



Title	特徴点抽出と直線近似による道路網地図のデータ圧縮法
Author(s)	青山, 達也; 北島, 秀夫; 白川, 智昭; 小川, 吉彦
Citation	北海道大學工學部研究報告, 164, 55-61
Issue Date	1993-05-28
Doc URL	http://hdl.handle.net/2115/42368
Type	bulletin (article)
File Information	164_55-62.pdf



[Instructions for use](#)

特徴点抽出と直線近似による道路網地図のデータ圧縮法

青山達也 北島秀夫
白川智昭 小川吉彦

(平成4年12月25日受理)

Roadmap Data Compression through Extraction of Characteristic Points and Linear Approximation

Tatsuya AOYAMA, Hideo KITAJIMA, Tomoaki SHIRAKAWA,
and Yoshihiko OGAWA

(Received December 25, 1992)

Abstract

This paper proposes an effective compression of roadmap data the volume of which is formidable if it is in an uncompressed, raw form.

The proposed method extracts features such as crossings, end points, and curves from a roadmap.

They are subsequently decomposed into position data and connectivity data which are compressed separately.

Simulation results are included to demonstrate that 1/70 compression is possible with this method.

1. ま え が き

近年、社会の要求の高度化、多様化に伴い、地図をコンピュータに入力し、各種のサービスを提供する地図情報システムに関する研究が盛んに行われている。ここで問題となるのは、地図に含まれる道路、建物、鉄道などの線図形のデータ量が膨大であるため、大容量の記憶装置が必要となることである。したがって地図情報システムを構築する場合、いかにしてこれらのデータを圧縮するかが問題である。

地図情報システムの具体的な応用として市街地のナビゲートシステムを考えた場合、地図に含まれる線図形の中でも道路情報は最も重要である。そこで本稿では道路網地図のデータ量を高い圧縮率で圧縮する手法を提案する。1/25000の地図から道路網地図を抽出する方法はすでに提案されており¹⁾²⁾、この方法によると道路の中心線からなる図形が道路網地図として抽出される。ここで提案する手法は、このようにして得られた道路網地図を入力として、圧縮するものである。目的をナビゲートシステムとすることにより道路網の位置はそれほど正確はでなくても道路の接続関係が保たれていればよいから、地図にある程度の近似を行うことができる。この性質を利用して圧縮率を高めることができる。

データ圧縮の手順としては、道路網から交差点、端点等の特徴点を抽出した後、道路に直線近似を行い曲がり角を検出する。そしてそれらの位置、接続関係のデータを作成し、これらのデー

タを2値画像に変換後、符号化することによって圧縮を行う。本稿の最後には、提案した方法を実際の道路網地図に適用した結果を示す。

2. データ作成方法

2.1 位置、接続データの作成

本稿では、始めから図1のような市街地の道路網地図を直接スキャナで2値画像として取り込

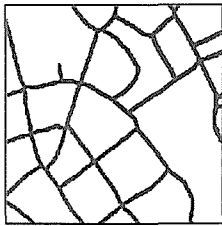


図1 入力する道路網地図

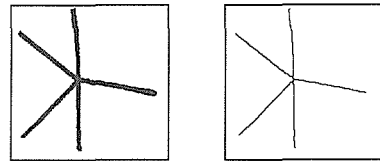


図2 細線化による交差点の分離

表1: 位置データ

点番号	座 標
1	(X_1, Y_1)
2	(X_2, Y_2)
3	(X_3, Y_3)
4	(X_4, Y_4)
⋮	⋮
⋮	⋮

表2: 接続データ

始点	近似点	終点
1	2,3	4
4	5	6
4	—	7
⋮	⋮	⋮
⋮	⋮	⋮
⋮	⋮	⋮

み、これを細線化することで文献²⁾のような道路網を擬似的に作成している(図9)。なお、以下では黒画素を1、白画素を0として話を進めていく。細線化の処理過程で、画像端はすべて0とする。位置、接続データを得るまでの処理手順を以下に示す。

- 交差点、端点等の道路のターミナルとなる点(以下、特徴点と呼ぶ)を抽出する。
- 各特徴点間を接続する道路に対して直線近似を行い、近似点(曲がり角)を抽出する。
- 細線化したときに図2のように4差路以上の交差点がいくつかの3差路等に分かれてしまう場合が生じる。そこで、ある一定距離以内にあるこれらの交差点を結合することによりデータの補正をする。この距離はデータを読み込む解像度に依存するが、詳しくは2.2で説明する。

