

歩行者向けデフォルメ地図生成ハードウェアエンジンの設計

荒幡 明† 奈良 竜太† 戸川 望† 柳澤 政生† 大附 辰夫†

† 早稲田大学大学院基幹理工学研究科 情報学専攻

〒 169-8555 東京都新宿区大久保 3-4-1

Tel: 03-3209-3211(5775), Fax: 03-3208-7439

E-mail: †arahata@togawa.cs.waseda.ac.jp

あらまし 携帯電話の普及から多様なアプリケーションへの要求が高まっている。現在携帯端末へ地図情報を配信するサービスが普及しているが、それらの地図の多くは PC 用の地図であり、微細な携帯端末用ディスプレイでの表示には適していない。地図情報は性質上リアルタイムの更新を必要とするため、あらかじめ視認性の高いデフォルメ化された地図を作成しておくのは現実的ではない。そのため地図情報を自動的にデフォルメ化する技術が多数開発・提案されているが、デフォルメ化処理をサーバ上で処理するとデフォルメ化地図をユーザの嗜好に合わせるのは困難である。携帯電話の処理能力ではデフォルメ化処理は処理量が大きくレスポンスタイムや消費電力の増大を招く。本稿では携帯電話向けデフォルメ地図生成ハードウェアエンジンを提案する。携帯端末向けの画像処理という点に着目し、一つのデフォルメ化処理を基本に分析し、ボトルネックを抽出、適切なハードウェアを設計する。設計した演算器を携帯電話に組み込むことで、従来の 2 分の 1 から 5 分の 1 の処理量で実行できる。

キーワード デフォルメ地図, 携帯電話, 演算器, 平方根

A Hardware Engine for Generation Deformed Map

Akira ARAHATA†, Ryuta NARA†, Nozomu TOGAWA†, Masao YANAGISAWA†, and Tatsuo OHTSUKI†

† Dept. of Computer Science and Engineering, Waseda University

3-4-1 Okubo, Shinjuku, Tokyo 169-8555, Japan

Tel: +81-3-3209-3211(5775), Fax: +81-3-3208-7439

E-mail: †arahata@togawa.cs.waseda.ac.jp

Abstract An image of map information for a computer display is complex to be shown in a mobile phone LCD. thus a deformed map is necessary in a mobile phone. An image of map information needs a renewal on real-time processing. hence an automatic generation of a deformed map is proposed. An automatic generation of a deformed map on the network server is not favor of individual and on the network client costs mobile phone load. This paper presents a hardware engine for generating deformed map of a mobile phone. We analyzed generating deformed map and detected a bottleneck of the processing. As a result, we proposes appropriate ALU for a mobile phone. Embedding the proposing ALU for a mobile phone, It is possible to execute generating deformed map from 50% to 20% of the processing of the past.

Key words deformed map, mobile phone, ALU, square root

1. まえがき

今や日本における携帯電話加入者数は9千万人を越え、それに伴い携帯電話によるIP接続サービスの契約者数も8千万人を超え、また第三代携帯電話も契約者数が7千万人に迫っている。これによって今までよりも大きなデータの通信・処理が可能となり、テレビ電話などのマルチメディアサービスが普及し始めている。加えて携帯端末にGPS機能搭載が実現したことにより、位置情報から地図情報を取得するアプリケーションが活発に研究、開発されており、歩行者を対象とする携帯端末用地図情報配信システムが考案、開発されている[10]。

地図情報はこれまでラスターデータで配信されていたが、近年の携帯端末の高性能化により、点の座標やそれを結ぶ曲線の描画情報であるベクタデータの動的な画面描画が可能となったため、ベクタデータでの配信が目目されている。ベクタデータはディスプレイの解像度に依存せず、拡大・縮小しても視認性が損なわれない利点がある。特に地図情報に関してはW3Cで定義されているSVGというXML形式のベクタデータ形式が目目されており、今後その普及が広がっていくと考えられる[3], [11]。

しかし携帯端末に配信される地図情報は元々PC用であるため、携帯端末上の微小なディスプレイにおいて見辛いものとなっている。そのため、現在携帯端末向けのデフォルメ地図を自動生成する技術が開発・提案されてきた[1], [5], [6], [9], [12]。デフォルメ化処理はマップ全体をデフォルメ化するもの[5], [12]と、現在地から目的地までの経路だけをデフォルメ化するもの[1], [9]がある。この二つは、ノードのデフォルメ化処理に違いがみられるが、必要な情報の取捨選択や再配置などそれ以外の処理はほぼ共通しているものが多い。

