(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第6248774号
(P6248774)

(45)発行日　平成29年12月20日(2017.12.20)　(24)登録日　平成29年12月1日(2017.12.1)

(51)Int.Cl.　F I
G06F 17/30 (2006.01)　G06F 17/30 350C
G06F 17/30 220Z

請求項の数 5　(全 16 頁)

(21)出願番号　特願2014-85624(P2014-85624)
(22)出願日　平成26年4月17日(2014.4.17)
(65)公開番号　特開2015-207055(P2015-207055A)
(43)公開日　平成27年11月19日(2015.11.19)
審査請求日　平成29年1月10日(2017.1.10)

(73)特許権者　000005223
富士通株式会社
神奈川県川崎市中原区上小田中４丁目１番１号
(74)代理人　100103528
弁理士　原田　一男
(72)発明者　武部　浩明
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内
(72)発明者　上原　祐介
神奈川県川崎市中原区上小田中４丁目１番１号　富士通株式会社内

審査官　齊藤　貴孝

最終頁に続く

(54)【発明の名称】最近傍探索のための情報処理装置及び方法

(57)【特許請求の範囲】
【請求項１】
複数のデータサンプルを格納するデータ格納部と、
前記データ格納部に格納された前記複数のデータサンプル間についての類似度行列に、前記複数のデータサンプルから選択された特定のデータサンプルとの類似度に基づく前記複数のデータサンプル間の類似度行列が近似するように、前記特定のデータサンプルを抽出する抽出部と、
を有する情報処理装置。
【請求項２】
前記抽出部が、
前記複数のデータサンプル間についての類似度行列を算出する類似度行列算出部と、
算出された前記類似度行列に対して固有値分解を行って固有ベクトルを生成する固有値分解処理部と、
生成された前記固有ベクトルの各成分の絶対値に基づき、前記複数のデータサンプルから前記特定のデータサンプルを特定する特定部と、
を有する請求項１記載の情報処理装置。
【請求項３】
前記特定のデータサンプルを用いて前記複数のデータサンプルの各々についてハッシュベクトルを算出する算出部と、
算出された前記ハッシュベクトルと、前記データサンプルについてのカテゴリ名とを対

応付けてデータベースに登録する登録部と、
あるデータについて前記特定のデータサンプルを用いて算出されたハッシュベクトルと前記データベースに登録されたハッシュベクトルとの距離を算出して、最短距離のハッシュベクトルに対応付けられたカテゴリ名を特定する処理部と、
をさらに有する請求項１又は２記載の情報処理装置。
【請求項４】
複数のデータサンプルを格納するデータ格納部に格納された前記複数のデータサンプル間についての類似度行列に、前記複数のデータサンプルから選択された特定のデータサンプルとの類似度に基づく前記複数のデータサンプル間の類似度行列が近似するように、前記特定のデータサンプルを抽出する
処理を含み、コンピュータが実行する情報処理方法。
【請求項５】
複数のデータサンプルを格納するデータ格納部に格納された前記複数のデータサンプル間についての類似度行列に、前記複数のデータサンプルから選択された特定のデータサンプルとの類似度に基づく前記複数のデータサンプル間の類似度行列が近似するように、前記特定のデータサンプルを抽出する
処理を、コンピュータに実行させるためのプログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、最近傍探索技術に関する。
【背景技術】
【０００２】
最近傍探索は、データが分布するｎ次元ベクトル空間において、与えられた点に最も近いデータを探す技術である。
【０００３】
高次元データ空間における最近傍探索技術として、高次元データ空間を低次元データ空間に写像することにより、大規模データを小さく圧縮してから最近傍探索を行う方法が有用であることが近年示されている。
【０００４】
このような方法において、元のデータ集合における距離関係を保存したまま低次元データ空間に写像するラプラシアン固有マップ法（例えば非特許文献１）を用いることが好ましい。
【０００５】
しかし、ラプラシアン固有マップ法を用いる場合、入力される未知のデータを低次元データ空間に写像するときに、未知のデータが低次元データ空間においてどこに写像されるかを知るために、未知のデータと既知のすべてのデータサンプルとの類似度計算を行うことになる。そのため、ラプラシアン固有マップ法を最近傍探索にそのまま適用すると、処理時間が膨大になってしまうという問題があった。
【０００６】
これに対して、Anchor Graph Hashing（例えば非特許文献２）という方法は、データ集合からアンカーと呼ばれるｋ個のデータサンプルを選んだ後に、各データサンプルを、それから最も近いアンカーからの距離（又は類似度）で表す。これにより、入力された未知のデータに対してもアンカーとの距離を計測することで低次元データ空間に写像することができる。すなわち、未知のデータを低次元データ空間に写像する場合、未知のデータとアンカーとの類似度計算を行えばよいので、処理時間の問題を解決できる。
【０００７】
しかしながら、アンカーをｋ－ｍｅａｎｓ法（Ｋ－平均法）によって選択しており、データの分布の形状によっては最近傍探索の精度が低下する問題があった。
【０００８】

