

Focus+Glue+Context マップにおける Focus 変形・結合・分裂手法

水谷 祐弥[†] 山本 大介^{††} 高橋 直久^{††}

[†] 名古屋工業大学工学部情報工学科 〒466-8555 名古屋市昭和区御器所町

^{††} 名古屋工業大学大学院工学研究科情報工学専攻 〒466-8555 名古屋市昭和区御器所町

E-mail: †mizutani@moss.elcom.nitech.ac.jp, ††{yamamoto.daisuke,naohisa}@nitech.ac.jp

あらまし Focus+Glue+Context マップ Emma は、拡大領域である Focus，周辺領域である Context，それらの歪みを吸収する Glue からなる．既存の Emma には、複数の Focus が重なって表示させる場合に道路の繋がりが正しく描画できない問題がある．そこで、本稿では、複数の Focus が近接した際、まるで水滴が表面張力により結合するかのようになり、自然な形で Focus が変形し結合する手法を提案することによりこの問題に対処した．

キーワード Web マップサービス，Focus+Glue+Context，Emma

Transformation, union, and division methods of Focuses in Focus+Glue+Context map

Hiroya MIZUTANI[†], Daisuke YAMAMOTO^{††}, and Naohisa TAKAHASHI^{††}

[†] Department of Computer Science and Engineering, Nagoya Institute of Technology

Gokiso, Showa, Nagoya, Aichi, 466-8555 Japan

^{††} College of Science and Technology, Nagoya Institute of Technology

Gokiso, Showa, Nagoya, Aichi, 466-8555 Japan

E-mail: †mizutani@moss.elcom.nitech.ac.jp, ††{yamamoto.daisuke,naohisa}@nitech.ac.jp

1. はじめに

Google Maps [1] をはじめとするウェブデジタルマップにおいて利用者が目的地域を適切に表示するためには、スクロールや縮尺変更を繰り返す必要がある．さらに目的地域と他の離れた場所を比較したい場合には、広域地図を用いて全体的な位置関係を把握しつつ、詳細地図を用いて適宜詳細な地図情報を閲覧する必要がある．そこで、この問題を解消するために、我々は、これまでに Focus+Glue+Context マップ Emma [2] [3] を提案した．Emma とは、注目する地域を拡大して表示する領域である Focus，周辺部を表示する領域である Context，周辺部と拡大部の歪みを吸収する領域である Glue からなる地図システムである．Emma では、複数の目的地域を探索するために、複数の Focus を配置し、それらの位置と大きさと縮尺を個別に操作することができる．

複数の Focus を操作した際、それらの Focus 同士が重なって Focus の一部が隠れてしまう場合がある．既存の Emma において Focus 同士が重なる場合に、偏り Focus 重なり防止機能によってこの問題の解決を試みた．偏り Focus 重なり防止機能とは、ユーザが操作している Focus が他の Focus と重なった場合

に、ユーザが操作していない方の Focus を重ならない位置に移動させることによって、Focus の重なりを防止する機能である．

しかしながら、既存の重なり防止機能では、2.2 章で述べるように、視認性や、道路ネットワークの連続性が維持できないなどの問題があった．そこで、我々は、複数の Focus が近接した際、まるで水滴が表面張力により結合するかのようになり、自然な形で Focus が変形し結合する手法を提案する．

2. Focus+Glue+Context マップ Emma

2.1 Focus+Glue+Context マップ Emma とは

Emma は、図 1 に示すように、利用者が注目する拡大地図 (Focus) と、全体的な位置関係を把握できる地図 (Context) と、縮尺の異なる Focus と Context の間の道の歪みを吸収し、両領域を結ぶ経路をもつ領域 (Glue) をもつ．Emma は、Glue にひずみを適切に集中させることによって、Focus と Context をひずみなく描画することが可能である．その一方で、Glue は放射方向の縮尺が小さくなりすぎるため、すべての道路を描画すると密集して見にくくなる問題があったが、道なり道路選別手法 [4] により Glue に描画する道路を効果的に削減することで

この問題を解消した。



図 1 Focus+Glue+Context マップ

2.2 複数 Focus の重なり問題

従来、複数の Focus を 1 つの Context に表示させる際、Focus 同士が重なって一部の Focus や Glue が見えなくなってしまう問題があった。また、道路ネットワークの連続性を維持するように、Glue の内側は Focus と Glue の外側は Context と隣接していなければならない。