地図情報そのものは更新されることが多く、リアルタイムにデフォルメ地図の作成が必要となる。一方、これまで提案されてきた携帯端末向けのデフォルメ地図はサーバ上で地図情報をデフォルメ化処理したものを配信することを想定しているが、それではユーザビリティが低く、ユーザの嗜好に合わせて頻繁に地図情報を編集することが困難である。携帯電話向けデフォルメ化処理では、クライアントとなる携帯電話自身でデフォルメ化処理を実現しユーザが望むデフォルメ化地図を自由に作成・表示することが望まれる。しかしながら携帯電話によるデフォルメ化処理は現在の携帯電話に搭載されているプロセッサのみを用いた処理能力では処理量が大きくレスポンスタイムの増大や電力消費の増大をまねく。携帯電話によるデフォルメ化処理を実現するには、デフォルメ化処理のステップ数を削減することが急務である。

そこで本稿では、携帯端末用デフォルメ地図生成ハードウェアエンジンを提案する。デフォルメ地図を自動生成する技術は多数提案・開発されているものの、これらは全てサーバ上での処理を仮定したものであり、現在は実用化の一手手前で実際の運用や実装まで考察が及んでいない。本稿では現在地から目的地までの経路デフォルメ化に注目しボトルネックを調査してハードウェア化を施す。文献[9]の手法に着目しこの処理を解析しボトルネックとなる処理を抽出した。解析したボトルネッ

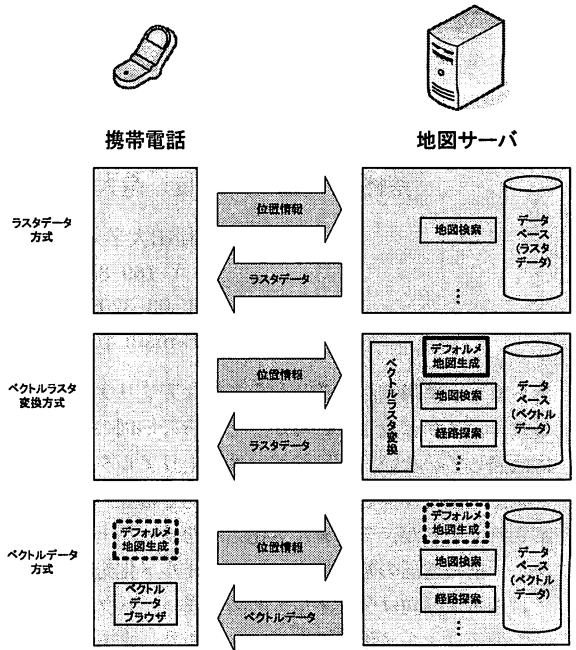


図1 地図情報配信システム概要図。

クおよび地図情報の特性から二つの演算器を提案し、これを実装することによってデフォルメ化処理の処理量が従来の2分の1から5分の1にまで抑えることが可能となった。

本稿は以下のように構成される。2章では地図情報配信システムと、デフォルメ地図の現状と概要、具体的な処理内容について説明する。3章では文献[9]の手法をハードウェア化するためにボトルネックを抽出し、そこに適したハードウェアエンジンを提案する。4章で実際に実装した結果から提案したハードウェアエンジンを検証する。

2. デフォルメ地図生成

現在、地図情報配信システムによる多くの歩行者向け地図情報サービスが提供されている。現在の地図情報配信システムと、今後の普及が考えられるデフォルメ地図生成について説明する。

2.1 地図情報配信システム [6], [7], [8], [13]

地図情報配信システムはGPS付携帯電話やユーザ入力によって位置情報を送信し、地図情報を扱うアプリケーションサービスプロバイダ(ASP)が受信しASP上のデータベースからその区画の地図情報を検索するシステムである。本稿では便宜上この地図情報を扱うASPを地図サーバと呼ぶ。検索した地図情報を携帯電話に配信する上で3つの方式が存在する(図1)。

