

2N-3

Raster演算を用いた 設備図面のシンボル認識

亀井克之 中村泰明 阿部 茂

(三菱電機 中央研究所)

1. はじめに

テンプレートマッチングは簡単かつタフな手法であるが、計算時間がかかるという欠点があった。テンプレートを動かす代わりに、テンプレート上のサンプル点に応じた距離だけ画像を動かし、元の画像との論理演算を行う操作を繰り返せば、シンボルの存在する候補点を抽出できる。これはワークステーションの Raster 演算で極めて高速に実行できる。この粗サーチの後、精密なマッチングを行い認識率を上げる。論理演算を工夫すれば粗サーチのリジェクト率を小さくできる。1000x800 の画像上で1種類多数のシンボルを認識する時間は、2~8秒と極めて速い。

2. 設備図面の認識

電力配電系統図や上下水配管系統図は、量が膨大な上、日々の更新が必要なことから、コンピュータによる図面管理が注目されている。設備図面の長所を以下に示す。

- (1) 地図(街路図)の上に設備を重ねて書く
 - (2) 設備は配線・配管(直線)と機器(シンボル)
 - (3) 清書図面(テンプレートを用いた手書き)
 - (4) シンボル・文字は、線上に書かれたり、他の線と接触していたりする。シンボル、文字を線画と分離するのが容易でない
- また設備図面認識では、設備を地図と分離して登録するため、次の機能が必要となる。
- (a) シンボルの種類と位置(角度)の認識
 - (b) 配線・配管の種類と位置の認識
 - (c) シンボル、配線・配管の間の接続の認識
 - (d) 文字を認識し、関連の設備と結びつける
- ここで紹介する手法は、(a)および(d)の一部の機能を果たす。自由な手書きシンボルの認識には適さないが、テンプレートを用いて書かれたシンボルや文字の認識には問題がない。また構造的な手法にない次の長所がある。
- (1) 認識速度が極めて速く、並列処理も可能
 - (2) 線上や線と接触したシンボル、一部かされたシンボルに対しても認識率が高い
 - (3) シンボルの位置も正確に求まる
 - (4) 対象図面が変わってもシンボルデータの変更がすみ、プログラム変更の必要がない

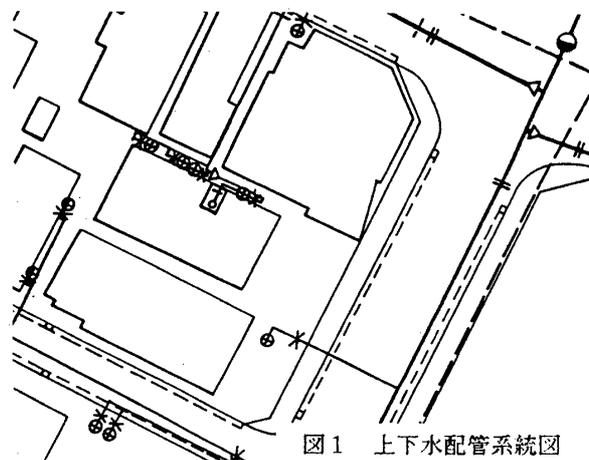


図1 上下水配管系統図

3. Raster演算によるシンボル候補の抽出と同定

3.1 画像間の論理演算によるシンボルの抽出

テンプレート上に基準点と数個のサンプル点(特長点)を選び、基準点からサンプル点までの距離をベクトル V_i ($i=1..n$)で表す。図2に示すように画像を $-V_i$ ずつずらして、それらの論理積(AND)をとれば(サンプル点がいずれも白なら白黒反転させた画像を用いる)、画像上でサンプル点と同じパターンつまりシンボルが存在する所は黒で残る。きれいな図面であれば4~6個のサンプル点をうまく選べば、シンボルの存在するその基準点の位置だけが黒(1)で残る。通常の1000x1000の画像であれば、数回の論理積をとるだけでシンボル候補点を数十から数百にしぼりこむことが出来る。

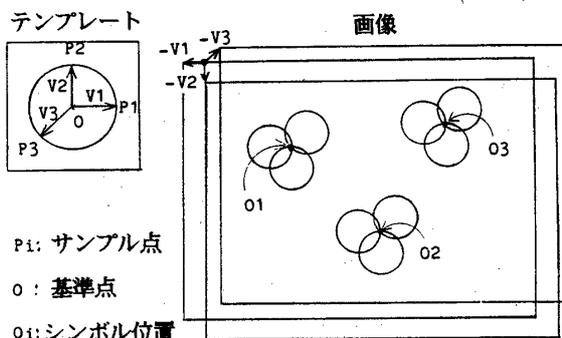


図2 画像間の論理積によるシンボルの抽出

Symbol Recognition with Raster Operation
Katsuyuki KAMEI, Yasuaki NAKAMURA, Shigeru ABE
Mitsubishi Electric Corporation

