2C3-01

汎用検索手法の２次元データへの拡張

Expansion of Generic Search Method for Two-dimensional Data

藤本 敦[*1] Atsushi Fujimoto　足立 史宜[*1] Fuminori Adachi　鷲尾 隆[*1] Takashi Washio　元田 浩[*1] Hiroshi Motoda　丹羽 雄二[*2] Yuji Niwa　花房 英光[*2] Hidemitsu Hanafusa

[*1]大阪大学産業科学研究所 I.S.I.R., Osaka University　[*2]原子力安全システム研究所 INSS Inc.

The generic search method on byte patterns applied mathematical invariance is developed in our laboratory. And we confirmed its efficiency on text data. In this work,we propose a way of expantion of the method to two-dimentional data. The experimental evaluation in this work indicates its high feasibility.

1. はじめに

当研究室では、数学的不変性を用いたビット列表現での汎用検索システムが開発されており、テキストデータにおいてその性能が確認されている。この手法を用いた２次元データ（図など）への拡張方法を提案し、その手法に基づく検索システムを計算機に実装した。次に、実在するビットマップ画像ファイルについてその手法を適用し、性能評価を行った。

2. 従来の汎用検索手法

開発された手法ではまず、個々の文書ファイルの先頭を開始点とする決められた長さの連続したデータ系列を切り出す。その切り出したデータ系列に離散フーリエ変換を行い、さらに量子化することによって求まった変換係数列をそのファイルの特徴の１つである「特徴ベクトル」と呼ぶことにする。そして、切り出し開始点を１バイトずらして変換を行い特徴ベクトルを作成する。この重ね移動窓による切り出しをデータ系列の末端がファイルの末端に到達するまで繰り返す。そうしてできた特徴ベクトル群を集計することで逆引き情報を作成し、対象ファイルすべてに対して逆引き情報を作成し、検索対象のファイルの特徴ベクトルと逆引き情報とを比較することによって検索を行っている。

この手法は文書ファイルでは有効であるが、２次元データは左右だけでなく上下にも相関があるため、２次元データに対しては適用することができない。また、２次元データは情報量が多いので、その損失を抑えつつデータを圧縮する必要がある。

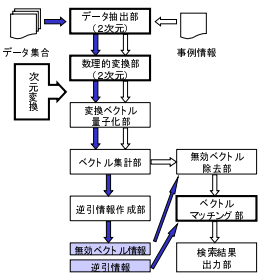

図 1: 汎用検索手法の概要

連絡先: 藤本 敦 大阪大学 産業化学研究所 567-0047 茨木市 美穂ヶ丘 8-1 Tel (06)6879-8542 Fax (06)6879-8544 e-mail: fujimoto@ar.sanken.osaka-u.ac.jp

図１は拡張された汎用検索手法の概要であり、図の太枠の部分が本研究で拡張した箇所である。

3. ２次元フーリエ変換の不変性

3.1 ２次元離散コサイン変換

離散コサイン変換 (DCT) は時間領域または位置領域のデータ系列を周波数領域のデータ系列に変換するもので、類似したデータ系列は周波数の偏りが類似し、多次元のデータ系列にも適用でき、変換後の係数が実数であるため計算機で扱いやすいことから、画像圧縮などの分野で非常によく使われている変換である。そこで、一様乱数でランダムに８×８バイトの２次元データを用意して２次元 DCT を用いて変換し、そのデータを反転、回転、巡回シフトなどを行ったデータ系列に対しても同様に DCT を行い、変化を調べた。

まず、２次元データを上下、左右に反転させたデータに DCT を行った系列を比べると、左右反転させた場合は奇数列が、上下反転させた場合は奇数行がそれぞれ逆符号になっているが、絶対値はほぼ同じであった。次に、２次元データを 90 度、180 度回転させたデータ系列について DCT を行った結果を調べると、90 度回転では (0,0)(7,7) 成分を繋ぐ直線を対称に入れ替わって奇数行が逆符号になっており、180 度回転では奇数行、列が逆符号になっていた。最後に、データを巡回シフトさせたデータ系列にについて変換した系列を比べると、水平移動させた場合は 0 列目の成分が変化せず、斜め方向に移動させた場合は (0,0) 成分以外のデータが全て変化した。

