ロジカルフォーマットによる帳票処理

5F－8

平山唯樹
日本アイ・ビー・エム株式会社 東京基礎研究所

1　はじめに

フィールドや罫線の位置や大きさなどは異なるが、そのトポロジカルな位置関係は共通しているフォーム(帳票)のフォーマットをここではロジカル(トポロジカル)フォーマットと呼んでいる。ロジカルフォーマットを用いた帳票処理の大きな特徴は、一つのフォーマットで異なるフィジカルフォーマットを持った複数の帳票を処理できることである。

筆者は前講演[1]でLSA(Line-Shared-Adjacent)セルフォーマットを用いてこのロジカルフォーマットを表現する方法について報告した。本稿では実際にスキャンしたフォームのイメージをどのように処理するかについて報告する。

2　フォーム解析の要点

フォーム処理の手法を現実に使われているフォームのイメージに適用する場合、次の点に留意する必要がある。(1) イメージには汚れやかすれが多く存在している場合がある。つまり、フォーム処理の手法が汚れやかすれに強くなくてはならないということ。(2) 実際のフォームからフォーマット（書式）情報を容易に取り出すことができること。(3) 新しいフォームが出現した時に、そのフォーマットをシステムに容易に取り込めること。ここでは、以上の点を念頭において本方法について述べる。

3　フォーム解析の手順

3.1　前処理

スキャンにより得られたフォームのイメージは前処理によって線分の情報が抽出される。縦横それぞれの線分は、始点・終点からなるベクトル情報として抽出される。このベクトル情報とフォーマットの情報をマッチングすることによってフォームを解析する。

Forms Processing by Using Logical Format
Yuki HIRAYAMA
IBM Research, Tokyo Research Laboratory, IBM Japan Ltd.

3.2　フォーマット変換

前節で述べたように、イメージからは線分のベクトルデータが抽出される。一方でLSAフォーマットはセル(線分で囲まれた矩形領域)を使って表現されている。セルや交点のデータはこの線分のデータを基に作られているので、これらの情報は線分データの信頼性に依存しているということができる。したがって、フォーマットとイメージ情報とのマッチングには線分レベルで行った方がより信頼性が高くなると考えられる。

そこで、LSAフォーマットを罫線による表現(LO(Line Oriented)フォーマット)に変換する。LSAフォーマットの基本単位は二つのセルの位置関係であるが、これは、すなわち二つのT字型に接続した罫線の位置関係と考えることもできる。これにより、二つのセルの位置関係を表すLSAは二つの罫線の位置関係の情報に同値変換をすることができる。これによりLSAフォーマット全体をLOフォーマットに同値変換をする。

Horizontal line										Vertical line					
Line	Connection									Line	Connection				
A(A)										a(L)					
A(B)	a	N(H)	b	N(T)	c	Z(H)	d	Z(T)	h	a(R)	A	N(H)	B	A(H)	D
B(A)	a	N(H)	b	N(T)	c					b(L)	A	N(H)	B		
B(B)	a	A(H)	e	A(T)	f					b(R)	A	N(T)	B		
C(A)	c	Z(H)	d	Z(T)	h					c(L)	A	N(T)	B		
C(B)	f	T(H)	g	T(T)	h					c(R)	A	Z(H)	C		
D(A)	a	A(H)	e	A(T)	f	T(H)	g	T(H)	h	d(L)	A	Z(H)	C		
D(B)										d(R)	A	Z(T)	C		
										d(R)	A	Z(T)	C		
										e(L)	B	A(H)	D		
										e(R)	B	A(T)	D		
										f(L)	B	A(T)	D		
										f(R)	B	T(H)	D		
										g(L)	C	T(H)	D		
										g(R)	C	T(T)	D		
										h(L)	A	Z(T)	C	T(T)	D

