自然言語処理　150－5
(2002. 7. 15)

1次元自己組織化マップを用いた高次元データの高速近傍検索

北　研二†　獅々堀正幹††

† 徳島大学高度情報化基盤センター　〒770-8506 徳島市南常三島町 2-1
†† 徳島大学工学部　〒770-8506 徳島市南常三島町 2-1
E-mail: †{kita,bori}@is.tokushima-u.ac.jp

あらまし　高次元空間における最近傍検索 (nearest neighbor search) は、マルチメディア・コンテンツ検索、データ・マイニング、パターン認識等の分野における重要な研究課題の1つである。高次元空間では、ある点の最近点と最遠点との間に距離的な差が生じなくなるという現象が起こるため、効率的な多次元インデキシング手法を設計することが極度に困難となる。本稿では、1次元自己組織化マップを用いた近似的最近傍検索の手法を提案し、提案した手法の有効性を類似画像検索と文書検索の2種類の実験により評価する。自己組織化マップを用いて、高次元空間での近傍関係をできる限り保ちつつ、高次元データを1次元空間へ配置し、1次元マップから得られる情報で探索範囲を限定することにより、きわめて高速な最近傍検索が可能となる。
キーワード　最近傍検索，多次元インデキシング，自己組織化マップ

Efficient Multidimensional Indexing Using One-dimensional Self-Organizing Maps

Kenji KITA† and Masami SHISHIBORI††

† Center for Advanced Information Technology, Tokushima University　Tokushima 770-8506, Japan
†† Faculty of Engineering, Tokushima University　Tokushima 770-8506, Japan
E-mail: †{kita,bori}@is.tokushima-u.ac.jp

Abstract　Nearest neighbor search in high dimensional spaces is an interesting and important problem which is relevant for a wide variety of applications, including multimedia information retrieval, data mining, and pattern recognition. For such applications, the curse of high dimensionality tends to be a major obstacle in the development of efficient indexing methods. This paper addresses the problem of designing an efficient multidimensional indexing structure for high dimensional nearest neighbor search. More specifically, using self-organizing maps (SOM), high-dimensional vector data are first transformed into one-dimensional units while preserving the higher order topology by mapping similar data items to the same or the neighboring unit. Then, given a query vector, only data items whose location is close to the unit location of the query are considered as candidates. Experimental results indicate that our scheme scales well even for a very large number of dimensions.
Key words　Nearest neighbor search, Multidimensional indexing, Self-organizing map

1. はじめに

計算機の高性能化や記憶容量の大容量化および低価格化にともない、情報のマルチメディア化が急速に進行しており、このような背景のもと、マルチメディア・コンテンツに対する情報検索技術の必要性がますます大きくなってきている。マルチメディア・コンテンツ検索では、マルチメディア情報そのものから得られる特徴量に基づき類似検索を行なうという内容型検索 (content-based retrieval) が近年の主流であるが、多くの場合、複数の特徴量を多次元ベクトルで表現し、ベクトル間の距離によりコンテンツ間の類似性を判定している。たとえば、文書検索の場合には、索引語の重みベクトルで文書や検索質問を表現することができるし[21], [22]、画像の類似検索の場合には、カラーヒストグラム、テクスチャ特徴量、形状特徴量などから成る特徴量ベクトルにより画像コンテンツを表現する[10], [19]。

特徴量ベクトルに基づくコンテンツの類似検索は、検索質問として与えられたベクトルと距離的に近いコンテンツ・データベース中のベクトルを見つけるという最近傍検索 (nearest

neighbor search) の問題に帰着することができる。データベース中のベクトルと逐次的に比較する線形探索では、データベースの規模に比例した計算量が必要となるため、データベースが大規模化した際の検索システムの処理効率に深刻な影響を及ぼすことになる。したがって、最近傍検索を効率的に行なうための多次元インデキシング技術の開発が重要な課題として、従来より活発に研究されてきた[1],[11]。

ユークリッド空間における多次元インデキシング手法には、R-tree[13], SS-tree[24], SR-tree[14] などが提案されており、また、より一般の距離空間を対象にしたインデキシング手法としては、VP-tree[25], MVP-tree[8], M-tree[9] などが提案されている。これらのインデキシング手法は、多次元空間を階層的に分割することにより、探索範囲を限定することを基本としている。しかし、高次元空間では、ある点の最近点と最遠点との間に距離的な差が生じなくなるという現象が起こるため[5],[7]、探索する領域を限定することができず、線形探索に近い計算量が必要になってしまうという問題点がある。