表 3.1 にデータを斜め方向に巡回シフトさせた結果を掲載する。巡回シフトとは、図 2 に示すようにデータをある方向に平行移動させ、はみ出た部分を移動させた向きと逆側の端に連結させることである。斜め方向シフトの場合は、斜めに１バイトデータをずらすことを 2 回繰り返すと 1/4 シフトとなる。

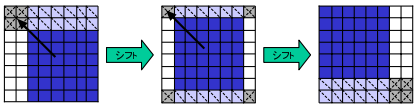

図 2: 1/4 斜め方向シフトの様子

表 3.1 : データの斜め方向巡回シフトによる DCT 係数の変化

元データ	0	1	2	3	4	5	6	7
0	1523	-124	424	220	321	202	619	-13
1	161	116	-48	68	249	-151	-228	22
2	243	160	84	-104	430	-72	-184	-284
3	91	421	-257	46	-403	-729	-152	-368
4	270	82	-167	542	121	-266	90	-681
5	-384	-42	-13	145	-147	302	236	468
6	-92	-329	-83	94	-204	583	-270	-772
7	92	306	191	-169	1023	-124	-62	-649

1/4 シフト	0	1	2	3	4	5	6	7
0	1523	-189	-235	427	-323	566	-210	-176
1	-355	195	-359	339	196	76	-281	98
2	130	-24	-131	136	-405	-112	-68	235
3	-158	16	-256	486	-485	-174	-554	-24
4	-272	172	153	658	121	-200	287	511
5	119	-211	59	-510	-38	-567	-464	-659
6	264	-234	101	60	-614	-296	-139	-435
7	76	-272	548	-262	-840	143	203	-216

1/2 シフト	0	1	2	3	4	5	6	7
0	1523	166	-425	-157	321	-174	-620	150
1	83	-275	81	336	-110	-245	31	-634
2	-244	229	84	-48	-431	141	-184	216
3	405	-262	40	364	426	-255	360	-656
4	271	188	166	189	121	-750	-91	-449
5	60	-249	-310	-434	903	-148	-57	-56
6	91	89	-83	778	203	635	-270	186
7	-122	-222	-35	234	531	-202	-12	-127

この巡回シフトによる係数の変化は、DCT が長さ N のデータを 2N の周期として周波数を計算するからである。

以上より、2 次元 DCT は 1 つの 2 次元データにつき 1 つの DCT 系列を持つが、絶対値を考えれば 1 つの DCT 系列で数種類の 2 次元データを表すことができ、情報を圧縮することが可能である。

3.2 2 次元離散フーリエ変換

離散フーリエ変換 (DFT) も時間領域や一領域のデータ系列を周波数領域に変換するものであり、ディジタル信号処理等において幅広く利用されている。本研究では、DFT の計算の冗長性を省き、計算を高速に行う高速フーリエ変換 (FFT) を 2 次元に拡張したアルゴリズムを用いて、8 × 8 バイトの 2 次元データを変換した系列を調べ、反転、回転、巡回シフトによる変化を調べた。ただし、フーリエ係数は複素数となるため、その係数の絶対値を考え、移送は絶対値に比べて類似性を確認できる要素ではないので切り捨てる。

まず、データを反転させて変換した系列を比べると、上下、左右反転ともに係数に変化はなかった。次に、データを回転して変換した系列を比べると、90 度回転では 0 行目と 0 列目の係数が入れ替わり、他の係数は 90 度回転させたように変化していた。また、180 度だとまったく係数が同じになっていた。最後に、巡回シフトしたデータのフーリエ変換後の係数を調べると、係数に変化は見られなかった。表 3.2 に巡回シフトの結果を掲載する。

表 3.2 : データを巡回シフトしたときのフーリエ係数の変化

元データ	0	1	2	3	4	5	6	7
0	3048	686	456	924	156	924	456	686
1	390	357	720	261	480	242	469	130
2	594	681	608	399	1170	643	1092	246
3	395	563	969	831	1070	748	327	255
4	20	581	1226	200	956	200	1226	581
5	395	255	327	748	1070	831	969	563
6	594	246	1092	643	1170	399	608	681
7	390	130	469	242	480	261	720	357