そこで、既存の Emma では偏り Focus 重なり防止機能により解決を試みた。偏り Focus 重なり防止機能とは、ユーザが操作している Focus が他の Focus と重なった場合に、ユーザが操作していない方の Focus を重ならない位置に移動させることによって、Focus の重なりを防止する機能である。しかしながら、Focus には注目する地図オブジェクトが含まれているため、Focus の緯度経度座標を変更することは望ましくない。そこで、偏り Focus 重なり防止機能では、Focus 領域内の緯度経度座標を固定したまま Focus の画面座標を移動させることで、Focus 領域内の注目する地図オブジェクトを維持したまま重なりを防ぐことを実現した。しかし、偏り Focus のまま移動し続けると図 2 のように Focus と Context 両方に同じ場所が表示される等の矛盾が発生する場合がある。そのため、重なり防止機能では、その矛盾を避けるために、Focus 同士が接近しすぎると図 3 のように他方の Glue が消滅するが、その結果、道路ネットワークの連続性が維持できなくなる。

重なり防止機能には以下の問題がある。

問題 1 Glue が消滅することによって、その Focus と Context との繋がりが分からなくなるなど、道路ネットワークの連続性が維持できない。

問題 2 地図の連続性を維持できないほど Focus を移動させることによって、Focus が本来見えるはずの Context 領域の一部をを覆い隠してしまう。

3. 提案システムの特徴

既存の偏り Focus 重なり防止機能の問題を解決するために、複数の Focus が変形・結合・分裂するシステムを提案する。以



図 2 Focus と Context に同じ場所が出現する問題



図 3 重なり防止機能による Glue 消滅．黒い影の部分をも眼鏡のように拡大している。

下に提案システムの特徴を挙げる。

特徴 1 Focus 同士が接近しても重ならないように両方の Focus が凹むように変形する。これにより図 4 のような複数の目的地域を Focus の位置に関係なく同時に見ることが可能になり、また Focus 同士が結合する際の Focus 間の距離を縮めることができる。Focus 変形の際、両 Focus 間の隙間を通常よりも狭くすることで、両 Focus の隣接部分の Glue が 1 つの Glue であるように見える。また、両 Focus の隣接部分は Glue の占める割合が高く視認性が低下する。そこで、隣接している部分のみ Glue の幅を半分にして視認性を向上させる（図 5）。それと共に隣接している互いの幅が半分の Glue 同士をもとの幅の一つの Glue が両 Focus の間に存在するように見せることができる。これにより Glue は幅が変化しても Glue の役目である Focus と Context の経路を繋ぐことに問題はない。Focus の変形は他の Focus と重なっている部分だけに適用され、他の Focus と重なっていない部分は変形前の形状を保っている。これにより Focus 同士が変形している状態は、水滴同士が表面張力により自然な形で変形しているようにみえる。Focus 同士が変形している状態から重ならない距離まで遠ざけると元の形の Focus に戻る。

特徴 2 Focus 同士が特徴 1 により変形しながら一定距離まで近づくと結合する。結合 Focus は結合前の Focus 同士の外周をかたどった形状であるので、Focus 同士の結合は水滴が表面張力により結合するようにみえる。結合 Focus の拡大率は近づいてきた方の Focus の拡大率と同じになる。また、図 6 のように結合する手前の距離まで近づくと結合した際の Focus の形を線で表示する。その線が表示されることで、ユーザは Focus 同士



図 4 隣接部分の Glue の幅を変えない場合 . Focus 同士ではさまれる Glue の幅が 2 倍あるように見える .



図 5 隣接部分の Glue の幅を半分にした場合 . Focus 同士ではさまれる Glue の幅が , そうでない Glue の幅と同じに見える .

が結合するタイミングと , 結合 Focus の形を結合前に知ることができる .



図 6 結合する手前で結合 Focus の形を黒線で表示

特徴 3 Focus をマウスドラッグで他の Focus に接近させ , それら 2 つの Focus が結合した後もマウスドラッグを続けることで , 結合 Focus がマウスポイントの移動に沿って変形する . この変形は , ユーザが操作している Focus が他方の Focus に吸収・放出されるように見える .

特徴 4 さらにマウスドラッグを続けて , マウスポイントを結合 Focus の外側に移動させると結合していた Focus が分裂しまた 2 つの Focus として操作できるようになる .