ラスターデータ方式 地図サーバのデータベース上に地図情報をラスターデータとして保存し、検索したラスターデータをそのまま送信する。サーバの処理量は少ないが、座標計算が複雑、地図のデータ量が膨大、決められた縮尺の地図情報しかない等の欠点も持つ。

ベクトルラスター変換方式 データベース上に地図情報をベクト

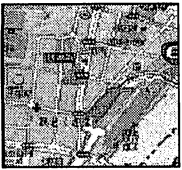
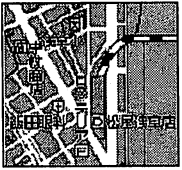
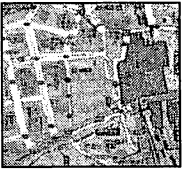
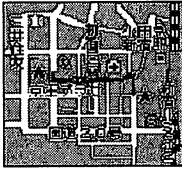
	オリジナルマップ	デフォルメマップ
浅草		
新宿		

図 2 地図全体デフォルメ化処理の具体例 [6].

ルデータとして保存し、サーバがラスターデータに変換し携帯電話に送信する。座標計算や縮尺の指定が容易であるが、ベクトルデータ変換のサーバ処理負荷が高い。

ベクトルデータ方式 近年の携帯電話の高性能化によって実現した。データベース上に地図情報をベクトルデータとして保存し、携帯電話にベクトルデータとして送信する。携帯電話側の視認性を損なわない縮小・拡大・回転、レイヤ合成が容易で、送信するデータ量も小さい。ベクトルデータの形式が統一化しておらず今後の標準化が期待されており、その中で SVG 形式が最も有力とされている。

現状ではベクトルデータ変換方式が最も普及しているが [7], [8], [13], 今後は視認性とデータ量に優れた携帯電話側の編集も容易なベクトルデータ方式に移行していくことが予想される [4]。ベクトルデータ方式では、データベース上のベクトルデータの地図情報は地理情報標準プロファイル (JPGIS) によって標準化されメタデータとして保持されている [3]。地図サーバはこのメタデータから必要な地図情報を取捨選択し、経路探索を行い、必要があれば地図サーバが外部のコンテンツサーバにアクセスし更に情報を付け加える。このベクトルデータの地図情報が SVG などに変換されて配信される。

デフォルメ地図生成処理は、現状では地図サーバ上でのデフォルメ化処理が想定される。地図情報配信システムは保持されている地図情報の書き換えが頻繁に起こるため、デフォルメ地図を予め作成することは出来ずリアルタイム処理しなければならない。デフォルメ化処理をユーザの嗜好に合わせて頻繁に編集する場合、サーバ上でのデフォルメ地図生成は何度もサーバにアクセスし地図情報を取得しなおすので、サーバ負荷の増大、携帯電話の通信処理の増大を招く問題を抱えている。携帯電話で処理する場合、一度地図情報を受信した後はユーザの嗜好に合わせて頻繁な編集が容易となるので、デフォルメ化処理に適している。

2.2 デフォルメ地図概要

携帯電話の微細なディスプレイ向けに視認性の高いデフォル

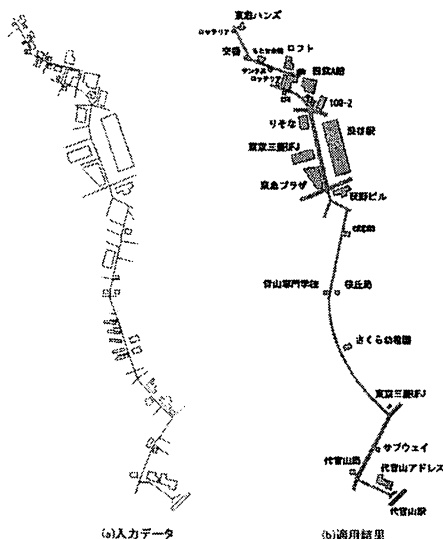


図 3 目的地までの経路単体デフォルメ化処理の具体例 [9].

