分散並列処理による文書画像データベースシステムの設計

4J−2

仙田修司　美濃導彦　池田克夫

京都大学工学部

1　はじめに

大量の文書画像を蓄積するシステムとして光ディスクを用いた画像ファイリングシステムがすでに実用化されているが、検索時に必要な書誌情報は人手で入力する必要があり、この作業がユーザに大きな負担となっている[1]。この負担を軽減するため、認識処理によって文書画像を理解し、自動的に書誌情報を登録することが考えられる。しかし、現在の認識処理技術では十分な認識率が得られないという問題がある。

これらの状況をふまえ、認識処理の途中結果をDBに登録し、検索手法をそれに合わせることで認識結果の誤りによる検索漏れを防ぐ手法を提案する。この手法では、検索時の処理量が既存のDBよりも大幅に増加するが、分散並列処理による高速化によって実用的な検索時間を実現することが可能となる。

2　文書画像認識結果によるDB構築

2.1　認識結果の意味レベル

データの持つ意味レベルは、同じ情報を表す二つのデータのうち、データを表現するための記憶量が少ない方を意味レベルが高いと定義する。認識処理とは、いわば、与えられたデータをより意味レベルの高いデータに写像する処理である。例えば、文字画像(文字を1, 背景を0とした2値画像)からパターンマッチングを用いて文字候補(複数の文字コード)を得る処理や、文字候補から単語照合を用いて単一の文字コードを得る処理などである。認識処理においては、一般に、認識結果の意味レベルが高いほど認識誤り率も高くなる。

このことから、意味レベルの高い認識結果は用いずに、網羅的で意味レベルの低い認識結果をDBに登録し、検索時に、ユーザから与えられたキーの意味レベルを下げてマッチングを行う手法が有効であると考えられる。例えば、文字認識結果を利用する場合、認識処理の途中結果である(複数の)文字候補をそのままDBに登録し、検索時には登録されている全ての候補についてマッチングを行う手法である(図1)。この手法により、単一の文字コードのみをDBに登録した場合に比べて検索漏れが大幅に減少し、十分実用に耐え得るDBを構成することが可能となる。

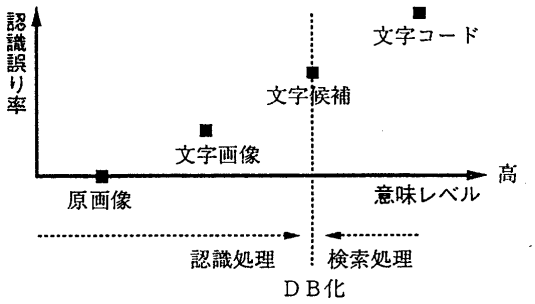

図1. 文字認識の意味レベルと認識誤り率

2.2　文書画像DBへの適用

文書画像の本文を文字候補としてDBに登録した場合には、全文検索の技術を応用した検索手法を用いることができる。具体的には、有限オートマトンを用いてマッチングを行うAho-Corasick法[2]を拡張し、状態遷移の際に文字候補の数だけの遷移を同時に行えばよい。この場合、現状態が一つではなく複数の状態の集合となり、次状態を求める際もこれら全ての状態について考える必要がある。原文の文字数をn、一文字当たりの文字候補数をc、オートマトンの状態数をmとすれば、検索対象となる文字列はc^n通りの組み合わせがあるのに対して、計算量は高々$O(n \times c \times m)$ですむ。よって、従来の単一文字コードのみを登録したDBを全文検索する場合と比べて、$c \times m$倍の計算量となる。

認識結果による文書画像DBの構成要素としては、文字認識結果以外のものも考えられる。例えば、文書画像を画像のまま扱い、文字の色や位置、領域の色や位置などを用いて検索することも可能である[3]。この場合、原画像の解像度や色数を縮小した画像と、ユーザが提示した概略画をぼかしながらマッチングすることで検索時間の短縮と曖昧な検索を実現している。

