

動的領域割付けによる部分パターンからの文字，設計図面の合成†

李 寿 洙** 坂 井 利 之**

部分パターンの動的割付けによる高品質の文字，設計図面の自動合成について述べる。動的割付けとは部分パターンの部分領域の属性（大きさ，相対的位置等）を部分領域の境界における状況に基づいて制御することを意味する。動的割付けによる文字，図面の合成を実現する条件として，i) 標準部分パターンが適応的にサイズなどを変換できること，ii) 部分パターンの結合関係を動的に決定できること等があげられる。そこでわれわれは文字領域および図面領域を分割する方針として，i) トップダウン的領域割付けと，ii) ボトムアップ的適応調整の概念を導入するとともに，その評価基準を設けた。局所的図形である文字作成の例として，部分パターンの数が少なくストロークが比較的単純な韓国語（ハングル）と，部分パターンの数が多くてストロークが複雑な漢字という極端な両文字を選択し，また紙面全体にわたる図面作成の例として，回路図を選び，graphic symbol の出現位置とそれらの接続関係のみをデータとして自動合成を行った。その結果，文字の大きさ，ストロークの幅等の多様な属性をもつ文字が良品質で作成できることを示すとともに，品質のよい図面も合成して，部分パターンの動的割付けの概念の有効性を示した。

1. ま え が き

従来の部分パターンの合成による漢字の合成に関する研究^{1),2)}では，部分パターン間の合成基準を外部から入力するとか，各文字ごとにあらかじめ定義した基準に従わせるなど，いわば固定的合成が多い。さらに文字の品質から見ても書体の表現や線ストローク幅の制御等に対しても改良すべき点が多い。

そこで，われわれは文字や図面の作成において，あらかじめ用意した部分パターンの動的割付けによる文字や図面の自動合成を目標とした。さらに品質のよい文字および図面を合成するために，トップダウン的領域分割とボトムアップ的適応調整の概念を導入し，それに基づいて文字と図面の自動合成を試みた。ここで動的割付けとは，部首等の文字パターンや graphic symbol のような図面の部分パターンの存在領域を，それに外接する長方形（以下，部分領域と称す）で近似表現し，その部分領域と他の部分領域間の関係に基づいて，部分領域の大きさ，位置等を動的に決定することである。

部分領域の動的割付けによる文字と図面の合成にあたり採用した仮定および評価の基準項目は以下のとおりである。

a) 仮 定

1) 領域分割のための基準軸：一つの文字のなかには基準軸が存在し，それによって文字領域を部分領域の集合へ分割することができる。図面の場合にも部分領域を整列させる基準が存在し，それに基づき部分領域を決定できる。

2) 文字一体化のための適応調整：文字あるいは図面の合成で，部分領域の隣接関係や領域境界線付近の状況および評価基準により部分領域の拡大，縮小，シフト等の適応的な操作を施すことにより，まとまりがあり，バランスのとれた品質のよい一文字（図面）が作成できる。

b) 評価の基準項目

1) 文字作成の場合

i) それぞれの文字の構造とその部分領域（偏，沓など）

ii) 部分領域の境界線付近のストロークの多少や凹凸状況

iii) 部分領域内における黒ストローク，白領域の存在量

2) 図面作成の場合

i) graphic symbol 間の最小距離

ii) graphic symbol に挿入する文字ストリングの数と大きさ

iii) graphic symbol 間を結ぶルート（線）と，ルート（線）間の隔り

iv) graphic symbol および文字の許容大きさ
文字合成例として，部分パターンの数が少なくてス

† A Study on Composition of Characters and Diagram by their Subpattern Utilizing Dynamic Allocation of their Regions by SOOYOUN LEE and TOSHIYUKI SAKAI (Faculty of Engineering, Kyoto University).

** 京都大学工学部情報工学科

トロークが比較的単純なハングル文字と，部分パターンの数が多くてストロークが複雑な漢字の二つの両極端な言語の文字を選択した。図面の場合には手書き図面をパターン理解してそこに含まれている graphic symbol の位置とその接続関係が抽出されたと仮定し，それらを記号で入力して，2次元全紙面上で図面の整形・合成を行った。実例としては電子回路を採用した。

2. 標準部分パターン

文字および図面の合成に必要な部分パターンは，あらかじめシステム内部に登録されているものと想定する。それらを標準部分パターンとして，1章の a) で述べた動的割付けの仮定によって決定した部分領域の属性に基づいて標準部分パターンを変換した後，それらを用いて文字および図面を合成する。したがって標準部分パターンの品質は合成文字や図面の品質を決める重要な要素の一つであるので，次のように記述と変換の方針を定めた。