メ地図が求められている。デフォルメ化とは、既存の地図情報 (道路・ランドマーク等) から必要な地図情報だけを抜き出し、ユーザにとって分かりやすいように地図情報を変形・再構成する処理である。デフォルメ地図には 2 種類あり、地図情報から区画を切り取りその区画全体をデフォルメ化するもの (図 2) と、通常の地図情報から必要な目的地までの道案内地図情報だけを抜き出してデフォルメ化したもの (図 3) がある。

デフォルメ地図の善し悪しはユーザの主観的な評価であるため一概にはいえないが、一般的に携帯電話の地図情報配信システムは道案内を第一の目的とし、ユーザがまず道を間違えない情報の提示が重要である。多くのデフォルメ地図自動生成の研究は、地図情報のうち道路形状を直線化・直交化させ画面に対して水平・垂直化させる処理であり、デフォルメ地図自体は単純で見やすいが道案内について考慮されていない。一方文献 [9] の地図生成手法は、歩行者が曲がり角が一つある時の方向判断基準、および交差点形状が与える心理的影響を加味したデフォルメ化処理を提案している。

そこで本稿では、実際に歩行者が街を歩くときの認知科学的基準を反映させた文献 [9] のデフォルメ地図生成手法に注目する。

2.3 処理内容

文献 [9] のデフォルメ地図生成手法を紹介する。このデフォルメ化処理は以下の工程から成る。

- (1) 前処理
- (2) 経路ノードデフォルメ化処理
- (3) ランドマーク選択・再配置処理
- (4) 後処理

サーバのデータベース上から地図情報を検索し経路探索をしたベクトルデータの地図情報が入力として与えられる。前処理でベクトルデータの地図情報から経路ノード・ランドマークの座標や形状等、地図を編集し再構成する上で必要な情報を抽出す

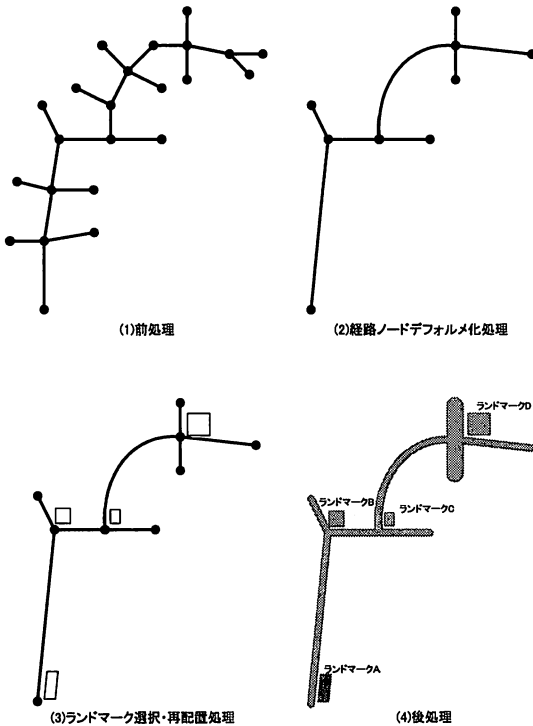


図4 デフォルメ地図生成手法概略図.

る。経路ノードデフォルメ化処理で、歩行者の認知科学的基準を反映させて経路ノードを合体・直線化・曲線化させ、不必要な小さい側道は省略する。デフォルメ化された経路ノードは元の地図情報と座標が異なるため、デフォルメ化された経路ノードに合わせて、道案内に重要なランドマークを選択し再配置する。最後にデフォルメ化された経路ノードとランドマークに色や文字等の付加情報をつけて、ベクトルデータのデフォルメ地図として出力する。

3. デフォルメ地図生成ハードウェアエンジンの提案

デフォルメ地図生成処理は、ユーザの嗜好に合わせるためにサーバではなく携帯電話でデフォルメ化処理させることが必要不可欠である。しかしながら携帯電話によるデフォルメ化処理は現在の携帯電話の処理能力では処理量が大きくレスポンスタイムの増大や電力消費の増大をまねく。本稿では本節 [9] の手法に注目し、ボトルネックとなる処理を解析した後、適切なデフォルメ地図生成ハードウェアエンジンを提案してデフォルメ化処理の削減を図る。

3.1 デフォルメ化処理の解析

文献 [9] のデフォルメ地図生成の処理をアセンブリコードに変換し^(注1)総ステップ数の見積もりを行った。デフォルメ化処理に必要なステップ数を、経路ノードデフォルメ化処理、ラン

表1 各パターンのステップ数。[万ステップ]

	経路の デフォルメ化	ランドマークの 再配置	その他	合計
A	54	1526	117	1697
B	27	510	112	649
C	13	180	110	303