ｋ－ｍｅａｎｓ法は、データサンプル群をクラスタに分類する代表的な方法として知られている。具体的なアルゴリズムは以下のとおりである。
１．各データサンプルに対してランダムにクラスタを割り振る。
２．割り振ったデータサンプルを基に各クラスタの中心を計算する。
３．各データサンプルについて各クラスタ中心との距離を求め、最も近いクラスタ中心のクラスタに割り当て直す。
４．上記の処理で全てのデータサンプルについてクラスタの割り当てが変化しなかったら終了する。そうでない場合は２及び３の処理を繰り返す。
【０００９】
この方法によってアンカーが選ばれる場合、データサンプルの分布の形状によっては最近傍探索の精度が低下する問題がある。
【００１０】
例えば、図１に模式的に示すような特徴空間におけるデータサンプル群（×印）があったとき、ｋ－ｍｅａｎｓ法でアンカーを算出すると、アンカーａは分布の中心付近となる。そうすると、各データサンプルはアンカーａからの距離がほぼ等しくなるので、アンカーを経由した類似度は、どれもほぼ同じになる。従って、図２に模式的に示すように、各データサンプルを低次元特徴空間に写像すると、これらのデータサンプルはおおよそ一カ所に密集した状態となる。
【００１１】
一方、図３に示すように、特徴空間において未知のデータＸが入力されて、未知のデータＸに対する最近傍点を求めるものとする。このとき、図４に模式的に示すように、未知のデータＸを低次元特徴空間に写像すると、密集したデータ集合の中に写像されるため、誤った点を最近傍点として抽出する可能性が高くなる。図５に示すように、本来特徴空間では、未知のデータＸはデータサンプルＺに最も近いが、低次元特徴空間では、アンカーａを経由した距離が同じであるデータサンプルＹが最も近いデータサンプルとして抽出されてしまう。
【先行技術文献】
【非特許文献】
【００１２】
【非特許文献１】“Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering” M. Belkin and P. Niyogi. NIPS 14, 2002
【非特許文献２】“Hashing with Graphs” W.Liu, J.Wang, S.Kumar and S.F.Chang. ICML’11, 2011
【発明の概要】
【発明が解決しようとする課題】
【００１３】
従って、本発明の目的は、一側面によれば、最近傍探索における未知のデータに対する識別精度を向上させるための技術を提供することである。
【課題を解決するための手段】
【００１４】
本発明に係る情報処理装置は、（Ａ）複数のデータサンプルを格納するデータ格納部と、（Ｂ）データ格納部に格納された複数のデータサンプル間についての類似度行列に、複数のデータサンプルから選択された特定のデータサンプルとの類似度に基づく複数のデータサンプル間の類似度行列が近似するように、特定のデータサンプルを抽出する抽出部とを有する。
【発明の効果】
【００１５】
一側面としては、最近傍探索における未知のデータに対する識別精度を向上させることができるようになる。
【図面の簡単な説明】

【００１６】
【図１】図１は、従来技術の問題を説明するための図である。
【図２】図２は、従来技術の問題を説明するための図である。
【図３】図３は、従来技術の問題を説明するための図である。
【図４】図４は、従来技術の問題を説明するための図である。
【図５】図５は、従来技術の問題を説明するための図である。
【図６】図６は、本実施の形態の概要を説明するための図である。
【図７】図７は、本実施の形態の概要を説明するための図である。
【図８】図８は、本実施の形態の概要を説明するための図である。
【図９】図９は、本実施の形態の概要を説明するための図である。
【図１０】図１０は、本実施の形態の概要を説明するための図である。
【図１１】図１１は、本実施の形態の概要を説明するための図である。
【図１２】図１２は、本実施の形態の概要を説明するための図である。
【図１３】図１３は、本実施の形態の概要を説明するための図である。
【図１４】図１４は、本実施の形態に係る情報処理装置の構成例を示す図である。
【図１５】図１５は、学習処理の処理フローを示す図である。
【図１６】図１６は、アンカー抽出処理の処理フローを示す図である。
【図１７】図１７は、初期類似度行列を説明するための図である。
【図１８】図１８は、識別処理の処理フローを示す図である。
【図１９】図１９は、コンピュータの機能ブロック図である。
【発明を実施するための形態】
【００１７】
まず、図６乃至図１３を用いて、本実施の形態の概要を説明しておく。
【００１８】
本実施の形態では、ＡＧＨ法におけるアンカーをｋ－ｍｅａｎｓ法で抽出するのではなく、特徴空間のデータ分布を近似するアンカーを算出する。
【００１９】
まず、アンカーを設定することによってデータ分布がどのように変化するのかを図６を用いて模式的に説明する。
【００２０】
ＡＧＨ法では、アンカーを設定した後、各データサンプル間の距離又は類似度を、アンカーを経由して計測する。アンカーａを経由して計測したときの２点Ａ及びＢ間の距離を｜ＡＢ｜$_{anchor}$と表すと、｜ＡＢ｜$_{anchor}$及び｜ＡＣ｜$_{anchor}$は、元の特徴空間における２点間距離｜Ａａ｜、｜Ｂａ｜及び｜Ｃａ｜によって以下のように表される。
【００２１】
【数１】

$$\left|AB\right|_{anchor} \approx \left\|Aa\right|-\left|Ba\right\|$$
$$\left|AC\right|_{anchor} \approx \left\|Aa\right|-\left|Ca\right\|$$

【００２２】
特徴空間のデータ分布を近似するアンカーを算出するということは、データサンプル間の距離をできるだけ保存するアンカーを求めるということである。具体的には、アンカーを経由した場合の距離の比｜｜Ａａ｜－｜Ｂａ｜｜：｜｜Ａａ｜－｜Ｃａ｜｜を、元の特徴空間における距離の比｜ＡＢ｜：｜ＡＣ｜と可能な限り同じにするようなアンカーを抽出する。
【００２３】
すなわち、データサンプル間の距離を可能な限り保存するアンカーを求めるという問題は、類似度行列によって定式化できる。すなわち、図７に模式的に示すように、アンカー

を設定したときのデータ分布を表す類似度行列Ｚ^$_{anchor}$（”^”は、以下、Ｚの上に置かれたものとする）が、図８に模式的に示すような元の特徴空間におけるデータ分布を表す類似度行列Ｚ^に可能な限り一致するようにアンカーａを設定する。