図1: LOフォーマット

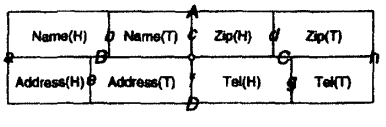

図2: LSAフォーマットにおける罫線

図1にLOフォーマットを示す。なお、A～D、a～hは図2にあるように、LSAフォーマットにおける罫線を

表す。このLOフォーマットは次のように見る。表の左半分は水平罫線、右半分は垂直罫線についての情報である。それぞれの罫線がどの罫線、セルと接続しているか、また接しているかを示している。例えば、左半分のA(A) は罫線 A の上側 (Above)、A(B) は下側 (Below) を示す。A(B) 欄では、下半分で、左から、罫線 a、セル N(H)、罫線 b、セル N(T)…セル Z(T)、罫線 h と接続する（接する）ことを示している。

3.3 マッチング

トポロジカルソート LO フォーマットの基本表現であるT字型の罫線の接続において、T字型の水平線を罫線α、縦線を罫線βとするとき、「βはαに依存している」と呼ぶことにする。これは、βの端点の位置がαの位置によって決まるからである。すべての罫線を、この依存関係にしたがってソート（トポロジカルソート）する。

マッチング 前処理によって抽出された線分とフォーマットの罫線とのマッチングを行う。マッチングは前節でソートされた順に行う。マッチングの結果は評価関数によって評価される。この評価関数では、そのマッチングの結果余っている線分の数や長さ、もしくはかすれている線分の数や長さ、マッチングによってできたセルに関する情報を評価値として表す。

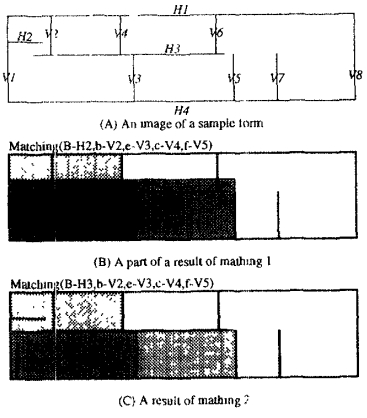

図 3: マッチング例

汚れ、かすれに対する耐性 ロジカルフォーマットを用いたフォーム処理では基本的には罫線やフィールドの長さや場所の情報を持っていないので、フィジカルフォーマットを用いた処理に比べて汚れやかすれにより受ける影響が大きい。そこで、本方法ではセルや交点といったより影響を受けやすい情報ではなく、よりプリミティブな情報である罫線を基にしたLOフォーマットを用いて解析をしている。また、LOフォーマットは罫線の接続情報が記述されているので余分な罫線がある場合やかすれている場合でも、より正確に抽出線分と罫線とをマッチングさせることが可能である。図3にマッチングの例を示す。(A) がスキャンで得られたサンプルフォームのイメージ。このイメージから得られた線分と、LOフォーマット中の罫線のマッチングを行う。(B) と (C) との大きな違いは罫線Bを線分H2 または線分H3 とマッチングしているということである。結果としては (C) がマッチングの結果として採用される。

4 フォーマットの生成および拡張

本方法では新規フォームからのフォーマットの生成およびそれによるフォーマットの拡張も容易に実現できる。

フォーマットの生成は次の手順によって行う。(1) 実際のフォームを解析して線分を抽出する (2) 線分の接続状況を調べて、LO フォーマットを生成する。(3)LO フォーマットを LSA フォーマットに変換する。

また、ここで得られた LSA フォーマットを次の手順で従来のフォーマットに統合し、拡張することができる。(1) 従来のフォーマットとこの新しいフォーマットの間で統合演算を行う。(2) 統合演算の結果、取り込むことができる場合にはフォーマットを更新する。また、取り込むことができない場合には従来と異なるフォーマットとして登録する。つまり、本方法では従来のフォーマットに取り込むことができるかできないかが統合演算によって明確にわかる。

5 まとめ

本方法の特徴は、フォーマットの生成、拡張などの管理はLSA フォーマットで行い、実際の処理はそれと同値変換された LO フォーマットで行うことができるということである。これにより、階層的クラスで管理されたフォーマットに基づいて、より汚れやかすれに強いフォーム処理が実現されている。

参考文献

[1] 平山唯樹, “Line-Shared-Adjacent(LSA) セルフォーマットを用いたフォーム処理,” 第 53 回情処全国大会, 1996, vol.2, pp.279-280.

[2] Y. Hirayama, “Analyzing Form Images by Using Line-Shared-Adjacent Cell Relations,” 13th ICPR, August 1996, vol.III, pp.768-772.