ドマーク選択・再配置処理、前処理と後処理を合わせたその他に分け表1に示す。経路A、経路B、経路Cの経路ノードとランドマークの数に差がある3つのデータを対象とした。

表1から前処理と後処理はほぼ一定のステップ数を必要とし、経路のデフォルメ化とランドマークの選択・再配置は経路ノードとランドマークの数に依存することが分かった。特にランドマークの選択・再配置に多くのステップ数を必要としていることが分かる。これは経路のデフォルメ化によってずれた経路ノードの座標に合わせて道案内に重要な、目立つ、もしくは曲がり角で目印となるランドマークを、歩行者が誤解しないように検証しながら繰り返し選択・再配置していくためである。このランドマークの選択・再配置における処理の80%~90%は経路ノードとランドマーク間の距離Dの算出、つまり

$$D = \sqrt{x^2 + y^2} \quad (1)$$

の計算に費やされている。よって式(1)を計算する二乗器と平方根器を設計する。

3.2 仕様

デフォルメ地図生成のハードウェアエンジンは携帯電話用であるため、小規模かつ高速であることが求められる。また経路ノードとランドマーク間の距離算出、及び地図情報処理の特性上、以下の3点に即して設計を行う。

- (1) 地図情報処理は画像処理のため、精度において問題のない小数点以下を切り捨てる。
- (2) 地図情報の座標値を扱えるだけのビット数があれば十分なのでそれ以上の上位ビットは切り捨てる。
- (3) 小さい値ほど誤差を少なくし、なるべく低面積で早く地図情報処理に十分なだけの精度を確保する。

(1)について、現在の携帯電話はQVGAサイズ(320×240ピクセル)が一般的であるので、小数点精度のデータを扱っても画面上には反映されない。(2)について、携帯電話の地図情報の座標は範囲が限られているためある数値以上は計算しない。本稿ではこの数値を12ビットとして設計した。これは図3を解析した結果による。(3)について、経路ノードとランドマーク間の距離は道路と建物の距離なので、道路から距離の遠い建物を画面表示する機会はデフォルメ地図では少なく、また道路から遠い建物は距離に誤差があっても道案内では問題とならない。つまり出力結果が小さい数値ほどデフォルメ地図に反映されまた画面上にその誤差が現れやすいのに対し、出力結果が大きい数値はデフォルメ地図に反映されにくくまた画面上に誤差が現れにくい。ここである一定値Kを境界にK以下の出力数値を小さい値、K以上の出力数値を大きい値とする。K以下の出力数値は実際の演算値と演算器を用いた出力値との誤差を少

(注1): コンパイラにGCC(the GNU Compiler Collection)のバージョン3.3.5を用い、i486向けアセンブリコードに変換した。

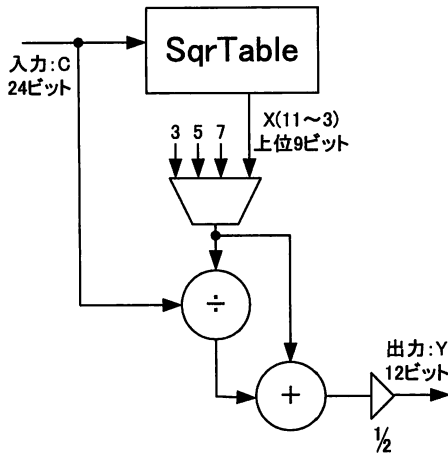


図 5 平方根器実装方式.

なく、 K 以上の出力数値は誤差が大きいかわりに低面積で実装させる。デフォルメ地図生成で街中のサンプルプログラムをシミュレーションした所、最終的に画面出力される経路ノードとランドマーク間の距離は全て 256 以下となったので、本稿では一定値 K を 256 とおく。

これにより処理速度と回路規模の縮小が可能だが計算結果に誤差が生じる。この誤差が携帯電話のディスプレイに表示した時に無視できるかどうかは最後に検証する。

3.3 二乗器

12 ビット入力、24 ビット出力の二乗器とし、アルゴリズムは Wallace ツリー [2] を導入した。Wallce ツリーは加算のキャリ伝達時間排除を目的としたアルゴリズムで、「キャリは順次上位へ伝播するもの」という考えを捨てて、「サム(和)は自けたの次処理への加算項、キャリは一つ上位桁の次処理への加算項」と考え、サムとキャリの区別をつけず計算する手法である。