- ① 標準部分パターンの記述を容易なものにする。
 - ② 変換後も良品質の部分パターンとするため，その原形の標準部分パターンは，その形，大きさ，線の太さ等のみならずトポロジカルな性質をも考慮する。
- 一つの標準部分パターンは部分領域を定義する長方形のなかで記述する。そのおもな記述様式を略述すると以下のとおりである。

a) 幅をもつ線⁷⁾

線(L)は2点以上の点から成り，図1に示すようにおのおのの点の座標 (x_i, y_i) ，方向ベクトル(BとC)と線の幅で記述する。線の幅は書体により変わりうるが2点間のストローク(座標と幅)は次式で与えられる。

$$L(t) = P_i + (3t^2 - 2t^3) \cdot A_i + r \cdot t(1-t)^2 \cdot B_i - s \cdot t^2(1-t) \cdot C_i,$$

$$A_i = P_j - P_i, |A_i| = |B_i| = |C_i|, 0 \leq t \leq 1$$

$$\phi = (\theta + \phi)/2, m = (1 + |\cos \phi|) \sin \phi,$$

$$r = |2 \sin \phi / m|, s = |2 \sin \theta / m|$$

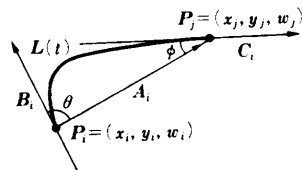


図1 補間曲線
Fig. 1 Interpolation curve.

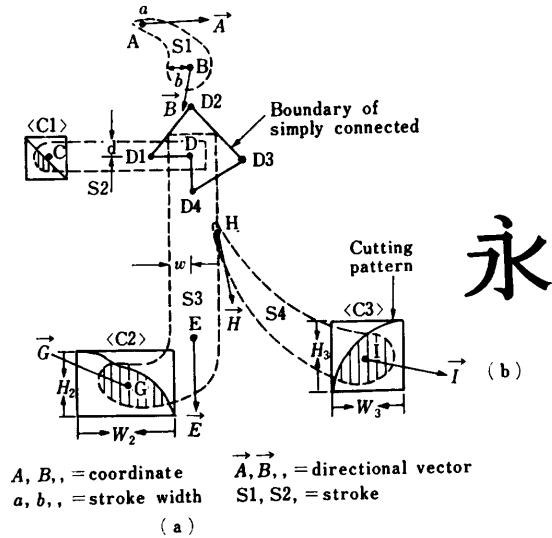


図2 標準部分パターンの記述例
Fig. 2 Description of subpattern.

b) 単連結領域

複雑な形状の領域を指定するため単連結領域はその境界を閉曲線として表現する。たとえば文字の屈曲点やうろこなどがこれに相当する。

c) カuttingパターン

線，領域の記述で表現しにくい部分パターン*を容易に記述するために採用した様式で，部分パターンの一部を削るために有効な手段である。この際，削る様子を記述した標準パターンがCuttingパターンである。

以上述べた様式に従って明朝体‘永’の一部の記述例を図2に示した。図2(a)のパターンは四つのストローク(S1, S2等)と三つのCuttingパターン(C1, C2等)と一つの単連結領域で構成されている。ストロークS1は2点の座標(それぞれAとB)から成り，A点でのストローク幅=a，方向ベクトル= \vec{A} ，B点でのストローク幅=b，方向ベクトル= \vec{B} で記述する。方向ベクトルの記述が省略されている場合には直線と見なす(例，S2)。Cuttingパターンは，そのパターンをかぶせる位置，そのパターンの種類，大きさを指定して記述する。たとえばC2 Cuttingパターンでは中心点の座標がG，幅= W_2 ，高さ= H_2 で，削る部分の指定は左下側の半分である。単連結領域D1の境界線は(D-D1-D2-D3-D4-D)のように6点の座標で記述する。図2(b)にこの記

*たとえば，閉領域がその中にホールをもつパターンや‘はね’等のような書体の表現のために必要なパターン等がある。

述により作成された結果を示した。

3. 部分パターンの領域割付けと適応調整

部分パターンによる文字の合成のために導入した動的領域割付けと適応調整の概念を、ハングル文字の自動合成への適用を具体例として述べる。

ハングル文字は表音文字であり、表1で示すようないくつかの音素の組合せによる単音節に対応している。すなわち一つの単音節は一つの文字に相当している。したがってここではハングル文字の基本要素である字素は、単音節を構成している要素である音素（子音、母音）と対応している。字素と音素の表現は表1にも示すようにとくに区分しないことにする。

ハングルの文字タイプは図3に示すとおり、6種類（以下、 T_i ; $i=1, 2, \dots, 6$ と表す）に分けられ、文字タイプによって字素の相対的位置などが決まるので、字素と対応する部分パターンによる自動合成が容易である。

3.1 合成および評価基準

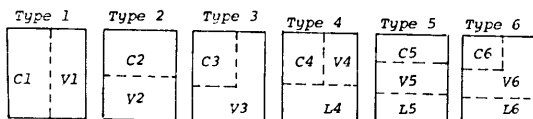
文字、字素の大きさは、それぞれの外接長方形の領域の大きさで表し、合成文字を字素の部分領域の大きさと相対位置を指定して記述する。

a) 基準軸と文字領域の大きさ

合成文字の品質の面からは、一つの文字領域を字素パターンの部分領域に分割するための基準軸の設定と、頁内での文字領域の相対的大きさの調和を保つことが大切である。そのために合成に際しては、文字領域からそれを構成する字素部分領域へのトップダウン

表1 ハングル文字の字素
Table 1 Element of Hangeol.