高次元空間における上記の問題点に対処するために、近似的な最近傍検索についても研究が進められている。たとえば、ハッシュ法に基づく近似検索手法[12]や空間充填曲線 (space-filling curve) を用いて高次元空間の点を索引付けする手法[17],[23] などが提案されている。

我々は、現在、テキストと画像のクロスメディア情報検索に関する研究の一環として、類似画像検索システムを開発しているが[3],[4]、クロスメディア情報検索では、ユーザとのインタラクションを通じて所望の検索結果を得ることが多々あるため、特徴量ベクトルに基づく最近傍検索の実行回数が必然的に多くなってしまう。このような場合、完全な最近傍検索は必要ではなく、むしろ高速な近似的最近傍検索のほうが望ましい。

本稿では、1 次元自己組織化マップを用いた、高速な近似的最近傍検索の手法を提案し、提案した手法の有効性を類似画像検索と文書検索という 2 種類の実験により評価する。最近傍検索を行なう際の一番のボトルネックは、2 次記憶上のデータへのアクセスであるが、提案する手法は、次元数がきわめて多い場合でも効率的にディスク・アクセスを行なうことができるという利点を持っている。

2. 自己組織化マップを用いた最近傍検索

2.1 自己組織化マップ

自己組織化マップ (self-organizing map; SOM)[15] は、教師なし競合学習により、高次元データを低次元データに写像する 2 階層型のニューラルネットワークである。自己組織化マップでは、高次元空間での近傍関係をできる限り保ちつつ、低次元空間へデータを配置するという位相的整列性と呼ばれる特徴を持っている。自己組織化マップの典型的な適用例は、多次元データの可視化であり、この場合には高次元データを 2 次元平面上に配置するということを行なう[16],[18]。

図 1 は、n 次元の入力データを 2 次元平面上に配置する自己組織化ネットワークの例を示している。ネットワークの入力層は、2 次元平面上に格子状に配置されたすべてのユニットと結合されており、各ユニットには、入力層に入力されるデータと同じ次元数の参照ベクトル (reference vector) が対応している。学習の過程では、入力層に入力されたベクトルと最も近い参照ベクトルを持つユニットを探し、このユニットとその近傍にあるユニットの参照ベクトルを入力ベクトルに近づけるという操作を繰り返す。このようにして、同じような位相的特徴を持ったユニットが近傍領域に集まり、結果的に入力データの位相的特徴を反映した自己組織化マップが作られることになる。

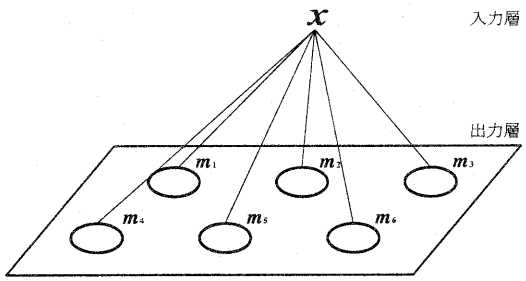


図 1 自己組織化マップ

自己組織化マップの学習アルゴリズムをまとめると、以下のようになる。

(1) 参照ベクトル $\mathbf{m}_i$ をランダムな値で初期化する。

(2) 入力ベクトル $\mathbf{x}$ に最も近い参照ベクトル $\mathbf{m}_c$ を持つユニット c を見つける。

$$\mathbf{m}_c = \underset{\mathbf{m}_i}{\operatorname{argmin}} \|\mathbf{x} - \mathbf{m}_i\| \tag{1}$$