この処理によって得られるデータを、各特徴点、近似点の座標とその点の座標番号から成る位置データ(表1)と、それらの点の接続関係を座標番号によって示した接続データ(表2)としてまとめる。

2.2 解像度

ここで目的としているナビゲートシステムのための地図には道路網だけではなく文字・記号も含まれる。本稿ではすでに文字・記号の部分が取り除かれた道路網のみの地図のデータ圧縮について検討するが、前処理として文字・記号の抽出を考えた場合、地図の読み取り密度としては、

文字・記号が認識できる程度以上であることが望ましい。ここではデータを読み込む解像度は 240 dots/inch として検討する。

1/25000 の地図では最も細い道路（幅 2.5～5.5 m）は 0.4 mm の平行線で表される。文献²⁾の方法による道路網に近づけるため、図 1 の道路網は約 0.4 mm の太さにした。この太さの線を 240 dots/inch で読み込むと 4 画素から 6 画素の太さとなる。よって細線化を行った後、2 つの交差点間の距離が 5 画素以内となったときは、もともと 1 つの交差点が処理の過程で 2 つに分離してしまったものであると見なすことができる。そこで、2.1 で説明した交差点結合距離は 5 に設定した。

2.3 道路の直線近似方法と近似の閾値について

直線近似については、いろいろな方法が提案されているが、ここでは距離を利用した方法³⁾を用いた。これは道路とその道路の端点を結んだ直線とを比較し、2 つの線が最も離れている箇所の距離があらかじめ定めておいた閾値よりも大きければその箇所で直線を 2 つに分割して、道路の線に近づけていく方法である。図 3 に処理の流れを示す。

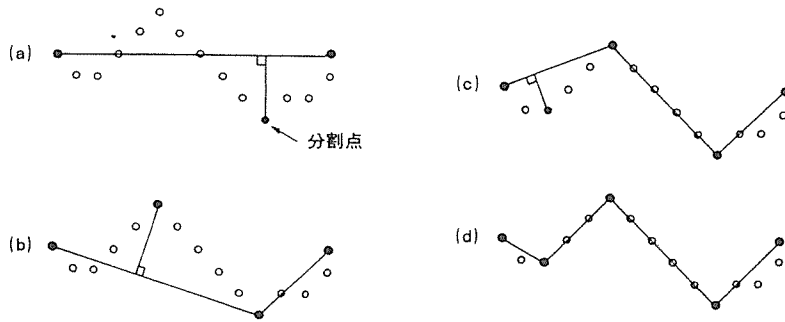


図3 近似処理

この近似を図 1 の細線化画像（図 9）に対して閾値 T を変えて行ったときの近似点の個数を調べたものが表 3 である。この結果を見ると、閾値 $T = 2$ のときは原画像との誤差はほとんどないが、近似点の個数はかなり多い。 $T = 3$ のときは原画像との誤差もそれほど小さくなく、近似点の個数も大幅に減る。他の何種類かの道路網地図に対しても同じような傾向がみられた。そこで近似点の数と原画像との誤差の程度から考慮して、閾値は $T = 3$ に設定した。

表3: 近似の閾値と近似点の数

閾値 T	近似点の数
2	33
3	15
4	8
5	6

2.4 各データからのビットマップ（2値画像）の作成

2.1 の処理で得られた位置データ、接続データから次のようにして 2 つのビットマップを作成する。

- 位置データ

原画像と同じサイズの画像を用意し、各特徴点、近似点の座標の画素を 1、それ以外の画素を 0 とする。（図 4）

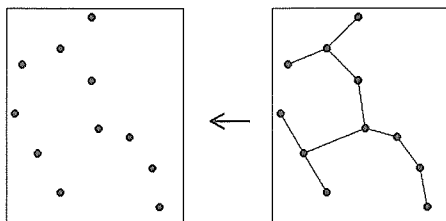


図4 位置データのビットマップ

・接続データ

まず要素がすべて0の $n \times n$ の行列を用意する。ただし、 n は位置データの個数とする。座標番号 x と座標番号 y の2つの点が接続している場合、