3.4 平方根器

24 ビット入力、12 ビット出力の平方根演算器とし概略図を図 5 に示す。平方根演算器はテーブルと、テーブルの出力値を補間するニュートン法演算部によって実装した。テーブルに全ての入力値に対応する平方根演算値を格納しておくことは出来ない。実際の値とテーブルから出力される値には誤差が生じる。そのためニュートン法による収束演算を行い実際の値に近づけるように補間する。

ここで図 5 のとおり入力値 C 、テーブル出力値 X 、平方根器出力値 Y とおく。テーブル出力値 X は、入力値 C に対して $8 \leq X \leq 256$ のとき 8 刻みで、 $512 \leq X \leq 2048$ のとき 256 刻みで、加えて $X = 4096$ となるように格納されている(表 3、表 4)。テーブル出力値 X の下位 3 ビットは 0 なので上位 9 ビットが出力される。ただし入力値 C が 63 以下の時は、3、5、7 がニュートン法演算部に出力される(表 2)。3、5、7 は 63 以下の平方根をニュートン法で求めやすいように実験的に算出した値である。

入力値 C が 0 以上 63 以下のときニュートン法演算部には、3、

表 2 入力値対応表 (3, 5, 7).

入力値 C	出力値
0~15	3
16~32	5
33~63	7

表 3 SqrTable 入出力対応表 (8~256).

入力値 C	出力値 X
$8^2 \leq C \leq 16^2 - 1$	8
$16^2 \leq C \leq 24^2 - 1$	16
⋮	⋮
$256^2 \leq C \leq 264^2 - 1$	256

表 4 SqrTable 入出力対応表 (512~4096).

入力値 C	出力値 X
$264^2 \leq C \leq 512^2$	512
$512^2 + 1 \leq C \leq 768^2$	768
⋮	⋮
$1792^2 + 1 \leq C \leq 2048^2$	2048
$2048^2 + 1 \leq C$	4096

5, 7 のいずれかが出力され、ニュートン法演算部を用いて補間される最終的な誤差は 0 か 1 となる。誤差が生じない場合が 93%、誤差が 1 生じる場合が 7% である。入力値 C が 8^2 以上 $264^2 - 1$ 以下のときテーブルは 8 以上 256 以下の値 X を出力し、実際の平方根値とテーブル出力値 X には誤差が 0 以上 7 以下生じる。この誤差をニュートン法演算部を一回用いて補間させた値と実際の値との誤差は 0 以上 2 以下である。誤差が生じず 0 の場合が 94%、1 生じる場合が 6%、2 生じる場合が 0.2% となる。入力が 264^2 以上のときテーブルから 512~4096 の値が出力される。512 以上 4096 以下の出力値 X が均等に分布して出力されたと仮定し誤差の平均値を算出すると 5~20 程度となる。

4. 実装結果

提案した二乗器と平方根器をハードウェア記述言語 VHDL で設計し、Synopsys 社の DesignCompiler W-2004.12-SP2 を用いて論理合成を行った。セルライブラリには STARC^(注2) (CMOS 90[nm]) の設計ルールを用い、制約なしでの測定した結果を表 5 に示す。この演算器を用いたとき、二乗演算および平方根演算に必要なステップ数はそれぞれ 1 ステップとなり、再びデフォルメ化処理に何ステップかかるか算出した結果が表 6 である。全体で 2 分の 1 から 5 分の 1 の処理量でデフォルメ化が終了している。

演算器設計の際に、小数点以下切り捨て、上位桁切り捨て、一定値以上の精度は低くという 3 点に即して設計を行った。そ

(注2) : STARC[90nm] ライブラリは東京大学大規模集積システム設計教育研究センターを通し、株式会社半導体理工学研究所(STARC)と株式会社先端 SoC 基盤技術開発(ASPLA)の協力で開発されたものである。

表 5 最大遅延時間及び面積.