(3) ユニット c および c の近傍領域の参照ベクトル $\mathbf{m}_i$ を次式により更新する。

$$\mathbf{m}_i = \mathbf{m}_i + h_{ci}(\mathbf{x} - \mathbf{m}_i) \tag{2}$$

ここで、h_{ci} はユニット c から離れるにつれ、小さな値になるように設定する。また、h_{ci} は学習が進むにつれ、単調に減少するようにする。

(4) ステップ 2 より繰り返す。

2.2 自己組織化マップを用いた最近傍検索手法

上で述べたように、自己組織化マップでは、高次元空間での近傍関係をできる限り保ちつつ、入力データを低次元空間へ配置することができるという特徴を持っている。この特徴を用いると、高次元空間での最近傍検索を低次元空間での最近傍検索問題に置き換えることができると考えられる。しかし、自己組織化マップの学習には誤差がともなううえ、低次元のマップ上では、高次元空間での距離が保存されていないため、低次元マップだけを用いて最近傍検索を行なうことは不可能である。

我々は、自己組織化マップにより得られた低次元空間での近傍関係から、最近傍検索の探索範囲を限定し、限定されたデータに関してだけ、元の高次元空間上で距離を計算するという方法を考えた。また、探索範囲の限定を効率的に行なうことができるように、1 次元の自己組織化マップを用いることにした。以下に、1 次元自己組織化マップを用いた最近傍検索手法をま

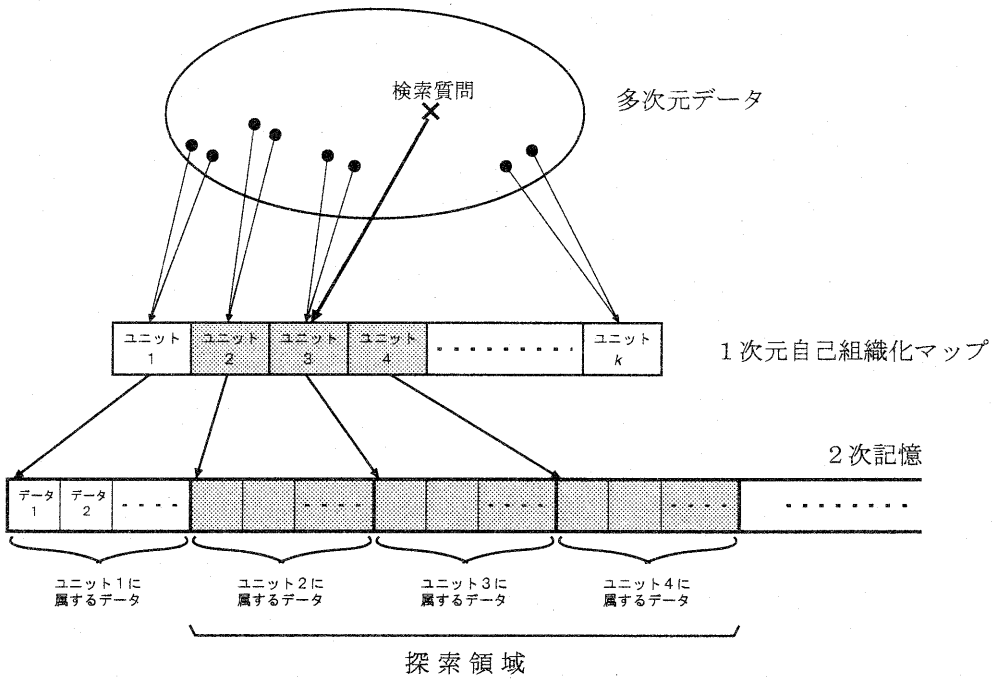


図 2　自己組織化マップを用いた多次元インデキシング

とめる (図 2 参照)。

多次元インデキシングの作成

（1） 自己組織化マップの学習アルゴリズムにより、多次元データを 1 次元上に配置する。ユニット数を k とすると、データは k 個のクラスタに分割されることになる。

（2） 各クラスタに属するデータを、2 次記憶上の連続した領域に格納する。また、この際、1 次元マップ上の各ユニットに 2 次記憶領域へのポインタを持たせる。なお、2 次記憶領域には、元の多次元データを格納する。

最近傍検索

（1） 与えられた検索質問ベクトルに最も近い参照ベクトルを持つユニット c を見つける。

（2） ユニット c の近傍ユニットに配置されたデータに対してのみ、検索質問との距離計算を行なう。距離計算の際には、2 次記憶上に格納されている多次元データを用いる。