【００２４】

類似度行列とは、データサンプルがｍ個与えられた場合はｍ×ｍの行列であり、行列の（ｉ，ｊ）成分はデータサンプルｉの特徴ベクトルｓ$_{i}$とデータサンプルｊの特徴ベクトルｓ$_{j}$の類似度を表すものである。

【００２５】

アンカーを設定したときのデータ分布を表す類似度行列Ｚ^$_{anchor}$は、アンカーをｌ個として、それらの特徴ベクトルをａ$_{1}$、ａ$_{2}$，．．．，ａ$_{l}$と表すと、ａ$_{1}$，ａ$_{2}$，．．．，ａ$_{l}$の関数となるが、類似度行列の定義からして非線形の極めて複雑な関数となる。具体的には、ＡＧＨ法における類似度の算出方法は足切りなどの演算が存在するので、ａ$_{1}$，ａ$_{2}$，．．．，ａ$_{l}$を陽に変数とした関数として表すことができず、以下のような方程式を立てて解くことは困難である。

【００２６】

【数２】

$$\hat{\mathbf{Z}} = \hat{\mathbf{Z}}_{anchor}\left(\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_l\right)$$

【００２７】

そこで、類似度行列Ｚ^を固有値分解して、Ｚ^を近似するアンカーを算出する。Ｚ^を固有値分解すると、Ｚ^は射影行列の和に近似でき、「どのデータサンプルをアンカーとして用いれば類似度行列に近い行列を得られるか」という問題が、「どのデータサンプルをアンカーとして用いれば射影行列に近い行列を得られるか」という問題に分解できる。そして、「どのデータサンプルをアンカーとして用いれば射影行列に近い行列を得られるか」という問題については、近似的な解を得ることができる。

【００２８】

すなわち、データサンプルｓ$_{j}$の射影行列σ$_{i}$ｖ$_{i}$ｖ$_{i}^{T}$に対する寄与度が固有ベクトルｖ$_{i}$の成分の絶対値で測れるので、固有ベクトルの成分の絶対値が大きいデータサンプルをアンカーとして選択する。

【００２９】

具体的には、本実施の形態におけるアンカーには、図９に模式的に示すような特徴空間においては、データサンプルｂのようなデータサンプルが選択される。そうすると、低次元特徴空間においては、図１０に模式的に示すように、１箇所に集中することなく、分散するようになる。このような場合、図１１に模式的に示すように特徴空間において未知のデータＸが入力された場合、図１２に模式的に示すように低次元特徴空間に写像されるわけであるが、分布に応じた低次元特徴空間となっているので、アンカーｂからの距離が近い他のデータサンプルが多数となることはない。従って、図１３に模式的に示すように、未知のデータＸに近いデータサンプルＺが正しく見つかるようになる。

【００３０】

次に、このような前提の下処理を行う、本実施の形態における情報処理装置の構成例を図１４に示す。

【００３１】

本実施の形態における情報処理装置１００は、第１データ格納部１０１と、アンカー抽出部１０２と、アンカー格納部１０３と、ハッシュベクトル算出部１０４と、登録部１０５と、データベース１０６と、データ入力部１０７と、第２データ格納部１０８と、照合処理部１０９と、データ出力部１１０とを有する。

【００３２】

第１データ格納部１０１は、例えばｍ個のデータサンプルの特徴ベクトル及びカテゴリ名が格納されている。

【００３３】
アンカー抽出部１０２は、類似度行列算出部１０２１と、固有値分解処理部１０２２と、アンカー特定部１０２３とを有する。
【００３４】
類似度行列算出部１０２１は、第１データ格納部１０１に格納されているデータサンプルの特徴ベクトルから類似度行列を算出する。
【００３５】
固有値分解処理部１０２２は、類似度行列に対して固有値分解処理を実行し、固有ベクトルを生成する。
【００３６】
アンカー特定部１０２３は、固有ベクトルを用いて、第１データ格納部１０１に格納されているデータサンプルのうちアンカーとして採用するデータサンプルを選定し、アンカー格納部１０３に格納する。
【００３７】
ハッシュベクトル算出部１０４は、第１データ格納部１０１に格納されているデータサンプルの特徴ベクトルから、アンカー格納部１０３に格納されているアンカーに基づきハッシュベクトルを算出し、登録部１０５に出力する。ハッシュベクトル算出部１０４によるハッシュベクトル算出処理については、従来と同じなので詳細には述べない。
【００３８】
登録部１０５は、ハッシュベクトルと、当該ハッシュベクトルの元となったデータサンプルのカテゴリ名とを、データベース１０６に登録する。
【００３９】
データ入力部１０７は、ユーザ又は他のコンピュータから、未知のデータの特徴ベクトルの入力を受け付け、第２データ格納部１０８に格納する。
【００４０】
照合処理部１０９は、ハッシュベクトル算出部１０４に、第２データ格納部１０８に格納されているデータの特徴ベクトルについてのハッシュベクトルを算出させる。そして、照合処理部１０９は、当該ハッシュベクトルと、データベース１０６に格納されている各ハッシュベクトルとのハミング距離を算出して、最短のハミング距離が算出されたハッシュベクトルに対応付けて格納されているカテゴリ名を特定し、データ出力部１１０に出力する。
【００４１】
データ出力部１１０は、出力装置（表示装置、印刷装置や他のコンピュータなど）に、特定されたカテゴリ名を出力する。
【００４２】
次に、図１５乃至図１８を用いて、情報処理装置１００の処理内容について説明する。
【００４３】
本実施の形態では、ＡＧＨ法と同様に、予め実施しておく学習処理と、学習処理の結果に基づき未知のデータに対してカテゴリ名を識別する識別処理とが行われる。
【００４４】
まず、学習処理について、図１５乃至図１７を用いて説明する。
【００４５】
アンカー抽出部１０２は、本実施の形態における主要部であるアンカー抽出処理を実行する（図１５：ステップＳ１）。アンカー抽出処理については、図１６を用いて説明する。
【００４６】
まず、類似度行列算出部１０２１は、第１データ格納部１０１に格納されているデータサンプルの特徴ベクトルから、類似度行列を算出する（図１６：ステップＳ１１）。
【００４７】
（Ａ）初期類似度行列Ｚの算出