表 1 Focus 定義データ

L_{OF}	Focus による拡大前の Context 領域上で Focus 領域に入る領域を決める閉じたポリライン
L_F	Focus 領域と Glue 領域の境界を決める閉じたポリライン
L_G	Glue 領域と Context 領域の境界を決める閉じたポリライン
P_F	その Focus による拡大の中心点 (不動点)
L_{NG}	その Focus が重なっていない状態での L_G
L_{FmG}	その Focus が他の m 番 Focus との重なりを避けるために変形している状態での L_G
$P_n(L)$	L の n 番目の頂点の座標

4. 提案システムの実現法

4.1 データ

提案システムは以下のデータを使用する . Focus の形は凸多角形で N 多角形のもの扱うものとする . 座標系は Context 系の XY 座標を基準とした座標系を扱うものとする .

Focus 定義データ ユーザが状況に応じ Focus の位置と形状を指定する . Focus と Glue の位置・形状は , 表 1 と図 7 に示すようなデータからなる . Focus 定義データには以下の制約がある .

- L_{OF} は L_F に , L_F は L_G に , P_F は L_{OF} に完全に内包される ($L_{OF} \subset L_F, L_F \subset L_G, P_F \subset L_{OF}$)
- L_{OF} と L_F は相似の関係にある ($L_{OF} \sim L_F$)

L_{OF} は , 拡大して Focus 領域に表示される前の Context 領域上での地図領域を表している . L_{OF} を不動点 P_F を中心に縮尺を変更し座標変換を行うと , その範囲の領域はちょうど Focus 領域 L_F の位置に収まる . P_F は L_{OF}, L_F から計算可能である . L_{OF} は L_F に , L_F は L_G に , P_F は L_{OF} に完全に内包される必要がある . L_{OF} と L_F は相似の関係にあり , Focus 領域の拡大率により L_{OF} の大きさが変化する . L_F は Focus 領域と Glue 領域の境界を表しており , L_G は Glue 領域と Context 領域の境界を表している .

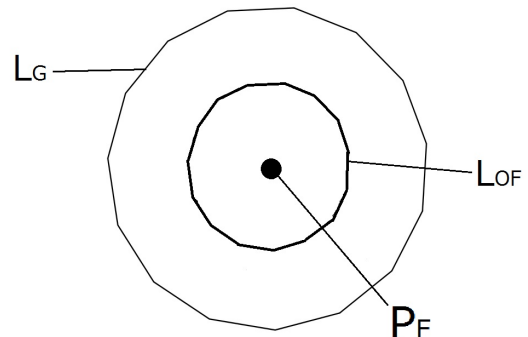


図 7 Focus 定義データ

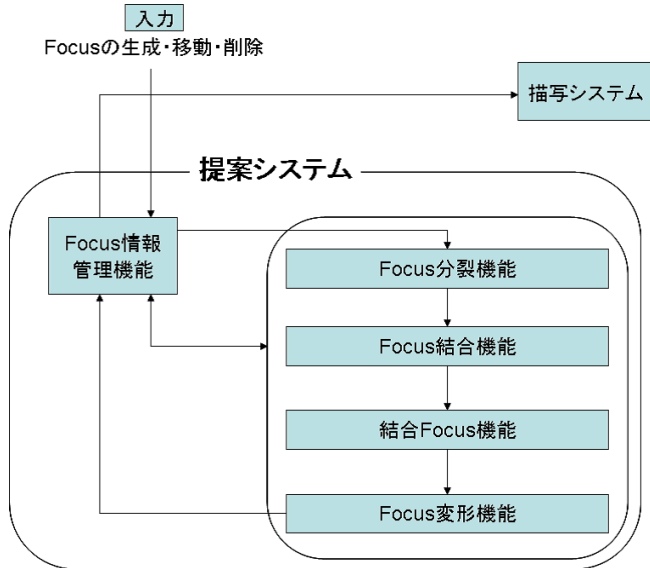


図 8 システム構成図

4.2 提案システムの構成

提案システムの構成を図 8 に示す。まず、Focus の生成・移動・削除が発生すると提案システムが呼び出され、Focus 情報管理機能が持つ Focus の情報を用いて、各機能の step を実行する。Focus 情報管理機能では、全ての Focus に識別子として番号を与え、各 Focus の m 番 (識別子) Focus に対して重なりを防ぐために変形した座標集合 L_{FmG} 、各 Focus の縮尺の値、各 Focus の元の形の座標集合 L_{NG} などを保持する。提案システムの各機能は Focus 情報に変更があれば、随時 Focus 情報管理機能に更新情報を反映させる。全ての処理が終わると Focus 情報管理機能のデータを描写システムに反映する。