（3） 上記で計算された結果を、距離の小さい順にソートし、これを最近傍検索の結果として出力する。

検索質問ベクトルと距離計算の行なわれるデータは、2 次記憶上の連続した領域に格納しているため、2 次記憶へのアクセスはきわめて効率的に行なうことが可能である。3 節で実験結果を述べるが、1 次元マップ上の各ユニットに割り当てられるデータ数が大きく偏ることはなく、概ね平均化している。したがって、2 次記憶へのアクセス回数は数回程度である。

3. 実験結果

自己組織化マップを用いた最近傍検索手法の有効性を調べるために、類似画像検索実験と文書検索実験を行なった。以下で実験の概要および実験結果について述べる。

3.1 類似画像検索実験

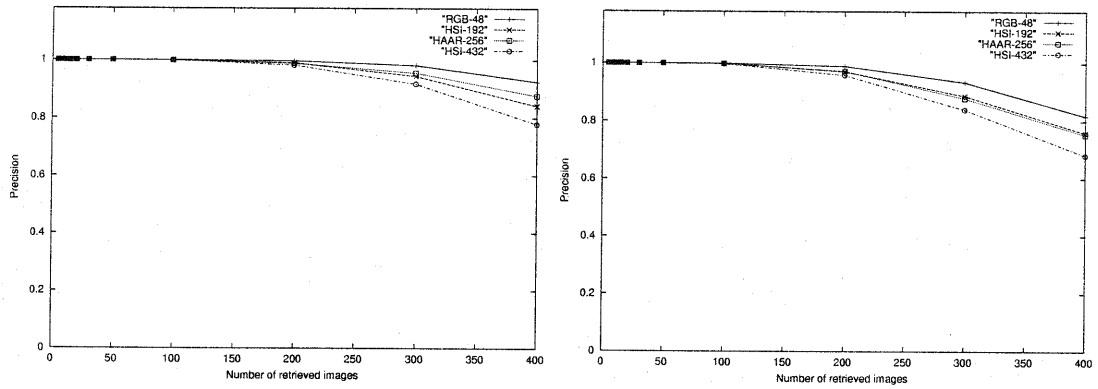
類似画像検索実験では、Corel データベースから抽出した 42,381 件のカラー写真画像を用いた。また、このうち、424 件 (全体の 1%) の画像データをランダムに抽出し、検索画像とした。これらの画像データから、表 1 に示すような、次元数の異なる 4 種類の特徴量ベクトルを作成した。

自己組織化マップを用いた最近傍検索の精度を調べるためには、検索された画像のうち、どれが正解であるかという情報が必要である。このため、各検索画像と全画像データとの間のユークリッド距離を線形探索により求め、距離の小さい 400 件を正解データとした。与えられた検索画像から、自己組織化マップを用いた最近傍検索により、上位 400 件の検索結果を出力し、これを正解データと比較することにより適合率を算出した。

自己組織化マップを用いた最近傍検索では、検索条件によって適合率は変化する。適合率が変化する主な要因は、1 次元マップ上の総ユニット数、および、検索の際に用いる近傍数である。ここで、近傍数とは、探索候補の絞り込みの際にいくつのユニットを参照したかを意味しており、具体的には、検索質問の属するユニットに加え、その近傍のユニットをいくつ参照したかを示す。以下では、検索質問の属するユニットのみを参

表 1　画像検索実験に用いた特徴量

特徴量	次元数	特徴量の概略
RGB-48	48	画像全体から 256 階調の R, G, B のヒストグラムを求め、各色 16 次元 (計 48 次元) に圧縮した特徴量
HSI-192	192	画像全体から 256 階調の色相 (hue)、彩度 (saturation)、輝度 (intensity) に関する HSI 特徴量を求め、各特徴量を 64 次元 (計 192 次元) に圧縮した特徴量
HAAR-256	256	画像全体の輝度成分に対して 2 レベルの Haar Wavelet 変換を行い、高域成分の Wavelet 係数を 16 × 16 の各部分画像領域ごと (計 256 次元) に加算平均した特徴量
HSI-432	432	画像全体を 3 × 3 の部分画像に分割し、各部分画像に対して HSI 特徴量を求め、各部分画像の HSI 特徴量を 48 次元 (計 432 次元) に圧縮した特徴量