類似度行列算出部１０２１は、各データサンプルの特徴ベクトルｓ$_i$に対して、自データサンプル以外の他のデータサンプルの特徴ベクトルｓ$_j$とのユークリッド距離に基づく類似度ｈ（ｓ$_i$，ｓ$_j$）を算出する。具体的には、以下のような式に従って類似度を算出する。

【００４８】

【数３】

$$h\left(\mathbf{s}_i,\mathbf{s}_j\right)=\exp\left(-\frac{D^2\left(\mathbf{s}_i,\mathbf{s}_j\right)}{T}\right)$$

【００４９】

なお、Ｄ（Ａ，Ｂ）は、ＡとＢとのユークリッド距離を表しており、Ｄ^2（Ａ，Ｂ）は、ユークリッド距離の二乗を表す。また、Ｔは、比例定数で、予め設定された値である。

【００５０】

このような計算をデータサンプルｉ及びｊのペアの各々について算出すれば、図１７に示すように、初期類似度行列Ｚのｉｊ成分の値が算出される。

【００５１】

（Ｂ）足切り処理

類似度ｈ（ｓ$_i$，ｓ$_j$）については、以下のような足切りを行う。

・｛ｈ（ｓ$_i$，ｓ$_j$）｜ｉ：fixed，１≦ｊ≦ｍ，ｊ≠ｉ｝において、上位ｒ個以外はｈ（ｓ$_i$，ｓ$_j$）＝０

【００５２】

すなわち、初期類似度行列Ｚの行毎に、類似度が大きい順にｒ＋１番目以降は類似度を０に設定する。図１７の例では、点線矩形で囲まれたｉ行目において、類似度を大きい順にソートして、ｒ＋１番目以降の類似度を０に設定する。

【００５３】

（Ｃ）正規化処理

足切り処理後の初期類似度行列Ｚの行毎に、成分の総和が１となるように正規化する。図１７の例では、点線矩形で囲まれたｉ行目において、類似度の総和が１となるように正規化する。

【００５４】

（Ｄ）対称行列生成

正規化処理後の初期類似度行列Ｚから、類似度行列Ｚ^を算出する。これによって、ｍ×ｍの対称行列となった類似度行列Ｚ^が得られる。

【００５５】

【数４】

$$\hat{\mathbf{Z}}=\mathbf{Z}\mathbf{Z}^T$$

【００５６】

次に、固有値分解処理部１０２２は、類似度行列Ｚ^に対して固有値分解処理を実行する（ステップＳ１３）。

【００５７】

本実施の形態では、固有値のうち大きい順に上位ｋ個の固有値σ$_i$を採用して、類似度行列Ｚ^を以下のように近似する。

【００５８】

【数５】

$$\hat{\mathbf{Z}}\approx\mathbf{U}_k\mathbf{\Sigma}_k\mathbf{U}_k{}^T=\sigma_1\mathbf{v}_1\mathbf{v}_1^T+\sigma_2\mathbf{v}_2\mathbf{v}_2^T\cdots+\sigma_k\mathbf{v}_k\mathbf{v}_k^T$$

【００５９】
【数６】

$$\mathbf{\Sigma}_k = diag(\sigma_t) = \begin{pmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \sigma_k \end{pmatrix}$$

【００６０】
【数７】

$$\mathbf{U}_k = (\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_k)$$

なお、ｖ$_i$は、固有値σ$_i$に対応する固有ベクトルである。また、σ$_i$ｖ$_i$ｖ$_i^T$は、射影行列と呼ばれる。

【００６１】
その後、アンカー特定部１０２３は、各固有ベクトルｖ$_t$から、アンカーとして用いるべきデータサンプルを特定し、当該データサンプルの特徴ベクトルをアンカー格納部１０３に格納する（ステップＳ１５）。そして呼び出し元の処理に戻る。

【００６２】
具体的には、固有ベクトルｖ$_t$（１≦ｔ≦ｋ）の各々について、当該固有ベクトルｖ$_t$において絶対値が最も大きい成分を特定する。そして、ｋ＞ｌであれば、ｋ個の固有ベクトルのうち、特定された成分の絶対値が大きい順にｌ個の固有ベクトルを選択して、選択された固有ベクトルについて特定された成分の順番ｉ$_u$のデータサンプルｓ$_{iu}$を、アンカーａ$_{iu}$として採用する。

【００６３】
すなわち、固有ベクトルｖ$_t$を以下のように表す。

【００６４】
【数８】

$$\mathbf{v}_t = (v_1, v_2, ..., v_m)^T$$

【００６５】
そして、固有ベクトルｖ$_t$において絶対値が最も大きい成分を特定するということは、以下のように表される。

【００６６】
【数９】

$$i_u = \underset{i \in \{1,2,...,m\} \setminus \{i_1,...,i_{u-1}\}}{\arg\max} \left(|v_i| \right)$$

なお、既に選択されたｉ$_1$乃至ｉ$_{u-1}$と同じｉ$_u$については選択できないので、次に絶対値が大きい成分の番号を特定する。

【００６７】
このような処理を実行すれば、ｍ個のデータサンプル間についての類似度行列Ｚ^に、ｍ個のデータサンプルから選択されたアンカーとの類似度に基づくｍ個のデータサンプル間の類似度行列Ｚ^$_{anchor}$が近似するように、アンカーが選択される。