- ・ x 行 y 列を1 ($x > y$)
- ・ y 行 x 列を1 ($x < y$)

としていく。ただし、 $x = y$ のときは同一点を示すことになるので対角要素は全て0とする。このようにしてできた行列をそのままビットマップにする(図5)。

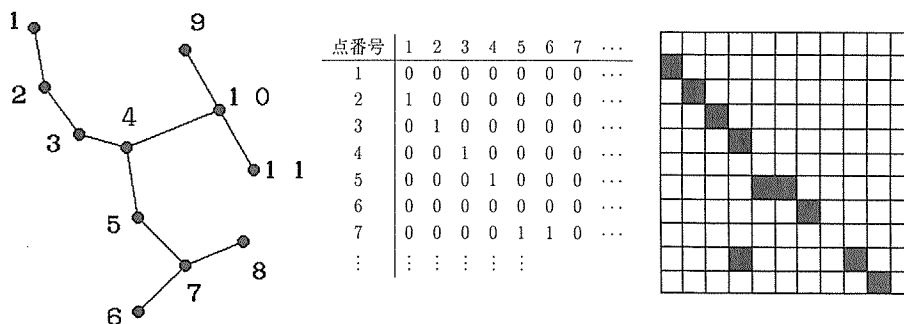


図5 接続データのビットマップ

これらの2つのビットマップを符号化することにより、道路網データの圧縮をはかる。

3. 各ビットマップの特徴と符号化

3.1 位置データの特徴

位置データから作成したビットマップには次のような特徴がある。

- ・道路網地図の読み取り範囲は有限であるため、読み取り範囲の周辺付近にある道路は途切れる。このため画像端(細線化の処理により、正確には画像端から1画素内側になる)に1の画素が多い。
- ・2.1, 2.2 で述べたように画素間距離5以内の交差点同士は結合してしまう。また画像端を除くと交差点と端点も画素間距離5以内にはほとんど存在しない。(もし存在すれば実際の道路の長さが10mもないことになるが、そのようなことはほとんどない)

これらの事を考えて、符号化の際は画像端と内側の部

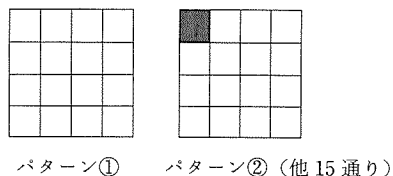


図6 ブロックのパターン

分に分けて、画像端はそのまま符号化、内側部分は4×4のブロックにまとめてから符号化を行う。これは内側部分では黒画素が互いに画素間距離で5以上離れているため、4×4ブロックで起こりえるパターンが図6で示す17通りしかないためビットマップの持つエントロピーが減少し、高い圧縮率が実現できるからである。

3.2 接続データの特徴

接続データのビットマップには次のような特徴がある。

- ビットマップの右上部分に黒画素はない。
- 長い道路がない場合、1つの道路の始点番号と、終点番号の差は小さいと考えられる。つまり、黒画素はビットマップの対角線に集まる傾向がある。ただし、この特徴は地図の形状や

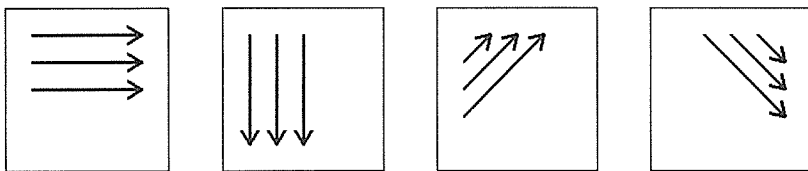


図7 座標番号のつけ方のパターン

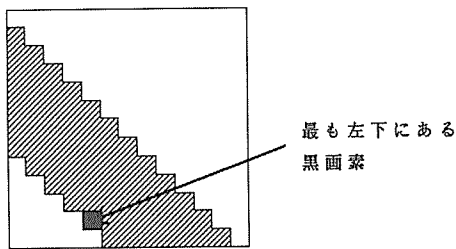


図8 接続データの符号化部分

位置データの座標番号のつけ方に大きく依存する。そこで図7のように4通りの座標番号のつけ方を用意して、その中からこの特徴を最も強めるものを選ぶことにする。

これらの特徴から、接続データのビットマップのうち、図8の斜線で示す部分だけを符号化すれば十分である。ただし、一番左下にある黒画素の位置を別に記憶しておかねばならない。