	最大遅延時間 [ns]	面積 [μm^2]
二乗器	1.74	1451
平方根器	2.69	5593

表 6 適用前後のステップ数. [万ステップ]

	ランドマークの再配置 (前 → 後)	適用前の合計 (前 → 後)	全体の処理速度 [倍]
A	1526 → 162	1697 → 334	5.1
B	510 → 54	649 → 193	3.4
C	180 → 26	303 → 149	2.0

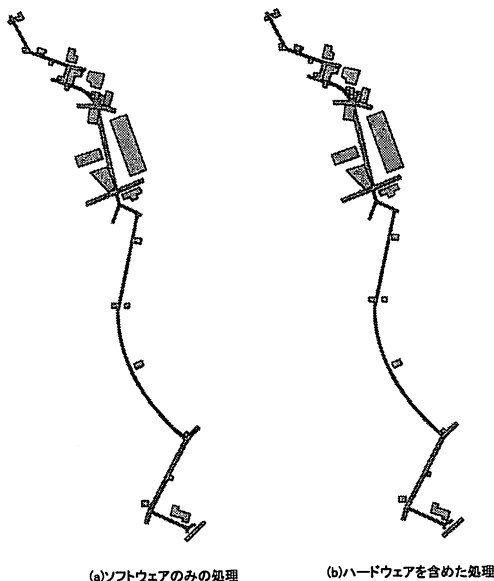


図 6 通常処理結果とハードウェア処理想定結果の描画誤差の比較.

のため設計した演算器を使うことによって誤差が生じてしまう。誤差を検証するために図 6 に通常のデフォルメ化処理 (a) をした場合と、この演算器を用いた場合を想定したデフォルメ化処理 (b) の例を示す。QVGA サイズより高い解像度を持つパソコンの UXGA サイズ (1600 × 1200 ピクセル) ディスプレイによって表示させた結果で、UXGA サイズで両者を見比べても目に付く誤差は発見出来ない。演算器による誤差は携帯電話上の微細な QVGA サイズでは十分無視できる程だということが分かる。

5. む す び

現状の地図情報配信システムとデフォルメ地図生成処理について説明し、文献 [9] のデフォルメ化手法を取り上げて紹介した。この手法を携帯電話上で実現する上でボトルネックとなる処理を抽出した結果、二つの演算器の実装を提案し実装した場合を見積もった。この演算器を組み込むことで従来の処理量の 2 分の 1 から 5 分の 1 で実行することが可能となる。

今後はハードウェアエンジンを組み込みデフォルメ化処理全体を扱うシステムを構築する必要がある。

文 献

- [1] 馬場口登, 堀江政彦, 上田俊弘, 淡誠一郎, 北橋忠宏, “経路理解支援のための略地図とその案内文の生成システム,” 電子情報通信学会論文誌, vol.J80-D-2 no.3, pp.791-800, 1997.
- [2] C. S. Wallace, “A suggestion for a fast multiplier,” in *Proceedings of IEEE Transactions on Electronic Computers*, vol.EC13-Issue no.1, pp.14-17, 1964.
- [3] 国土交通省ホームページ, <http://www.mlit.go.jp/>
- [4] 小林亜令, “The graphics information sharing platform for mobile computing based on SVG,” KDDI 研究所, 2005.
- [5] 丸山貴志子, 谷崎正明, 嶋田茂, “デフォルメマップ生成のための道路形状正規化モデル,” 電子情報通信学会論文誌, vol.J87-A no.1, pp.108-109, 2004.
- [6] 丸山貴志子, 谷崎正明, 嶋田茂, 伏木匠, “交通情報のデフォルメ地図へのマッピング方式とモバイル交通情報提供システムへの適用,” <http://www.ipsj.or.jp/10jigyo/fit/fit2004/fit2004program/html/event/pdf/17maruyama.pdf>, 2004.
- [7] MapFan ホームページ, <http://www.mapfan.com/>
- [8] NAVITIME ホームページ, <http://www.navitime.co.jp/>
- [9] 二宮直也, “歩行者ナビゲーションにおける微小画面での視認性とユーザの迷いにくさを考慮した略地図生成手法 (交通における計測・一般),” 電子情報通信学会技術研究報告, ITS2006-34, 2006.
- [10] 総務省ホームページ, “情報通信白書平成 19 年版,” <http://www.soumu.go.jp/>
- [11] W3C, <http://www.w3.org/Graphics/SVG/>
- [12] 山守一徳, 本田宏, 長谷川純一, “ストリート単位の変形に基づく道路網の整形手法,” 情報処理学会論文誌, vol.J84-D2, no.9, pp.2058-2069, 2001.
- [13] ZENRIN ホームページ, <http://www.zenrin.co.jp/>