Phoneme C or V	Class	Sym- bol	Detailed kind	NR
Consonant	Simple	C ₁	ㄱ ㅋ ㆁ ㄷ ㅌ ㄴ ㄹ ㄷ ㅌ ㄴ ㄹ ㅅ ㅆ ㅈ ㅊ ㅊ ㅊ	14
	Double	C ₂	ㄱ ㅋ ㆁ ㄷ ㅌ ㄴ ㄹ ㅅ ㅆ ㅈ ㅊ ㅊ ㅊ	5
	Compound	C ₃	ㄱ ㅋ ㆁ ㄷ ㅌ ㄴ ㄹ ㅅ ㅆ ㅈ ㅊ ㅊ ㅊ	11
Vowel	Simple	V ₁	ㅏ ㅑ ㅓ ㅕ ㅗ ㅛ ㅜ ㅠ ㅡ	5
		V ₂	ㅓ ㅕ ㅗ ㅛ ㅜ ㅠ ㅡ	5
	Compound	V ₃	ㅓ ㅕ ㅗ ㅛ ㅜ ㅠ ㅡ	4
		V ₄	ㅓ ㅕ ㅗ ㅛ ㅜ ㅠ ㅡ	7



$C1=C2=C3=C4=C5=C6=\{Cs, Cd\}$ $L4=L5=L6=\{Cs, Cd, Cc\}$
 $V1=V4=\{Vv\}$ $V2=V5=\{Vh\}$ $V3=V6=\{Vc\}$

図3 ハングルの文字タイプとその境界

Fig. 3 Structure of Hangeol and its boundary.

的な領域割付けのアプローチを採用するとともに、次のような内容を組み込んだ。

1) 文字領域の大きさは文字の複雑さに比例させる。

2) 基準軸は文字タイプ別に選ぶ。すなわち、 T_1, T_3, T_4, T_6 のように垂直母音（表1の V_v ）を含む文字の基準軸は母音の最右端の垂直ストロークにし、 T_2, T_5 においては文字領域の中心線を基準軸とする。

b) 子音、母音領域の大きさ

字素パターンが子音あるいは母音を表す際のその部分領域をそれぞれ子音領域、母音領域と呼ぶことにする。

1) 母音領域の大きさは母音の複雑さに比例させる。

2) 子音領域の大きさはそれを含む文字タイプとその文字に含まれる母音によって決定し、つぎに子音領域の境界を、その境界線と平行に走る近傍の線ストロークの存在量に反比例するように拡大する。

c) 字素パターンの部分領域

上記の a), b) によって決まる部分領域を一体化し均整のとれた良品質の文字とするために、これらの部分領域を一部分互いに overlap させる、部分領域を接合 (trim) する、部分領域のシフト・拡大等の処理を施す。

1) overlap, trim させるための観測基準項目

i) 字素パターン領域の境界 (図3) において、その境界線をよぎる線素が両方とも点である。

ii) 片方が点で、他方には境界線の近くに concave タイプの線がある。

境界線近傍に平行なストロークがあるときは、overlap, trim の処理対象から遠ざける。

2) 部分領域の拡大、シフトの操作

字素パターンの部分領域間の境界付近で生ずる過大な白領域を判断し、部分領域の拡大によって白領域を縮小したり、部分領域のシフトによって白領域を変化させる。このような操作は、バラバラの字素パターンの単純合成による文字のまとまりの悪さを改善するものである。

3.2 領域分割アルゴリズム

まず一つの文字の全体領域を文字タイプにより、トップダウン的に構成要素の基本領域へ分割する。次に字素パターンの相互関係によって、基本割付けの領域をボトムアップ的に調整・変更する。前者は文字タイプによって決まるので主としてテーブルにより、後者

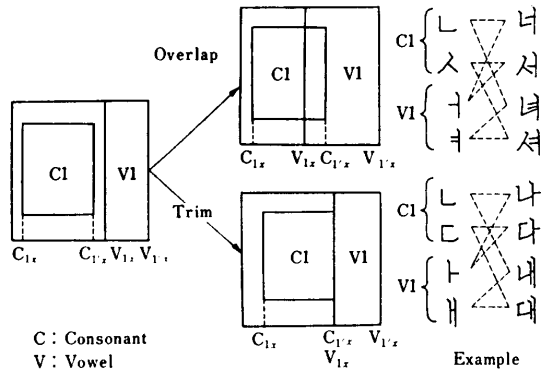


図4 Overlap と trim
Fig. 4 Overlap and trim process.