1/4 シフト	0	1	2	3	4	5	6	7
0	3048	686	456	924	156	924	456	686
1	390	357	720	261	480	242	469	130
2	594	681	608	399	1170	643	1092	246
3	395	563	969	831	1070	748	327	255
4	20	581	1226	200	956	200	1226	581
5	395	255	327	748	1070	831	969	563
6	594	246	1092	643	1170	399	608	681
7	390	130	469	242	480	261	720	357

1/2 シフト	0	1	2	3	4	5	6	7
0	3048	686	456	924	156	924	456	686
1	390	357	720	261	480	242	469	130
2	594	681	608	399	1170	643	1092	246
3	395	563	969	831	1070	748	327	255
4	20	581	1226	200	956	200	1226	581
5	395	255	327	748	1070	831	969	563
6	594	246	1092	643	1170	399	608	681
7	390	130	469	242	480	261	720	357

係数に変化が見られないのは、DFT が長さ N のデータを N の周期で無限に拡張した関数のフーリエ変換に相当するからである。また、図 3 に示すように 2 次元 DFT は複素対称性を持ち、変換領域においておよそ半分の値が冗長であり、切り捨てることができる。

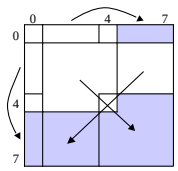

図 3: 2 次元 DFT の冗長部

この結果から、DCT や DFT を行うことで情報をあまり失うことなくデータを圧縮し、類似のデータを判別できると考えられる。今回構築するシステムは 2 次元 DFT を用いたものにした。その理由は、(1) データの反転、回転などに同じフーリエ係数を持ち、1 つの情報で多くの情報を有することができ、圧縮につながる。(2) 2 次元 DFT は約半分の情報が冗長であり、データの削減ができる。(3)FFT アルゴリズムを取り入れた変換をするため、DCT に比べて計算が高速にできる。

4. 2 次元データへの拡張法

この手法を 2 次元データにおいても適用可能にするためには、前節で述べたようにデータの上下の相関も考慮に入れなければならないため、切り出す移動窓を 2 次元の正方形 (ただし一辺は 2 のべき乗) にする必要がある。切り出し方は、まずデータの左上から正方形の窓の大きさのデータを切り出し、1 つのデータ系列とする。そして、切り出しの開始点を 1 バイト右にずらしながら同様の窓のデータをデータ系列として切り出し、窓の右端部が 2 次元データの右端部に到達するまで繰り返す。それが終われば切り出し開始点を 1 バイト下にずらした状態で左端に戻して上のことを繰り返し、窓の右下部がファイルの右下部に到達した時点で終了する。これを図に表すと以下の図 4 のようになる。

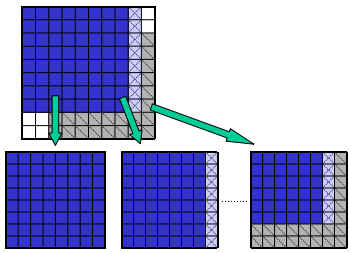

図 4: データの切り出し (一辺が 8 ドットの場合)

そして切り出されたデータ系列全てに対して２次元フーリエ変換を行う。ここで、２次元フーリエ変換は複素数であり絶対値と位相の情報を持つが、位相は絶対値より類似性を確認できる要素ではないのですべての変換系列を絶対値に置き換える。また、２次元フーリエ変換は複素対称性を持っているため約半分のデータが冗長となるため切り捨てることができる。さらに、２次元データは隣接部と相関が強く、低周波成分に電力が集中する傾向があるために高周波部を切り捨てることでデータ量を少なくすることができる。これらより、ジグザグスキャンを行うことによって、情報の損失を抑えつつ１次元のデータに変換することができる。これは、図 5 で表される。ただし、色の濃い部分は冗長部である。以上から、２次元データを従来の汎用検索システムで利用できるような１次元の特徴ベクトル群に変換することができる。

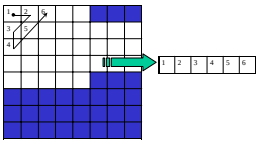

図 5: ２次元から１次元への変換 (スキャン数６の場合)