3.3 各ビットマップの符号化

両ビットマップとも白画素と黒画素の比率の差が大きいことから符号化にはMEL符号⁹⁾を用いた。MEL符号は2値画像の画素値0、1の出現確率の高い方を優勢シンボル、低い方を劣勢シンボルとして、データを各シンボルの出現状態によって1つの通報に区切り、符号語を与えるものである。接続データについては、このMEL符号をそのまま適用する。また、位置データについては図6の①を優勢シンボル、②の16パターンをまとめて劣勢シンボルとして、②については

表4: MEL符号の符号語例
(優勢シンボル0, 劣勢シンボル1の場合)

通報	符号語例	位置データの内側の符号化の場合、この符号語の後に4bitのブロックパターンを示す符号が必要
0000	0	
1	100	
01	101	
001	110	
0001	111	

表5: 図9に対するデータ量(bit)

	位置データ	接続データ	総データ
MEL符号	531,143	332	1006
算術符号	777	393	1170
チェーン符号	5747		

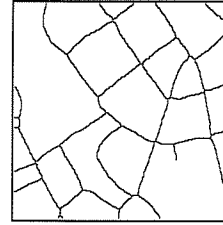


図9 細線化した道路網地図①

表6: 図9に対するデータ量(bit)

	位置データ	接続データ	総データ
MEL符号	324,133	196	653
算術符号	624	232	856
チェーン符号	5248		

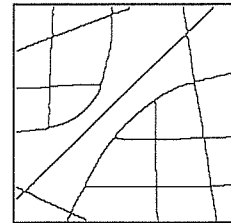


図10 細線化した道路網地図②

符号語の後にブロックパターンを示す番号をつけることにより、MEL符号を2値以上のデータにも適用できるように拡張する(表4)。また、これと比較するために算術符号とチェーン符号による符号化も行った。ただし、算術符号では位置データは内部をブロック化せずに2.4で作成したデータをそのまま符号化している。符号化を行った画像は、図1の細線化画像(図9)と、ある方向に長い道路が存在する道路網地図の細線化画像(図10)である。それぞれの画像のサイズは256×256画素、読み取り密度は先に述べた通り、240 dots/inchである。実験結果を表5,6に示す。なおMEL符号の欄の位置データは左が内側部分の符号量、右が画像端の符号量である。表より、提案した方法によりデータ量が70分の1程度に圧縮できることがわかる。線画の圧縮に有効とされているチェーン符号化を用いてもこれほどの高い圧縮率を得ることはできない。また、算術符号を用いた場合にも、データ量はMEL符号を用いた場合よりもわずかに増加する。したがって、提案した方法が道路網地図データ量の圧縮に非常に有効であると言えることができる。

4. ま と め

本稿では道路網地図から特徴点等の位置と接続関係を示す2枚の2値画像を構成し、それぞれの特徴を考慮にいたした符号化を行うことにより道路網データを圧縮する方法を提案した。また、実際の道路網地図に対して本手法を適用することにより、70分の1前後のデータ圧縮率が実現できることを示した。

本手法の利点は処理が容易であることと、作成したデータの特徴がつかみやすく、効率的な符号化が行えることである。したがって高速でありながら圧縮率の高いデータ圧縮を行うことが可能である。一方欠点としては、MEL符号による圧縮率が優勢シンボルと劣勢シンボルの比率に依存するため、圧縮率は交差点や曲がり角といった特徴点の個数にかなり依存することが挙げられる。しかし、いくつかの道路網地図を用いて実験を行った結果、地図の尺度、読み取り密度を変化させなければ概ね高い圧縮率を得ることができた。

参 考 文 献

- 1) 長尾, 安居院, 中嶋: 信学技報, PRU 87-35, (1987), pp.59~68
- 2) 長尾, 安居院, 中嶋: 信学論, Vol.J 72-D-II (1989), No.10, pp.1627~1634
- 3) 長谷川, 興水, 中山, 横井: 画像処理の基本技法, pp.79~80, 技術評論社
- 4) 加藤茂夫: インターフェース 12月号(1991), pp.152~154, CQ 出版社