は後述のように図形パターンになるので計算によって領域調整を施す。

a) 基本領域の割付け

3.1 節で述べた方針を具体化するため次の3種類のテーブルを採用した。

1) RP テーブル (region partition table)

文字の基準軸に基づいた母音領域の割付けと，母音領域に依存した子音領域の割付けの基準になるテーブルである。文字の基準軸は文字領域（水平方向）の80% (T₁, T₃, T₄, T₆) の位置にした。

2) CW テーブル (consonant weight table)

子音領域の大きさの調整を文字タイプに関連させて制御するための子音の重み係数の表である。

3) CD テーブル (consonant direction table)

文字タイプ別に子音の拡大方向およびその拡大を制御するための方向別重み係数の表である。

b) 基本割付けの部分領域の変更

前述の overlap および trim 処理のための判断基準は文字タイプによって異なる。ここで文字タイプ T₁ の処理例を示す。

1) overlap 処理 (図4)

条件；

$$C1 = \{L, C, 人, ス, 天, 正\}$$

$$V1 = \{ナ, ニ, ヒ, フ\}$$

$$\text{処理； } V1_x \leftarrow C1_x + 2W/3$$

図3に示すように文字タイプ T₁ では，部分領域の境界は C1 領域の長方形の右辺と V1 領域の左辺の1カ所である。上記の“条件”で書いた字素がその境界線をよぎる線素は点または concave タイプになっている。図4で示したように，“処理”

として母音領域の左辺 V1_x が処理式に示すように子音領域の幅の2/3の位置に設定されるので，子音領域の右側の1/3が母音領域と overlap されている。

2) trim 処理 (図4)

条件；

$$C1 = \{L, C, 人, 正\}$$

$$V1 = \{ナ, ニ, ヒ, フ, \text{ } \}$$

$$\text{処理； } C1'_x \leftarrow V1_x$$

$$C1_x \leftarrow C1_x + (V1_x - C1'_x)/4$$

処理式の第1式で示すように，子音の右辺位置を母音の左辺位置と一致させ，子音の右辺の移動量の1/4程度，子音の左辺を子音領域の幅を縮めるようにシフトする。

領域分割アルゴリズムの概要を，図5 (a) にフローチャートで示した。同図での T_i は文字タイプ (図3) で，丸印の数字はステートを示している。文字タイプ T₁ で説明した overlap・trim 処理と同様に，すべての文字タイプについて処理ルールはそれぞれ一定で，その係数が定まっている。図5 (b) は文字タイプ T₆ の1文字の合成過程を示したもので，アルゴリズムの単純化のために，たとえば，図5 (a) の丸印のステート6での処理にはステート3での処理結果を縮小して利用した。

c) 自動合成の手順

1) ハングル文字を1文字ずつ，コードとして入力する。

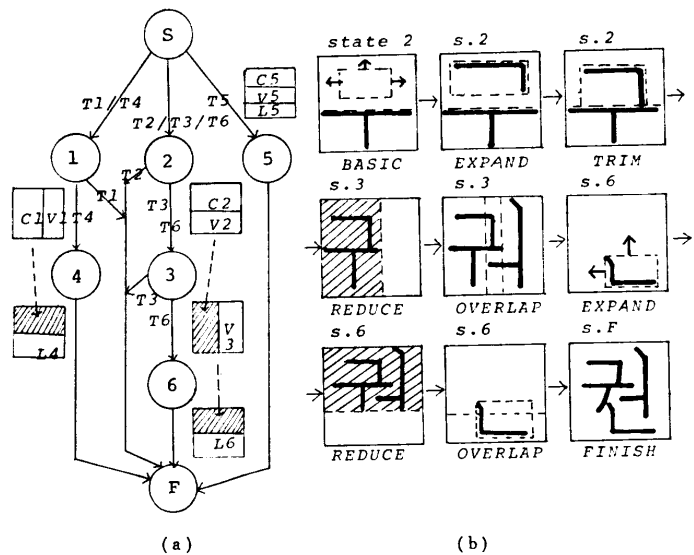


図5 領域分割アルゴリズムとその例
Fig. 5 Algorithm of region partition.

정보학과 의교육연
구현황과제경도학
정보전공사카이캉
미자와갈치전달콤
뮤니케이션의소푸

정보학과 의교육연구현황과제경도
학정보전공사카이캉콤뮤니케이션
의미자와갈치전달소푸터레이포
컴퓨터영어및아르헨티나는포크
런던영화협회실시간비밀정보담당
전중요정보설계실질금관대구모면
독문언어교육남은것강은대원창원
칠면조기업수강비학습생서통신자
연언어생물공

图 6 ハングル文字の作成例

Fig. 6 Example of Hangeol generation.