【００６８】
すなわち、ＡＧＨ法における識別処理の高速性を保持しつつ、識別精度を向上させるこ

とができるようになる。

【００６９】

すなわち、本実施の形態では、高速且つ高精度な最近傍探索を行うことができる。特に、高次元空間内においてデータ分布がいかなる状態であっても高精度な最近傍探索を行うことができる。

【００７０】

図１５の処理の説明に戻って、ハッシュベクトル算出部１０４は、第１データ格納部１０１に格納されている各データサンプルの特徴ベクトルから、アンカー格納部１０３に格納されているアンカーのデータに基づき、ハッシュベクトルを算出し、登録部１０５に出力する（ステップＳ３）。ハッシュベクトルは、ｎ次元のベクトルであって、個々の成分の値は０又は１である。この処理は、従来と同じであるから、詳細な説明については省略する。

【００７１】

そして、登録部１０５は、ハッシュベクトル算出部１０４によって算出されたハッシュベクトルと、当該ハッシュベクトルの元となったデータサンプルのカテゴリ名とを、対応付けてデータベース１０６に登録する（ステップＳ５）。

【００７２】

このようにすれば、学習処理が完了して、識別処理を行う準備ができたことになる。

【００７３】

次に、図１８を用いて、識別処理の処理フローを説明する。

【００７４】

まず、データ入力部１０７は、ユーザ又は他のコンピュータから、未知のデータの入力を受け付け、第２データ格納部１０８に格納する（図１８：ステップＳ２１）

【００７５】

そうすると、照合処理部１０９は、第２データ格納部１０８に格納されている未知のデータの特徴ベクトルに対するハッシュベクトルを、ハッシュベクトル算出部１０４に算出させる（ステップＳ２３）。この処理も、従来と同じであり、演算内容としてはステップＳ３と同様であり、詳細な説明は省略する。

【００７６】

そして、照合処理部１０９は、未知のデータに対するハッシュベクトルと、データベース１０６に格納されている各ハッシュベクトルとのハミング距離を算出し、最もハミング距離が短いハッシュベクトルに対応付けられているカテゴリ名を特定する照合処理を実行し、処理結果をデータ出力部１１０に出力する（ステップＳ２５）。

【００７７】

データ出力部１１０は、照合処理部１０９から処理結果であるカテゴリ名を受け取ると、ユーザ又は他のコンピュータに対してカテゴリ名を出力する（ステップＳ２７）。

【００７８】

以上のような処理を行うことで、アンカーが適切に設定されているので、高精度なデータ識別を行うことができるようになる。

【００７９】

以上本発明の実施の形態を説明したが、本発明はこれに限定されるものではない。

【００８０】

例えば、図１４の情報処理装置１００の機能ブロック構成は、プログラムモジュール構成とは一致しない場合がある。また、処理フローについても、処理結果が変わらない限り、順番を入れ替えたり、並列に実行する場合もある。

【００８１】

なお、本実施の形態におけるデータは、画像データであったり、文字データであったり、個人情報であったりする。

【００８２】

さらに、情報処理装置１００は１台のコンピュータではなく、複数台のコンピュータで

機能分担する場合もある。

【００８３】

なお、上で述べた情報処理装置１００は、コンピュータ装置であって、図１９に示すように、メモリ２５０１とＣＰＵ（Central Processing Unit）２５０３とハードディスク・ドライブ（ＨＤＤ：Hard Disk Drive）２５０５と表示装置２５０９に接続される表示制御部２５０７とリムーバブル・ディスク２５１１用のドライブ装置２５１３と入力装置２５１５とネットワークに接続するための通信制御部２５１７とがバス２５１９で接続されている。オペレーティング・システム（ＯＳ：Operating System）及び本実施例における処理を実施するためのアプリケーション・プログラムは、ＨＤＤ２５０５に格納されており、ＣＰＵ２５０３により実行される際にはＨＤＤ２５０５からメモリ２５０１に読み出される。ＣＰＵ２５０３は、アプリケーション・プログラムの処理内容に応じて表示制御部２５０７、通信制御部２５１７、ドライブ装置２５１３を制御して、所定の動作を行わせる。また、処理途中のデータについては、主としてメモリ２５０１に格納されるが、ＨＤＤ２５０５に格納されるようにしてもよい。本技術の実施例では、上で述べた処理を実施するためのアプリケーション・プログラムはコンピュータ読み取り可能なリムーバブル・ディスク２５１１に格納されて頒布され、ドライブ装置２５１３からＨＤＤ２５０５にインストールされる。インターネットなどのネットワーク及び通信制御部２５１７を経由して、ＨＤＤ２５０５にインストールされる場合もある。このようなコンピュータ装置は、上で述べたＣＰＵ２５０３、メモリ２５０１などのハードウエアとＯＳ及びアプリケーション・プログラムなどのプログラムとが有機的に協働することにより、上で述べたような各種機能を実現する。

【００８４】

以上述べた本実施の形態をまとめると、以下のようになる。

【００８５】

本実施の形態に係る情報処理装置は、（Ａ）複数のデータサンプルを格納するデータ格納部と、（Ｂ）データ格納部に格納された複数のデータサンプル間についての類似度行列に、複数のデータサンプルから選択された特定のデータサンプルとの類似度に基づく複数のデータサンプル間の類似度行列が近似するように、特定のデータサンプルを抽出する抽出部とを有する。

