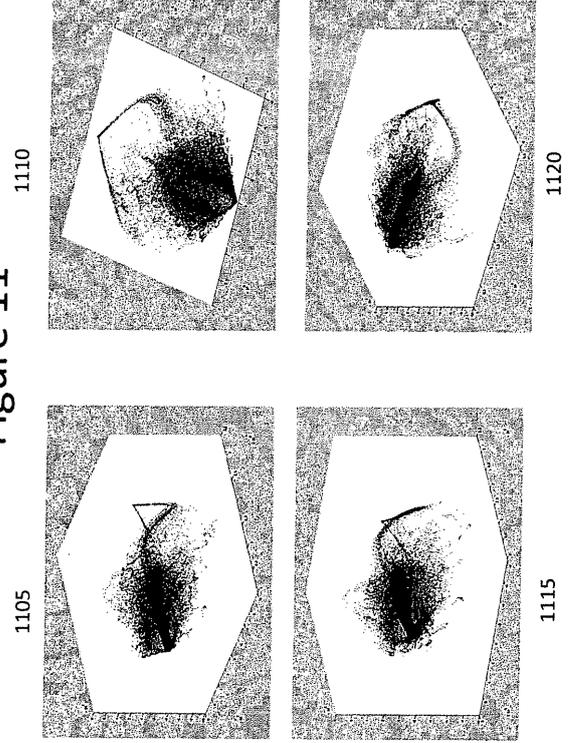


Figure 12

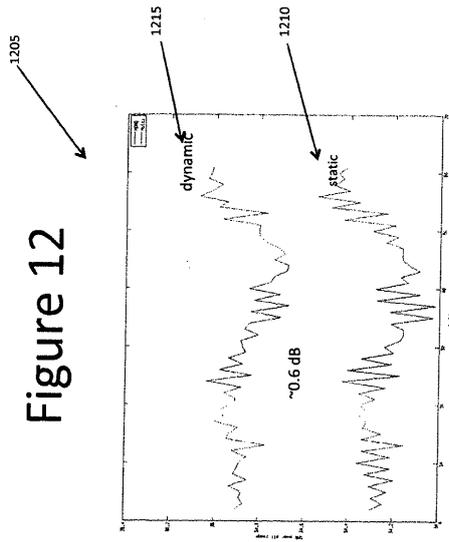


Figure 13

SNR performance comparison

SNR for field 03	# of eigenvectors	Number of classes			
		Uniform	Clustering	\w random initialization	128
3	27	34.55	34.62	34.63	34.79
5	27	n/a	34.87	34.92	35.02
7	32	n/a	34.93	34.95	35.12
9	64	n/a	34.90	34.86	35.11
	128				35.30
	35-03				35.36
	35-35				35.35

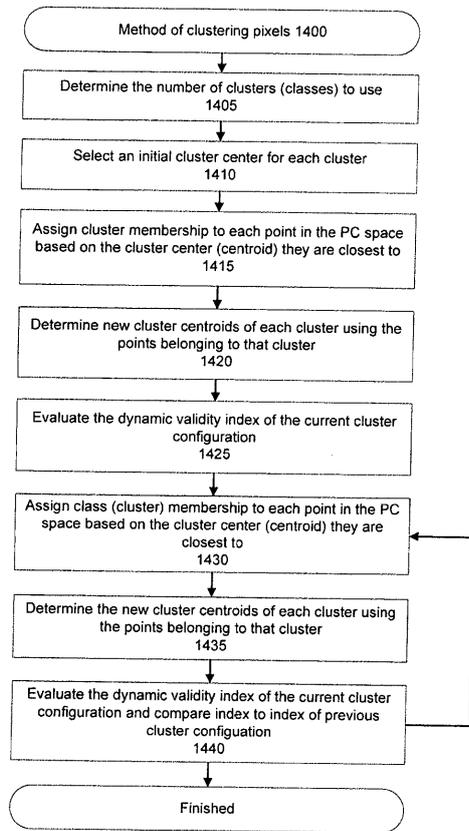


Fig. 14