- 2) 文字タイプを判別し，動的領域割付けを行う。
- 3) 文字タイプに似合う字素パターンを選択する。
- 4) 標準部分パターンを変換して文字のドットパターンを作成する。

3.3 実験

ハングル文字の自動合成で採用した標準部分パターンは計55種で，表1から七つのV_vを除いた44種の基本字素と他の11種の字素から成っている。11種の字素は文字の品質を確保するために採用したものであって，文字タイプによって形が変わるものが含まれている。論理的に合成可能な文字(約12,000)*のすべての合成を試みた。合成された文字の品質の評価はハングル文字を知っている6人に依頼して行った。

文字の品質は，文字タイプ T₁ から T₆ まで，三つのテーブル，overlap・trim 処理の条件・処理式のルールに従って自動合成した全文字 12,369 字に対して，ほぼ満足できるものであった。合成文字の一部を图6に示した。

4. 漢字の合成

ハングル文字の合成においてその有効性が示された領域分割と適応調整の概念および評価基準を部首等の組合せによる漢字の合成へ適用しその有効性を検討した。漢字はいくつかの部首の階層構造で表現できる。漢字の構造は部首に基づいて偏，旁，かまえ等に分けられる⁶⁾。

部分領域相互間の面積割付け比率を容易にすることも考慮し，漢字を階層構造で表現して，部分領域をキーポイント(KP)部分パターンの部分領域(以下，KP領域と略す)とノンキーポイント(NK)部分パターンの部分領域(以下，NK領域と略す)に分けた。さらにKP領域を一つの漢字文字のなかでの位置，すなわち，最左側，最右側，最上部，最下部の位

*表1で文字タイプ T₁ は C₆ (14個) または C_d (5個) の一つと V_v (9個) との組合せで総数 19×9 となる。以下，同様で，正確には 12,369 個になる。

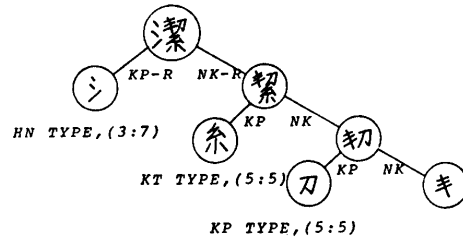


图 7 漢字の階層構造

Fig. 7 Hierarchical structure of Kanji.

表 2 KP 領域による漢字の分類

Table 2 Classification of Kanji by key-point.

漢字タイプ	領域比	キーポイント	例
ヘンタイプ	(3:7)	イシロ土日	依没吸境映
	(4:6)	オ女βイオ	投始限役狂
	(5:5)	糸小石糸木	維快破移械
ツクリ	(6:4)	多βリ寸力	形郊列封効
	(5:5)	欠爻頁爻隹	欧段願救維
カムリ	(3:7)	山サ々四竹	岩荷完罪符
	(4:6)	立主曲日田	竜青農界累
	(5:5)	宀雨西自立	究電栗皇辛
クツ	(6:4)	心皿大王白	忘盜吳望皆
	(5:5)	女木示衣糸	要栗粟製緊
	(4:6)	貝月日見	貨育肯覚

置に従って分類し，ヘンタイプ(HN)，ツクリタイプ(TK)，カムリタイプ(KM)，クツタイプ(KT)と称することにする。

KP領域の選択基準には①漢字の部首に類似のものを選定し，②上下左右から分離しやすいこと，③多くの漢字に共通していること等を基準にした。

图7に示すように漢字をKP領域とNK領域に分け，NK領域はさらにKP領域とNK領域に分けて，KP・NK領域の階層構造として表した。そして部分パターンの合成によって漢字を合成するにはKP・NK領域の階層構造を利用すること，さらにKP領域の属性(タイプ，KPとNK領域間の割付け比率)によるトップダウン的な領域分割とボトムアップ的な適応調整が有効であろうと考えた。

「基準明朝体の当用漢字」フォント⁹⁾を調査し，それを用いていくつかの漢字の合成を試みた。KP領域の属性により分類した漢字の一部を表2に示した。ただし分類の際，ハングルと類似性のあるもののみを考慮した。したがって漢字の固有の構造である垂れ，

電電電電	没没没没
背背背背	欧欧欧欧
没没没没	究究究究
欧欧欧欧	皆皆皆皆

(a)

(b)

電欧較究楽	境郊岩臭貨
始般望栗皆	投部累忘資
形段却没依	素効竜盗盲
段鏡荷青票	設刊育書背

(c)

図 8 漢字の作成例

Fig. 8 Example of Chinese character.

しゅう，かまえを含む漢字を除外している。

図 8 (a) は部分領域の割付け比率を変更しながら作成した合成漢字の例である。図 8 (b) は表 2 で示した KP 領域の属性を用いて領域分割を行い，KP 領域と NK 領域間の overlap と trim 比率を変化させて作成した例である。これらより 5~10% ぐらいの領域の overlap と trim 処理の適応調整が適当であることが明らかとなった。図 8 (c) は上述の領域分割と適応調整のルールを適用して自動合成した漢字の例である。