(a) ユニット数 10, 近傍数 3　　(b) ユニット数 20, 近傍数 3

図 3　画像検索の平均適合率

表 2　画像検索の R 適合率および平均距離計算回数

特徴量	RGB-48			HSI-192			HAAR-256			HSI-432		
ユニット数	5	10	20	5	10	20	5	10	20	5	10	20
近傍数	1	3	3	1	3	3	1	3	3	1	3	3
R 適合率	0.73	0.93	0.82	0.60	0.84	0.76	0.73	0.88	0.75	0.61	0.78	0.68
平均距離計算回数	8940	12226	6211	9320	11998	6209	8884	11190	6121	9005	11528	5989

照したときは近傍数 1、検索質問の属するユニットに加え、その左右両側のユニットを参照したときは近傍数 3 というように表すことにする。なお、近傍数 3 の際、検索質問が 1 次元マップ上の左端 (あるいは右端) のユニットに属している場合には、そのユニットの右側 (あるいは左側) しか参照しない。

図 3 は、ユニット数が 10 あるいは 20、近傍数 3 のときの適合率曲線を示している。横軸方向は検索結果数を、縦軸方向は平均適合率を表しており、グラフは上位 n 件の結果が検索された時点での平均適合率をプロットしたものである。また、表 2 は、さまざまな条件のもとでの平均 R 適合率(注1) と検索質問 1 件当たりの平均距離計算回数を示している。表 2 から分かるように、同じ検索条件のもとでは、適合率は次元数が大きくなるにつれ低下する傾向にあるが、距離計算回数は次元数によらずにほぼ一定である。したがって、本手法は、次元数が増大した場合にも高速性が失われることはない。なお、距離計算回数が次元数によらず一定である理由は、各ユニットに割り当てられるデータ数が概ね平均化しているためである。図 4 に、ユニット数 20 の際に各ユニットに割り当てられたデータ数を示すが、1 次元マップ上の各ユニットに割り当てられるデータ数が極端にばらついていないことを読み取ることができる。

3.2　文書検索実験

3.1 において、類似画像検索を対象にした実験結果を示した。画像特徴量の次元数は、数 10 ～ 数 100 次元程度であるが、これよりも次元数が大きい場合の手法の有効性を調べるために、ベクトル空間モデル (vector space model; VSM) に基づく文書検索を対象とした実験を行なった。

(注1)：R 適合率 (R-precision) とは、検索質問に適合する結果の総数を R とするとき、上位から R 番目までの検索結果を出力した時点での適合率を意味する [2]。R 適合率は、上位に順位付けされた検索結果の有効性を示す評価尺度である。

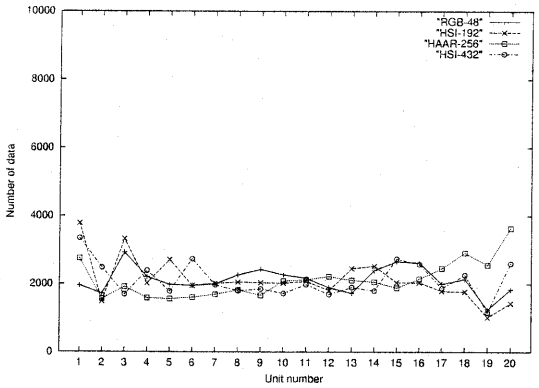


図 4　各ユニット中のデータ数

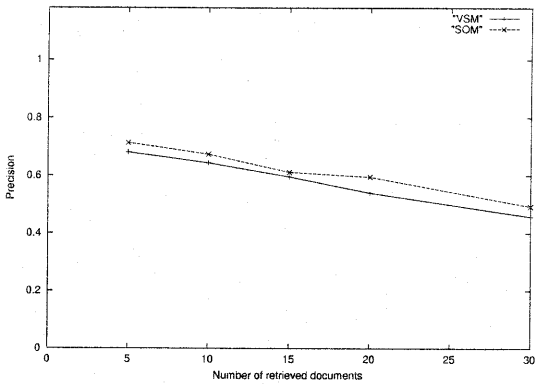


図 5　文書検索の平均適合率

ベクトル空間モデルでは、文書中から索引語を抽出し、文書を索引語の出現頻度に基づくベクトルで表現する [2], [21]。文書ベクトルの次元数は、文書集合全体にわたる索引語の総数と等しいため、次元数はきわめて大きくなる。