【００８６】

このような特定のデータサンプルを抽出してＡＧＨ法におけるアンカーとして用いれば、識別精度を向上させることができるようになる。

【００８７】

なお、上で述べた抽出部が、（ｂ１）複数のデータサンプル間についての類似度行列を算出する類似度行列算出部と、（ｂ２）算出された類似度行列に対して固有値分解を行って固有ベクトルを生成する固有値分解処理部と、（ｂ３）生成された固有ベクトルの各成分の絶対値に基づき、複数のデータサンプルから上記特定のデータサンプルを特定する特定部とを有する。

【００８８】

例えば、固有ベクトルの各成分の絶対値のうち最も大きい絶対値が得られた成分の順番を用いて、複数のデータサンプルから上記順番における特定のデータサンプルを特定するものである。

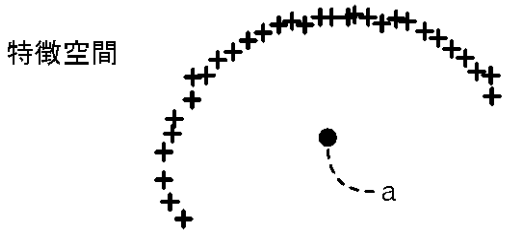
【００８９】

さらに、上で述べた情報処理装置は、（Ｃ）特定のデータサンプルを用いて複数のデータサンプルの各々についてハッシュベクトルを算出する算出部と、（Ｄ）算出されたハッシュベクトルと、データサンプルについてのカテゴリ名とを対応付けてデータベースに登録する登録部と、（Ｅ）あるデータについて特定のデータサンプルを用いて算出されたハッシュベクトルとデータベースに登録されたハッシュベクトルとの距離を算出して、最短距離のハッシュベクトルに対応付けられたカテゴリ名を特定する処理部とをさらに有するようにしても良い。このようにすれば、ＡＧＨ法で最近傍探索が行われる。

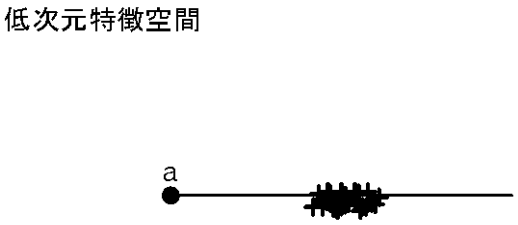
【００９０】
なお、上で述べたような処理をプロセッサ又はコンピュータに実行させるためのプログラムを作成することができ、当該プログラムは、例えばフレキシブル・ディスク、ＣＤ－ＲＯＭなどの光ディスク、光磁気ディスク、半導体メモリ（例えばＲＯＭ）、ハードディスク等のコンピュータ読み取り可能な記憶媒体又は記憶装置に格納される。なお、処理途中のデータについては、ＲＡＭ等の記憶装置に一時保管される。
【００９１】
以上の実施例を含む実施形態に関し、さらに以下の付記を開示する。
【００９２】
（付記１）
複数のデータサンプルを格納するデータ格納部と、
前記データ格納部に格納された前記複数のデータサンプル間についての類似度行列に、前記複数のデータサンプルから選択された特定のデータサンプルとの類似度に基づく前記複数のデータサンプル間の類似度行列が近似するように、前記特定のデータサンプルを抽出する抽出部と、
を有する情報処理装置。
【００９３】
（付記２）
前記抽出部が、
前記複数のデータサンプル間についての類似度行列を算出する類似度行列算出部と、
算出された前記類似度行列に対して固有値分解を行って固有ベクトルを生成する固有値分解処理部と、
生成された前記固有ベクトルの各成分の絶対値に基づき、前記複数のデータサンプルから前記特定のデータサンプルを特定する特定部と、
を有する付記１記載の情報処理装置。
【００９４】
（付記３）
前記特定のデータサンプルを用いて前記複数のデータサンプルの各々についてハッシュベクトルを算出する算出部と、
算出された前記ハッシュベクトルと、前記データサンプルについてのカテゴリ名とを対応付けてデータベースに登録する登録部と、
あるデータについて前記特定のデータサンプルを用いて算出されたハッシュベクトルと前記データベースに登録されたハッシュベクトルとの距離を算出して、最短距離のハッシュベクトルに対応付けられたカテゴリ名を特定する処理部と、
をさらに有する付記１又は２記載の情報処理装置。
【００９５】
（付記４）
複数のデータサンプルを格納するデータ格納部に格納された前記複数のデータサンプル間についての類似度行列に、前記複数のデータサンプルから選択された特定のデータサンプルとの類似度に基づく前記複数のデータサンプル間の類似度行列が近似するように、前記特定のデータサンプルを抽出する
処理を含み、コンピュータが実行する情報処理方法。
【００９６】
（付記５）
前記抽出する処理が、
前記複数のデータサンプル間についての類似度行列を算出し、
算出された前記類似度行列に対して固有値分解を行って固有ベクトルを生成し、
生成された前記固有ベクトルの各成分の絶対値に基づき、前記複数のデータサンプルから前記特定のデータサンプルを特定する
処理を含む付記４記載の情報処理方法。

【００９７】
（付記６）
複数のデータサンプルを格納するデータ格納部に格納された前記複数のデータサンプル間についての類似度行列に、前記複数のデータサンプルから選択された特定のデータサンプルとの類似度に基づく前記複数のデータサンプル間の類似度行列が近似するように、前記特定のデータサンプルを抽出する
処理を、コンピュータに実行させるためのプログラム。
【００９８】
（付記７）
前記抽出する処理が、
前記複数のデータサンプル間についての類似度行列を算出し、
算出された前記類似度行列に対して固有値分解を行って固有ベクトルを生成し、
生成された前記固有ベクトルの各成分の絶対値に基づき、前記複数のデータサンプルから前記特定のデータサンプルを特定する
処理を含む付記６記載のプログラム。
【符号の説明】
【００９９】
１０１　第１データ格納部
１０２　アンカー抽出部
１０３　アンカー格納部
１０４　ハッシュベクトル算出部
１０５　登録部
１０６　データベース
１０７　データ入力部
１０８　第２データ格納部
１０９　照合処理部
１１０　データ出力部