5. 設計図面の合成

以上で述べたデジタル的な文字の合成は，文字を一定の格子点上でのドットの集合として表現しており，格子の間隔は比較的小さい場合である。これに対して，回路図やケミカルプラント等の図面の整形・作成は graphic symbol (以下，GS と略す*) とこれを記述する文字ストリングを格子点によって位置ぎめし，GS 間の接続関係を線分によって表現 (以下，ルーティングと称す) することにより記述できる。この場合，格子の間隔は大きく，格子点間の距離も GS の大きさ等の図面の属性に依存して変化しうる。したがって図面の合成において，領域分割とは GS を配置するために格子**の大きさや格子間の距離を決めることであり，適応調整とは GS 間の相互関係や GS を記述する文字ストリングとの関係から GS の大きさ，文字の大きさ，ルーティングを状況に応じて動的に決定

* 電子回路ではダイオード，トランジスタ，コンデンサ等を意味する。

** 格子は縦横同じ間隔の正方形である。

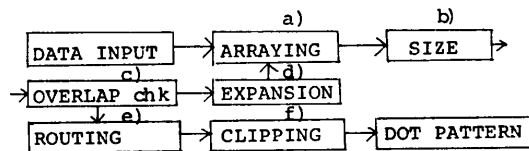


図 9 図面合成の流れ

Fig. 9 Outline of diagram composition.

することである。

そこでわれわれは手書き図面をパターン理解して，そこに含まれる GS の位置とその接続関係が抽出されたとし，それらを記号入力して図面の整形・作成を試みた。入力条件は次のとおりである。

①GS の条件: GS の数と種類，GS の中心位置，GS (大きさ) の決定に文字ストリングを配慮すること。

②ルーティングの条件: GS 間の関係を表すための各 GS の端子部分 (以下，ターミナルと称す) は GS に外接する長方形の辺の四つの方向 (図 10 (a)) で近似表現させるとともに，同一方向の辺に複数のターミナルがある場合にはそれらを長方形の各辺ごとに反時計方向に順序づける。したがって二つの GS 間の関係は二つのターミナル間の関係，すなわちそれが位置している辺の方向とその辺におけるターミナルの序数で表す。

文字合成において採用した部分パターンの外接長方形と同様に GS の外接長方形を図面の部分領域として取り扱い，トップダウンの領域分割とボトムアップの適応調整を施すことによって，局所的な文字合成と同様の概念と処理で全画面にわたる図形を良質に自動合成することを試みた⁴⁾。以下，図面合成のアルゴリズムについて述べる (図 9)。

a) GS の位置ぎめ

手書き図面のように不揃いである GS の中心位置を整列させる処理で，GS の中心位置を最も近い格子点にマッピングする。格子の大きさは，入力 GS の数と，GS をマッピングした格子点の数が同じであり，かつ格子点の間隔が最大のものを採用する。

b) GS および文字の大きさの決定

GS はまず標準大きさに置換してから，その GS に内包させる文字ストリングが多い場合には，その数に基づいて GS を拡大する。最大の大きさを越えるときは文字の大きさを縮小し，GS 内に文字ストリングが収容できるようにする。

c) overlap チェック

GS 間に overlap が生じているか否かをチェック

し、重なった場合は GS の中心位置を基準にして GS を縮小する。縮小に伴って GS に内包される文字の大きさも同様に縮小する。GS が最小の大きさに達するか、文字の大きさがある基準(例: 英数字では 9×9 画素)になった時点においても、なお GS 間に overlap があれば本処理は失敗とする。

d) 拡散

図面が局所的に複雑なため、c) の処理に失敗したときには格子の間隔を大きくし、したがって図面全体の面積も大きくなるが、GS の中心位置を拡散させる。拡散の許容範囲は図面全体の大きさに依存する。

e) ルーティング⁵⁾

GS のターミナル間をマンハッタン距離(ルート)を用いて連結する。図面の単純さを保つために、水平ラン(水平線分)と垂直ラン(垂直線分)の変化回数が少ないルーティングを最短距離のルーティングより優先させる。ルーティング処理は GS からの出、あるいは入りのターミナルの方向の組合せに基づいて行う。GS のターミナル方向を長方形の辺と同様な 4 方向に近似表現するので、二つのターミナル方向の組合せ数は 16 種になる。ターミナル方向を DX, DY と表し、その方向に基づいたルーティングを (DX, DY) と表すと (DX, DY) は次のように三つのクラスに分けられる。

①同一方向の組合せ: (1, 1), (2, 2), (3, 3), (4, 4).

②方向が 90 度異なる組合せ*: (1, 2), (2, 3), (3, 4), (4, 1).

③方向が 180 度異なる組合せ*: (1, 3), (2, 4).