本実験では、情報検索評価用のテストコレクションである MEDLINE を用いた。MEDLINE は、検索対象文書 1,033 文書、検索質問 30 文書から成る小規模なコレクションであり、各検索質問には、どの文書が適合しているかという適合情報が用意されている。なお、各検索質問に対する平均適合文書数は 23.2 文書である。

まず前処理として、MEDLINE コレクションから “a” や “about” などの不要語 439 単語、および全文書中に 1 回しか出現しなかった単語を削除した。その後、ポーター・アルゴリズム (Porter algorithm) [20] によるステミングを行なった結果、4,329 個の索引語が得られた。以上の処理により得られた索引語から 4,329 次元の文書ベクトルを構成した。この際、索引語の重み付けとして、局所的重み付けには対数化索引語頻度を、大域的重み付けにはエントロピーを、文書正規化にはコサイン正規化を用いた [2]。

文書検索の評価では、通常のベクトル空間モデルに基づく最近傍検索 (線形探索) と自己組織化マップを用いた最近傍検索の両者とも 30 件の検索結果を出力し、出力結果を MEDLINE の適合情報と比較することにより適合率を求めた。この際、自己組織化マップを用いた最近傍検索では、ユニット数 20、近傍数 3 の条件で検索を行なった。図 5 に、文書検索の適合率曲線を示すが、自己組織化マップを用いた検索のほうがわずかながら良い結果を与えている。なお、ベクトル空間モデルに基づく検索の R 適合率は 0.53 であり、自己組織化マップを用いた検索の R 適合率は 0.58 であった。また、自己組織化マップを用いた最近傍検索の平均距離計算回数は 1 検索質問当たり 141 回であり、これは線形探索の約 1/7 に相当する。

以上は MEDLINE コレクションの適合情報に対する評価であるが、次に、自己組織化マップによる最近傍検索の近似誤差について述べる。ベクトル空間モデルの検索結果を正解とみなした場合、自己組織化マップを用いた検索結果の R 適合率は 0.68 であった。したがって、上位 30 件までの検索では 32% の近似誤差が生じていることになる。しかし、近似誤差があるにもかかわらず、MEDLINE の適合情報に対する評価では、通常のベクトル空間モデルよりも適合率が高くなっている。潜在的意味インデキシング (latent semantic indexing; LSI) [6] などによる検索では、次元数を削減すると検索精度が逆に向上することなどから、高次元空間そのものにおける検索が質的に良い検索結果を与えるとは限らない。我々の提案した手法の近似の程度と検索精度の関係等を調査することは、今後の課題となりそうである。

4. おわりに

本稿では、1 次元自己組織化マップを用いた高次元データの近似的な最近傍検索手法を提案した。提案した手法では、自己組織化マップを用いて、高次元空間での近傍関係をできる限り保ちつつ、高次元データを 1 次元マップ上に配置することにより、最近傍検索の探索範囲を大きく削減することができる。また、本手法では、実際に距離計算の行なわれるデータは、2 次記憶上の連続した領域に格納できるため、2 次記憶へのアクセスを効率的に行なうことができるという大きな利点を持っている。このため、大規模なデータ集合に対しても、きわめて高速な最近傍検索を行なうことが可能である。

従来の SR-tree 等の正確な最近傍検索では、高次元の場合に線形探索に近い計算量が必要となってしまうという問題点があるため、現実的、応用的な場面においては、本手法のような高速な近似的最近傍検索のほうが望ましいと考えられる。

謝辞：本研究の実験の一部に協力頂いた修士課程 1 年の原一眞君に感謝する。また、本研究の一部は、財団法人 放送文化基金の援助によった。

文　献

[1] 片山 紀生, 佐藤 真一 : “類似検索のための索引技術”, 情報処理, Vol. 42, No. 10, pp. 958–964, 2001.
[2] 北 研二, 津田 和彦, 獅々堀 正幹 :『情報検索アルゴリズム』, 共立出版, 2002.
[3] 小泉 大地, 柘植 覚, 獅々堀 正幹, 北 研二 : “色情報分布に基づく類似画像検索システムの開発”, 情報処理

学会四国支部研究シンポジューム論文集, pp. 35–38, 2002.