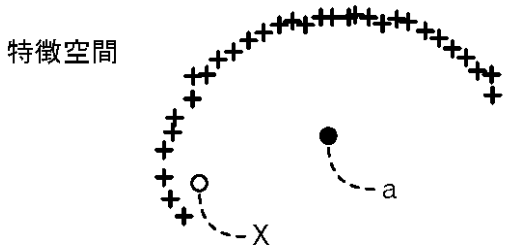
【図１】



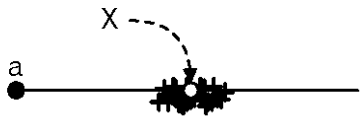
【図２】



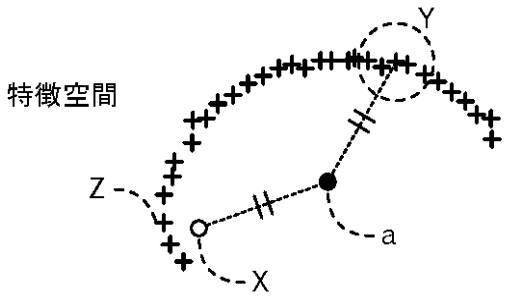
【図３】



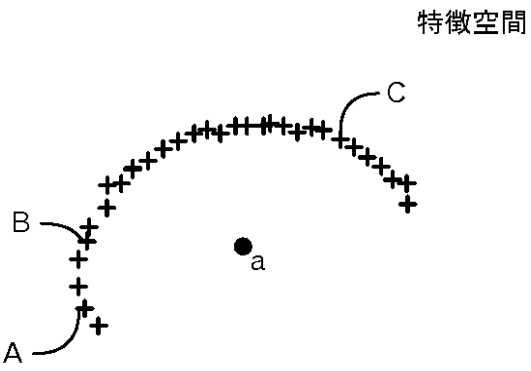
【図４】



【図５】

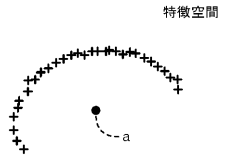


【図６】

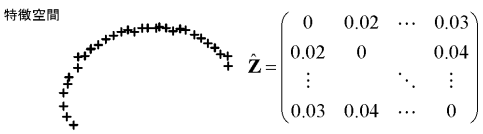


【図７】

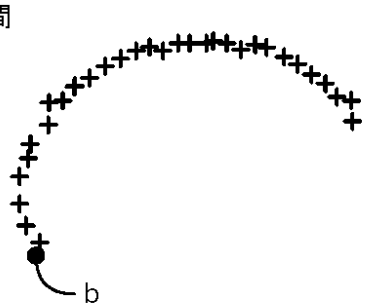
$$\hat{\mathbf{Z}}_{anchor} = \begin{pmatrix} 0 & 0.05 & \cdots & 0.04 \\ 0.05 & 0 & & 0.035 \\ \vdots & & \ddots & \vdots \\ 0.04 & 0.035 & \cdots & 0 \end{pmatrix}$$