さらに上記のクラスは、実は同一クラスの他の要素を 90 度の整数倍回転させたものにすぎない。したがってルーティングの処理は三つのクラスからそれぞれ代表として、一つの組合せを選んで行えばよい。GS のターミナルからのルーティング方向を図 10(a) のように定義する。例として同図(b)の GS1 の長方形で D2 方向のターミナルと他の GS の D1 方向のターミナル相互間のルーティング、(1, 2) のすべてを示すと図 10(b) のようになる。

二つのターミナル間のルーティング処理は、ターミナルの方向、相対的位置、他の GS との overlap の排除、すでに設定されたルートとの最小間隔の保持を考慮しながら状況に応じた問題解

決的な手法を採用して実行される。以下、GS1 の T5 ターミナル (D2 方向) と GS2 の D1 方向のターミナル間のルーティング処理を例として説明する。

1) 両ターミナルの方向と相対的位置により、(1, 2) の基本的ルーティング(図 10(b)において GS1 の T4 と GS3 のターミナル間のルート)が可能であるかどうかを判別する。

2) そこで GS1 の T5 から D2 方向上の適当な位置に仮想的なターミナル (VT) をもうける。この際、T5 と VT ターミナル間の間隔は、3) の処理が成立するまで単純に増加させていく。

3) VT と GS2 のターミナル間のルーティングを (1, 3) と見なして次のような (1, 3) ルーティング処理を施す。ただしターミナルの座標をそれぞれ、 (x_1, y_1) , (x_2, y_2) とする。

i) A 点(座標= (x_a, y_a)) と B 点((x_b, y_b)) の位置を次の ii) の条件が受容される状態でいったん決める。ただし、 $x_a = x_2$, $x_b = x_1$, $y_a = y_b = (y_2 - y_1)/2$ 。

ii) A, B 点をランの変化点とするルートが他の GS やすでに設定したルートと overlap していないことを調べる。ただし水平ランと垂直ランの垂直の交差のみは許される。

iii) もし overlap があると、A, B 点の位置を上、下に移動させながら、ii) の処理を反復する。もし A, B 点の $y_a (= y_b)$ 値が $y_1 < y_a < y_2$ の範囲内で overlap すると iv) の処理にはいる。さもなければルート(T5, B, A, GS2 のターミナル間で、ラン数

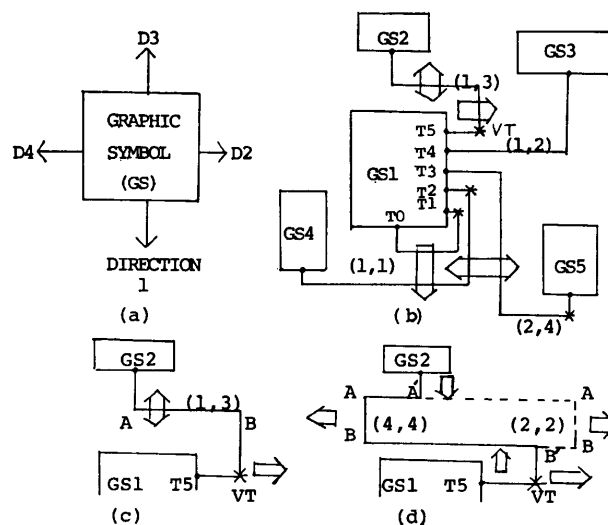


図 10 ルーティング (1, 2) の処理
Fig. 10 Process of routing (1, 2).

* ルーティング処理を簡単にするため、(DX, DY)=(DY, DX) (DX≠DY のとき) と考える。

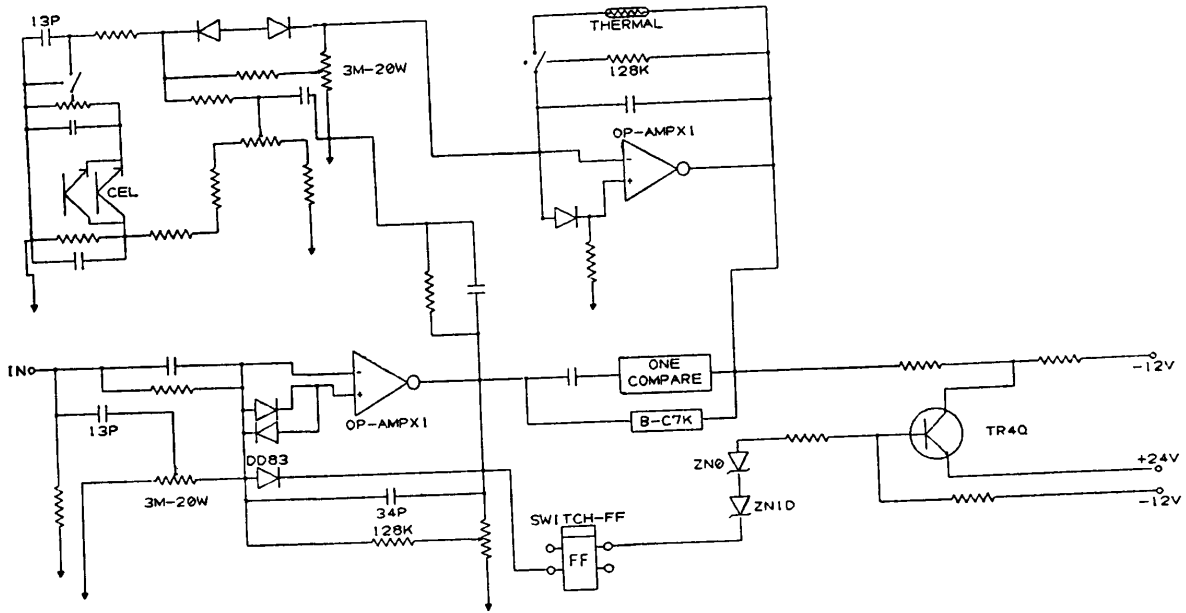


図 11 図面の作成例
Fig. 11 Example of diagram generation.