[4] 小泉 大地, 柘植 覚, 獅々堀 正幹, 北 研二: "テキストと画像のクロスメディア情報検索に向けた画像キーワード登録システムの開発", 情報処理学会情報学基礎研究会, 2002.

[5] Aggarwal, C.C., Hinneburg, A. and Keim, D. A.: "On the surprising behavior of distance metrics in high dimensional space", *Proceedings of the 8th International Conference on Database Theory*, pp. 420–434, 2001.

[6] Berry, M. W., Drmač, Z. and Jessup, E. R.: "Matrices, vector spaces, and information retrieval", *SIAM Review*, Vol. 41, No. 2, pp. 335–362, 1999.

[7] Beyer, K.S., Goldstein, J., Ramakrishnan, R. and Shaft, U.: "When is "nearest neighbor" meaningful?", *Proceedings of the 7th International Conference on Database Theory*, pp. 217–235, 1999.

[8] Bozkaya, T. and Özsoyoglu, Z. M.: "Indexing large metric spaces for similarity search queries", *ACM Transactions on Database Systems*, Vol. 24, No. 3, pp. 361–404, 1999.

[9] Ciaccia, P., Patella, M. and Zezula, P.: "M-tree: An efficient access method for similarity search in metric spaces", *Proceedings of the 23rd International Conference on Very Large Data Bases* (VLDB'97), pp. 426–435, 1997.

[10] Flickner, M. et al: "Query by image and video content: The QBIC system", *IEEE Computer*, Vol. 28, No. 9, pp. 23–32, 1995.

[11] Gaede, V. and Günther, O.: "Multidimensional access methods", *ACM Computing Surveys*, Vol. 30, No. 2, pp. 170–231, 1998.

[12] Gionis, A., Indyk, P. and Motwani, R.: "Similarity search in high dimensions via hashing", *Proceedings of the 25th International Conference on Very Large Data Bases*, pp. 518–529, 1999.

[13] Guttman, A.: "R-trees: A dynamic index structure for spatial searching", *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 47–57, 1984.

[14] Katayama, N. and Satoh, S.: "The SR-tree: An index structure for high-dimensional nearest neighbor queries", *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 369–380, 1997.

[15] Kohonen, T.: *Self-Organizing Maps*, Springer, 1995.

[16] Kohonen, T. et al: "Self organization of a massive document collection", *IEEE Transactions on Neural Networks*, Vol. 11, No. 3, pp. 574–585, 2000.

[17] Liao, S., Lopez, M. A. and Leutenegger, S. T.: "High dimensional similarity search with space filling curves", *Proceedings of the 17th International Conference on Data Engineering*, pp. 615–622, 2001.

[18] Oja, E., Laaksonen, J., Koskela, M. and Brandt, S.: "Self-organizing maps for content-based image database retrieval", *Kohonen Maps*, (Oja, E. and Kaski, S. Eds.), Elsevier, pp. 349–362, 1999.

[19] Pentland, A., Picard, R. and Sclaroff, S.: "Photobook: Content-based manipulation of image databases", *International Journal of Computer Vision*, Vol. 18, No. 3, pp. 233–254, 1996.

[20] Porter, M. F.: "An algorithm for suffix stripping", *Program*, Vol. 14, No. 3, pp. 130–137, 1980.

[21] Salton, G., Wong, A. and Yang, C. S.: "A vector space model for automatic indexing", *Communications of the ACM*, Vol. 18, No. 11, pp. 613–620, 1975.

[22] Sasaki, M. and Kita, K.: "Vector space information retrieval using concept projection", *Proceedings of the 19th International Conference on Computer Processing of Oriental Languages* (ICCPOL 2001), pp. 73–76, 2001.

[23] Shepherd, J., Zhu, X. and Megiddo, N.: "A fast indexing method for multidimensional nearest-neighbor search", *Proceedings of the SPIE Conference on Storage and Retrieval for Image and Video Databases*, pp. 350–355, 1999.

[24] White, D. A. and Jain, R.: "Similarity indexing with the SS-tree", *Proceedings of the 12th IEEE International Conference on Data Engineering*, pp. 516–523, 1996.

[25] Yianilos, P. N.: "Data structures and algorithms for nearest neighbor search in general metric spaces", *Proceedings of the Fourth ACM-SIAM Symposium on Discrete Algorithms*, pp. 311–321, 1993.