Design of a Document Image Database System by Distributed Parallel Processing

Shuji SENDA, Michihiko MINOH, Katsuo IKEDA

Kyoto University

3 分散並列処理環境

3.1 分散型文書画像DBシステムの構成

分散並列処理環境における文書画像DBシステムの概要を図2に示す。システムは認識処理部と検索処理部に大別できる。ユーザは、まず、スキャナで取り込んだフルカラー文書画像を分散データマネジャに登録する。そして、分散データマネジャにあらかじめ登録されている文書画像認識用のモジュールパッケージを起動する。文書画像認識モジュールパッケージでは、書式情報を用いた構造解析モジュールにより領域の判別を行い、さらに、それぞれの領域に固有の認識モジュールを起動して認識結果を分散データマネジャに登録する。文書画像を検索する場合、ユーザは分散データマネジャに対して検索用モジュールパッケージの起動を要求する。検索用モジュールは複数個同時に起動され、与えられたキーと登録された文書画像とのマッチングを行い、一致した項目を分散データマネジャに返す。このように、認識処理部・検索処理部ともに、分散並列処理による高速化を特徴としたシステム構成となっている。

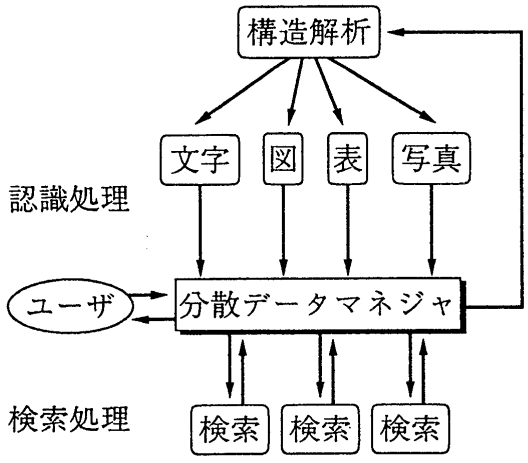

図2. 分散並列処理型文書画像DBシステム

3.2 分散データマネジャと関数型モジュール

本稿では、分散データマネジャによるデータ分散、関数型モジュールによる自律的な負荷分散を特徴とする分散並列処理モデルを提案する。

まず、このモデルの二つの構成要素について述べる。

分散データマネジャ システム全体で共有されるデータとシステムを管理するためのメタデータを保有する。分散データマネジャは各プロセッサ毎に全く同一の機能を有するものが存在するが、それぞれが保有するデータおよびメタデータは同一ではない。しかし、どの分散データマネジャからも全ての共有データが等価にアクセスでき、キャッシュやパスの選択によりアクセスにかかるコストの最適化を行う。また、ユーザはユーザインタフェースを介して、共有データの追加、削除、モジュール呼出しによるデータの加工を行うことができる。

関数型モジュール 入力データに対して出力データが一意に定まる関数的な処理ルーチンである。モジュールを呼び出す際には、複数のモジュールをストリーム(データの通る路)で接続することにより、分散データマネジャを介さずにパイプライン的な処理を行うことができる。実際にモジュールがどのプロセッサに割り当てられるかは動的かつ自律的に決定される。

上記のモデルの利点には次のようなものがある。まず、共有データを管理する分散データマネジャを設けたことで、分散の透明性が達成される。しかも、分散データマネジャは、分散DBMSのサブセットといえるので、分散DBMSの開発に用いられた手法の多くが利用可能である。また、実際にデータに対して処理を行うモジュールを関数型に制限したことで、モジュールを動的にプロセッサに割り当てること、障害が起きた際に別のプロセッサで処理の続きを継続することが可能となる。並列処理の観点からは、複数のモジュールのパイプライン結合により、効率的な負荷分散を行うことができる[4]。

4 おわりに

認識結果の意味レベルに関して考察し、意味レベルの低いデータを用いてDBを構成する手法を提案した。また、分散データマネジャと関数型モジュールから構成される分散並列処理環境を提案し、その環境で動作する文書画像DBシステムを設計した。

今後は、提案した分散並列処理環境と文書画像DBの実装を行っていく予定である。

参考文献

[1] 中野, 藤澤: "自動ファイリングのための文書理解の一方式," 信学論, vol.J71-D, no.10, pp. 2050-2058.

[2] A.V.Aho, M.J.Corasick: "Efficient String Matching: An Aid to Bibliographic Search," Communications of the ACM, vol.18, no.6, pp. 333-340, 1975.

[3] 森, 美濃, 池田: "概略画を用いたマルチメディア文書画像の検索," 第45回情処全大, 1992.

[4] 仙田, 美濃, 池田: "文書画像理解のための分散処理方式," 第44回情処全大, 1-233, 1992.