ワークステーションではマルチウインドウ表示を高速に行うため、bitmap画面の矩形領域に対応したメモリ上のデータと、同種の別データとの間で、bit単位に論理演算を行い、他のメモリ領域に書き込む専用ハードウェア（Raster演算装置）がある。上述の画像間の論理演算はこのRaster演算で極めて高速に（数Mbyte/sec）行うことが出来る。

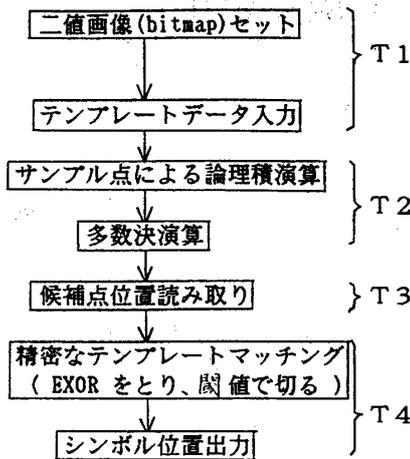


図3 フローチャート

3.2 ジョブの流れ (図3)

画像間の論理積だけでは、画像にとぎれや汚れがある場合シンボルをリジェクトする恐れがあるし、サンプル点だけのマッチングでは誤認の可能性も高い。リジェクトを防ぐため、数点の特長点の論理積結果を例えば5組用意し、その内3組以上が黒(1)であるなら、そこをシンボル候補点とするような多数決演算（これもRaster演算で行う）を導入する。

A組(a1, a2, a3, a4) [特長(サンプル)点]が ai(i=1..4)で $A = a1 \cdot a2 \cdot a3 \cdot a4$]
同様にB組, C組, D組, E組を決めると解は $S = AB(C+D+E) + C(A+B)(D+E) + DE(A+B+C)$

[一時記憶領域数個とRaster演算十数回必要] 誤認を防ぐため、しぼり込まれた各シンボル候補点において、精密な重み付き(マスク付き)テンプレートマッチングを行う。これもEXOR等のRaster演算とカウント演算ですむ。

3.3 サンプル点の選定

図面は白地の割合が大で、線幅は3~6dotが多いため、サンプル点は周りが同値の黒点を互いに十分離して(相関小)3点選ぶ(a1, a2, a3)。同様に対象図面特有のときれによってもA~Eの内3組以上が残るように黒点b1~e3を選ぶ。線上のシンボルも認識出来るように、白点(a4, b4, c4, d4, e4)はどの3点も同一直線上にないように選ぶ。

3.4 角度依存性シンボルの認識

円のような角度依存性のない部分を含むシンボルは、まずこれを利用して抽出した後、適当なサンプル点群の値から角度を決める。他の角度依存性シンボルは、角度を変えたテンプレートをいくつか用意するか、輪郭線追跡等構造的手法を併用する。(1)

1: 黒
0: 白
2: マスク
A, B, C, D, E:
サンプル点
多数決: 3/5組

図面	サイズ	認識	誤認	reject	時間
1	A3 x 2枚 12dot/mm	167 93 %	0 0 %	13 7 %	346 秒

図4 テンプレートと認識結果

図面	サイズ	シンボル	認識	誤認	reject	時間
2	A3 x 4枚 6 dot/mm	6種	633 98 %	7 1 %	15 2 %	22.6秒 /種.A3
3	A3 x 4枚 6 dot/mm	3種	479 99 %	3 1 %	3 1 %	43.5秒 /種.A3

図面2: きれいな図面 図面3: あまりきれいでない図面
図5 他の設備図面の認識結果

図面	シンボル	T1	T2	T3	T4	計(秒)
1	⊕	1.2	3.8	0.5	1.0	6.5
2	⊙	0.9	0.8	0.9	0.2	2.8
3	●	0.8	2.9	0.6	0.3	4.6

図6 処理時間 (1000 x 800 画像 by apollo DN560)

4. 実行例

図1の図面で⊕印を認識した時のサンプル点、認識率、計算時間を図4に示す。他の設備図面での認識結果を図5に、図3の各stepの処理時間を図6に示す。きれいな図面2は多数決演算が不必要なためT2が短い。

5. むすび

Raster演算装置は画像処理のための第二のALUであり、膨張・圧縮等の局所処理も高速に実行できる。本テンプレートマッチング法は、68020のような32bitマイクロプロセッサであれば、ソフトでRaster演算を代用しても十分早い。また印刷文字の認識、bitmap画面上のイメージサーチ等にも利用できる。