は 4) を決定する。

iv) A' 点を GS2 の D1 方向へ，B' 点を VT ターミナルの D3 方向へ位置ぎめする。A'，B' 点の位置は v) の処理が成立するまでそれぞれ D1，D3 方向へ移動させる。両点の垂直移動範囲は y_1 より大きく， y_2 より小さい。

v) A'，B' 点を新しいランの変化点と見なして，(2, 2) と (4, 4) のルーティング処理，すなわち図 10 (d) に示すように A，B 点とも A'，B' 点の左側または右側へ相互に移動させながら A' と B' 点間のルートを求める。この際，ルートは T5，VT，B'，B，A，A'，GS2 のターミナル間で形成される。

f) 分割出力

以上のアルゴリズムで決定した GS，文字ストリング，ルーティング結果に基づき，ドットパターンゼネレータにより図面の合成を行う。出力装置の分解能の制限からセグメント化して出力する。図 11 に電子回路図面合成の一例を示した。この図面は $960 \times 1,920$ 画素の図面に相当するものをそれぞれ重なりがある 512×512 の八つのセグメントに分けて出力したものををり合わせたものである。

6. むすび

部分パターンから動的領域割付けによる文字・図面の自動合成のために，トップダウン的領域割付けとボ

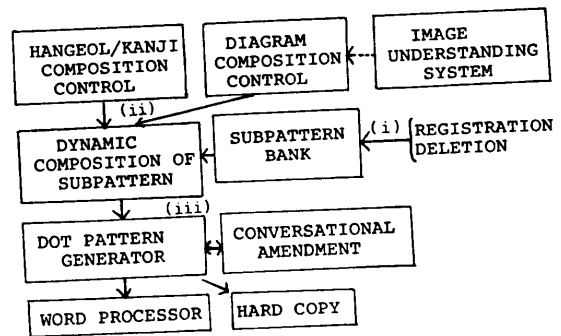


図 12 合成システム
Fig. 12 Composition system.

トムアップ的適応調整を併用した線図形作成手法を提案した。文字作成の例として部分パターンの数が少なくストロークが比較的単純であるハングル文字と部分パターンの数が多くてストロークが複雑な漢字の一部の自動合成を試みた。

その結果，文字の属性を自由に変更しながら高品質のハングル文字の自動合成 (12,000 余種) と漢字の中の同じタイプの文字作成により，文字作成において設定した仮定の妥当性と手法の有効性を示した。

文字の作成における領域割付けと適応調整の概念を拡大して，紙面全体にわたる図面の作成においては，手書き図面のパターン理解が完了した入力を仮定して電子回路の図面の整形・清書を試みた。その結果，graphic symbol や文字ストリングの部分領域の動的

割付け方法が図面の作成においても有効であることが明確になった。

ちなみに本システムは図 12 で示したように、(i) 標準部分パターンの管理部分、(ii) 部分パターンの動的組合せの制御部分、(iii) ドットパターンの生成部分等から構成され、ハードウェア構成はミニコン (64 KB) を中心に、ファクシミリ、タブレット、ディスプレイ、静電プリンタプロッタ等から成る小規模なものである。

今後の研究方向としては、漢字のタイプとその数を増やしての自動作成と、図面の種類を増やして、設計図面の理解、蓄積検索システムの汎用化を行う研究等がある。

参 考 文 献

- 1) 坂井利之: 部分パターンによる漢字の合成, 情

- 報処理, Vol. 10, No. 5, pp. 285-293 (1969).
 2) 大岸, 内藤, 渡辺: 汎用端末装置による漢字フォントの作成, 電子通信学会研究会資料, IE 79-4 (1979).
 3) 佐藤敬之輔: 日本字デザイン 1, pp. 58-77, 丸善, 東京 (1979).
 4) 李 寿淵, 坂井利之: 図表の自動組込み機能を持つ英文原稿清書システム, 信学論 (D), Vol. J65-D, No. 5, pp. 558-565 (1982).
 5) 林 孝雄: デジタル計算機の自動設計, p. 285-334, 産業図書, 東京 (1973).
 6) 日本電子工業振興協会: 日本語情報処理の標準化基礎調査, pp. 273-318, 東京 (1980).
 7) Knuth, D.E.: TAU EPSILON CHI-A System for Technical Text, in Knuth, D.E.: *TEX and METAFONT*, Digital Press and American Mathematical Society (1979).

(昭和57年9月9日受付)

(昭和58年1月17日採録)