5. ビットマップファイルによる性能評価

提案した汎用検索手法を実装したプログラムを作成し、性能評価を行う。使用するデータは原子力発電所に関するビットマップ画像ファイルであり、そのファイル数は 101 個、サイズは 5KB から 8MB である。別に画像の種類 (写真、概観図、断面図等) の特徴を記したラベルデータが提供されている。そして、ファイル群に対する重ね移動窓のサイズを 8×8、ベクトル量子化の分割数を 12、無効ベクトル情報の作成の閾値を 70%として逆引きファイルを作成し、評価を行う。

用いた評価指標は比率の差の検定で、この検定は２つの母比率 π_1、π_2 の相当性を検定するものである。つまり、π_1 はファイル全体における正解ファイルの割合であり、π_2 は検索結果のファイル群における正解ファイルの割合である。検索システムの評価はランダムで引いたときと比べてその結果が良いかどうかであるので、π_1 と π_2が等しくなく検索結果の方が優れているということが言えればよい。これを確かめるには、検索結果に対してある有意水準 α(結論を誤る確率) を設定して右片側検定を行い、その実現値 z が棄却域 (優れているという結果が偶然でないと言える領域) に含まれれば検索システムはランダムで引いた結果よりも優れていることになる。

ここで、実現値 z について説明する。まず、母比率が不明であるので、２つの標本比率より母比率を推定する。ある現象を独立に n 回試みたとき、x 回生起したときの標本比率を $p = x/n$ とすると

$$\hat{\pi_1} = p_1 = \frac{x_1}{n_1}, \hat{\pi_2} = p_2 = \frac{x_2}{n_2}$$

母分散の推定を行い、２項分布の正規近似を用いて

$$z = \frac{p_1 - p_2}{\sqrt{\frac{\hat{\pi_1}(1-\hat{\pi_1})}{n_1} + \frac{\hat{\pi_2}(1-\hat{\pi_2})}{n_2}}}$$

$$= \frac{p_1 - p_2}{\sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}}} \sim N(0.1)$$

で求められる。また、今回は標本数が少ないため Yates 補正を行う。補正は次式で行われる。

$$z = \frac{|p_1 - p_2| - 0.5(1/n_1 + 1/n_2)}{\sqrt{\frac{p_1(1-p_1)}{n_1} + \frac{p_2(1-p_2)}{n_2}}}$$

この z に関しての棄却域を求めて検定を行えばよい。

今回の有意水準は 0.05 に設定した。この α に対応する棄却域に含まれれば 95%の信頼度で有意差があると言える。評価は、検索結果上位５ファイルについてラベルが一致しているかの検定を行った。また、所持ベクトル数を考慮に入れた規格化式を数種類検討し、評価的、視覚的に最良の規格化式を選出する。

まず、逆引きファイルに記された i 番目の２次元データの持つ特徴ベクトルが、検索対象のファイルの持つ特徴ベクトルと一致した数を $f(i)$、検索対象の２次元データの持つベクトル数を x、i 番目の２次元データの持つベクトル数を $y(i)$ とし、規格化後の一致数を $F(i)$ とする。最初に考案した式は、

$$F(i) = f(i) \cdot \frac{x}{y(i)}$$

である。ベクトル数の比率をかけることで規格化がなされると考えた。これを用いて検定を行うと表 5.1 のようになった。

表 5.1：ベクトル数で規格化した結果

ファイル名	全体に含まれる正解ファイル数	p_1	検索結果に含まれる正解ファイル数	p_2	実現値
31	11	0.11	1	0.2	-0.10
61	8	0.08	1	0.2	0.12
91	39	0.39	5	1.0	2.24

まず、$\alpha = 0.05$ のときの棄却域は $R = \{z||z| > 1.65\}$ であり、この結果を見ると 1/3 しか棄却域に含まれておらず検索システムとして有効であるとは言いにくい。また、実際のファイルを見ると検索によって抽出されたファイルはサイズの小さいものに偏っていた。そして、視覚的にも似ているとは言い難く、写真のような特徴ベクトルを多く持ったファイルが抽出された。