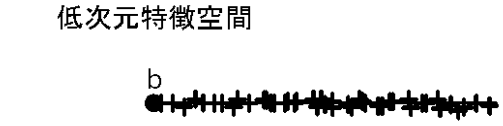
【図８】



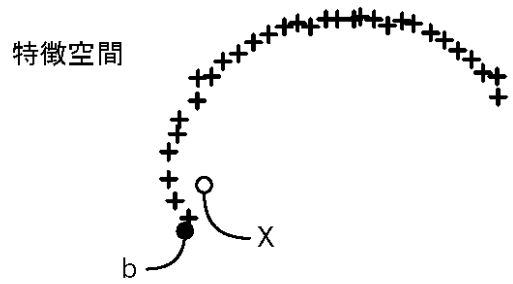
【図９】



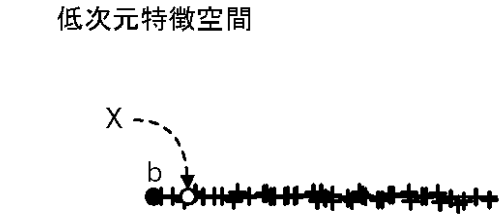
【図１０】



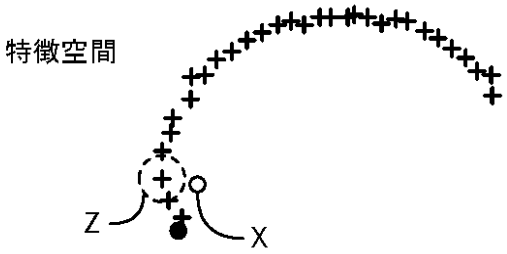
【図１１】



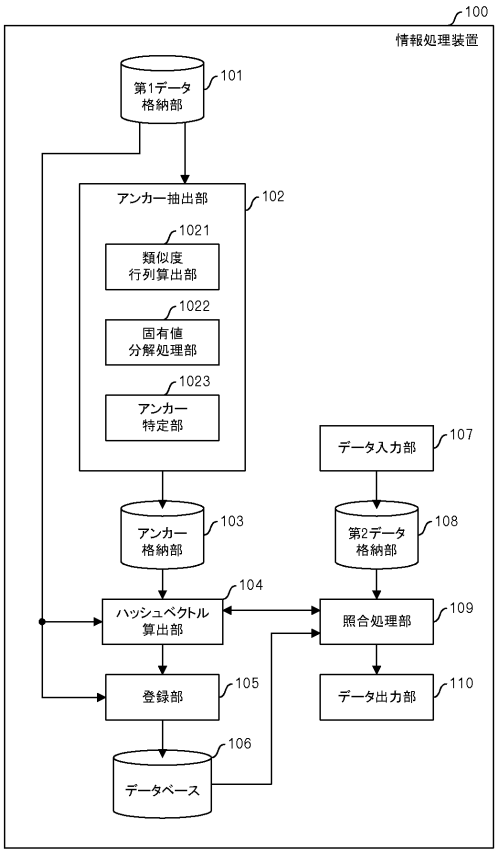
【図１２】



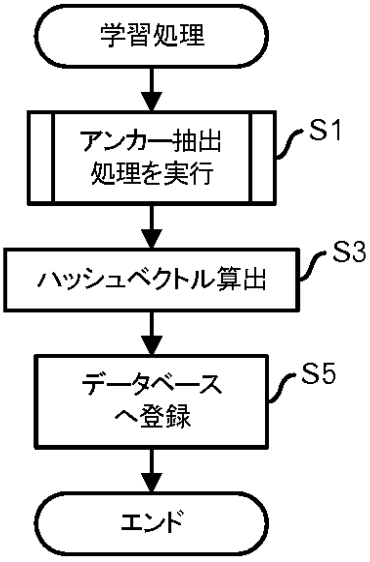
【図１３】



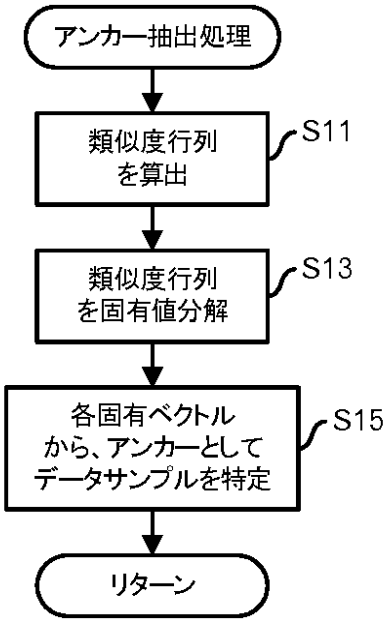
【図１４】



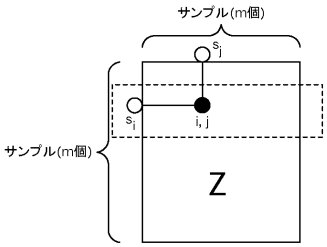
【図１５】



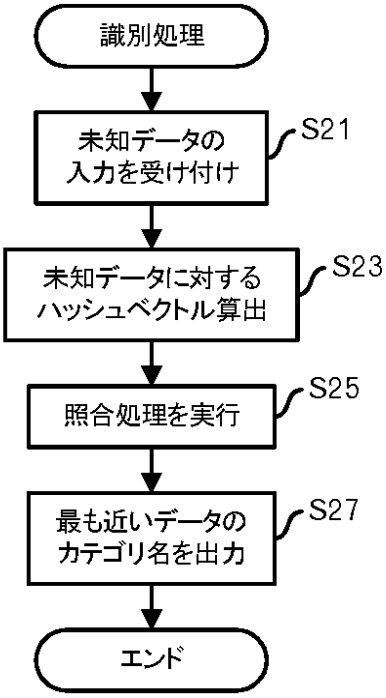
【図１６】



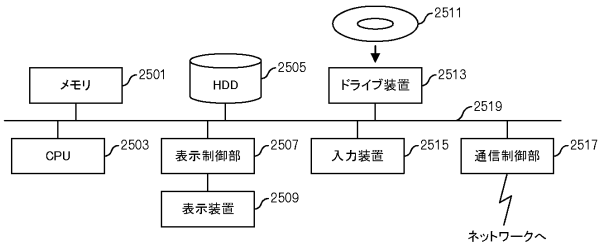
【図１７】



【図１８】



【図１９】



フロントページの続き

(56)参考文献 特開２０１２－０３８２７２（ＪＰ，Ａ）
国際公開第２０１２／０５０９５２（WO，Ａ1）
米国特許出願公開第２０１３／００８６５５３（ＵＳ，Ａ1）
特開２０１０－２５０３７７（ＪＰ，Ａ）
特開２０１２－０８８９７２（ＪＰ，Ａ）
特開２０１３－０２０２９０（ＪＰ，Ａ）
田中　竜仁、外4名，スケッチを入力とした3次元モデル検索の高速化，Ｍｅｄｉａ　Ｃｏｍｐｕｔｉｎｇ　Ｃｏｎｆｅｒｅｎｃｅ　２０１２　２０１２年度　画像電子学会第４０回年次大会　予稿集　Ｖｉｓｕａｌ　Ｃｏｍｐｕｔｉｎｇ／グラフィクスとCAD合同シンポ　［DVD－ROM］　Ｍｅｄｉａ　Ｃｏｍｐｕｔｉｎｇ　Ｃｏｎｆｅｒｅｎｃｅ，日本，一般社団法人画像電子学会，２０１２年　7月　2日，ｐ．１－４
鈴木　郁美、外3名，「ハブの出現しやすさ」から見たラプラシアンベースカーネル，電子情報通信学会技術研究報告，日本，社団法人電子情報通信学会，２０１１年１１月　2日，第１１１巻，第２７５号，ｐ．２５７－２６２
三川　健太、外3名，テキスト分類問題におけるカテゴリ情報を用いた適応的距離学習に関する一考察，電子情報通信学会技術研究報告，日本，一般社団法人電子情報通信学会，２０１２年１０月３１日，第１１２巻，第２７９号，ｐ．８３－８８

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１７／３０