このことを踏まえて次の案を考えた。最初に考案した規格化式は、結果からサイズの小さなファイルに有利に働くため、所持ベクトル数の比率を和らげた次の式を検討する。

$$F(i) = f(i) \cdot \sqrt{\frac{x}{y(i)}}$$

表 5.2：ベクトル数の比率を和らげた結果

ファイル名	全体に含まれる正解ファイル数	p_1	検索結果に含まれる正解ファイル数	p_2	実現値
31	11	0.11	2	0.4	1.23
61	8	0.08	3	0.6	2.97
91	39	0.39	5	1.0	2.24

表 5.2 を見ると２つのファイルが棄却域に含まれ、検索システムとして有意性が上がったと言える。しかし、依然として写真ファイルが多く抽出される傾向がある。これは写真のような所持ベクトルの多いファイルは必然的に一致するベクトル数が多くなり、構造図のような所持ベクトルの少ないデータでは類似していたとしても写真データの一致数を超えない可能性があるからと考えられる。

これより、ベクトル一致数が画像の種類によって変化することを考慮に入れたものを考案した次の式を用いて検定を行った。表 5.3 を見ると全てのファイルが棄却域に含まれ、実現値も上がっている。また、抽出されたファイルの写真への偏りもなくなり、この規格化法が一番有効であると考えられる。

$$F(i) = \frac{f(i)}{\sqrt{x \cdot y(i)}}$$

表 5.3 : 画像の種類によるf(i)の変化を考慮に入れた結果

ファイル名	全体に含まれる正解ファイル数	p_1	検索結果に含まれる正解ファイル数	p_2	実現値
31	11	0.11	3	0.6	2.48
61	8	0.08	4	0.8	4.24
91	39	0.39	5	1.0	2.24

6. おわりに

実用データの評価結果より、提案手法は２次元データ検索として使用できる可能性があることがわかったが、まだ今回の実験ではサンプル数が少なく、まだ実用的であるとは判断できない。評価に用いたラベルが人為的につけられているためにあいまいになってしまっていることも評価を下げた原因ではないかと考えられる。また、２次元データはデータ量が多く、計算に時間がかかってしまう。

本研究では、汎用検索手法の２次元データへの拡張について提案し、実在データへの適用を行い、その性能を評価した。また、数学的変換を施し、その結果に対して量子化を行うことによって、パターンをベクトルと呼ぶ量に変換することにより、2 つのデータ系列が類似しているか否かの判定を行えることが 2 次元においても言えることがわかった。また、本研究の手法では２次元データとしてビットマップファイルしか扱えないため、JPEG ファイルなどの多種の２次元データにも対応させて、ファイル形式に囚われない検索手法にすることが今後の課題である。

参考文献

[1] Baeza-Yates, R.A. String Searching Algorithms, Information Retrival, Data Structures & Algorithms, Capter 10, ed. Baeza-Yates, R.A., New Jersey.

[2] Harman, D.,Fox, E. and Baeze-Yates, R.A. Inverted Files, Information Retrieval, Data Structures & Algorithms,Capter 3, ed. Baeze-Yates, R.A.,New Jersey Prentice Hall, pp.28-43, 1992

[3] Faloutsos, C. Signature Files, Data Structures & Algorithms, Chapter 4, ed. Baeze-Yates , R.A., New Jersey Prentice Hall, pp44-65, 1992

[4] Faloutsos, C. Access Methods for Text. ACM Computing Survey. pp.17,50-74, 1985

[5] Belkin, N.J. and Croft, W.B. Retreval Techniques, Annual Review of Information Science and Thechnology, ed. Williams, M., New York. Elservier Science Publishers, pp.109-145, 1987.

[6] 寺田 文行 , 中村 直人 , 釈氏 孝浩 , 松井 辰則. ライブラリ理工基礎数学-7 情報数学の基礎 －暗号・符号・データベース・ネットワーク・CG －.

[7] 白井 良明 , 谷内田 正彦. 新コンピュータサイエンス講座 パターン情報処理. オーム社,pp14, 1998.

[8] 電子通信学会. ディジタル信号処理 第１０版. 技報堂, pp.49-61, 1983

[9] 南 敏 , 中村 納. テレビジョン学会教科書シリーズ 1 画像工学 -画像のエレクトロニクス-. コロナ社, pp.61-64, 1989