4.3 Focus 変形機能

Focus 重なり問題の解決を試みた手法として偏り Focus 重なり防止機能を挙げたが、重なりを避けるために移動した Focus が他の Focus と重なる可能性があることから、ユーザが移動させる Focus 以外は極力移動を避けるべきである。そこで、Focus 同士を結合させて重なりを解消する方法が考えられるが、Focus と縮尺の関係上、4.4 章で述べるように、重なりの浅い Focus 同士を結合させるのは問題がある。したがって、Focus 同士がある程度深く重なるまで Focus を近づける必要があり、Focus 変形機能によりそれを実現する。

Focus の形を構成する各頂点が他の Focus の領域に入ったとき、その Focus の領域から出るように座標を変更する。各頂点を自分の Focus の中心点に向かって移動させる。変形していた Focus 同士が離れる場合、各頂点は元の位置に戻ろうとする。変形している部分の Glue 領域同士の間には Context 領域を存在させる。これにより Emma では地図の連続性が保障されているから、隣接している Glue 領域同士に表示される経路はつながっていると見える。Focus 変形機能のアルゴリズムを以下に述べる。

M 個の Focus が存在する時、Focus 情報管理機能は各 Focus に $0 \sim M - 1$ までの番号をつける。また、Focus 情報管理機能

は、元の変形していない Focus の各頂点の座標 L_{NG} と、変形後の Focus の各頂点の座標を各 Focus ごとに保持する。変形機能が呼び出されると、以下の処理を実施する。

step.1 Focus の識別子の番号として $i = 0, j = 1$ を設定する。

step.2 変形前の i 番 Focus と j 番 Focus が重なっているかを判定する。 i 番 Focus の $P_n(L_{NG})$ が j 番 Focus の L_{NG} の領域内に入っているかという条件と、 j 番 Focus の $P_n(L_{NG})$ が i 番 Focus の L_{NG} の領域内に入っているかという条件の 2 つの条件について判定を行う。その 2 つの条件の判定を n が 0 から $N - 1$ までの値をとる全ての場合で行い、どちらかの条件を満たす点が 1 つでも見つければ両 Focus は重なっているので、step.3 へ移行する。条件を満たす点が見つからなければ両 Focus は重なっていないので、step.6 へ移行する。

図 9 は、2 つの Focus が重なっている状態を表している。左の Focus の $P_1(L_{NG})$ が右の Focus の領域内に入っており、右の Focus の $P_8(L_{NG})$ と $P_9(L_{NG})$ が左の Focus の領域内に入っている。 $n = 1$ の時に左の Focus の $P_n(L_{NG})$ について条件が真になり、これらの Focus は重なっていると判断される。

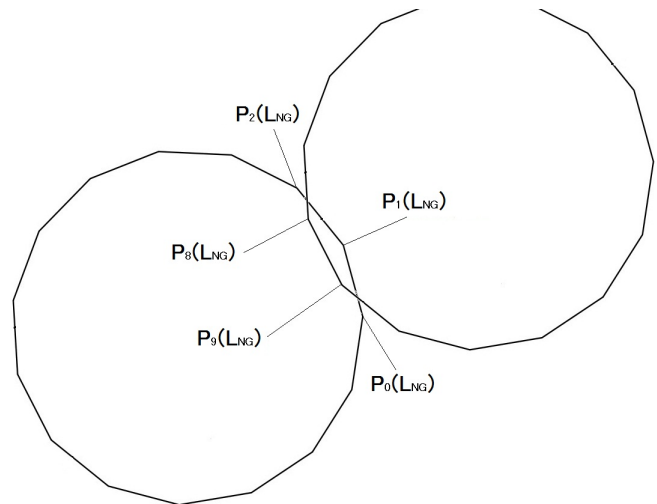


図 9 変形前 Focus の重なり判定

step.3 それぞれの Focus の L_{NG} の各頂点ごとに不動点 P_F との傾き R を計算する。 i 番 Focus の L_{FjG} と j 番 Focus の L_{FiG} の各頂点を、それぞれ不動点 P_F と各頂点を結び傾き R の線分に沿って少しずつ不動点 P_F から離れる方向に移動させる。その移動を、各頂点が以下の 2 つの条件のどちらかを満たすまで行う。

- 変形前の元の位置である $P_n(L_{NG})$ の位置まで戻る
- i 番 Focus なら j 番 Focus の L_{FiG} の領域に入る、 j 番 Focus なら i 番 Focus の L_{FjG} の領域に入る

step.4 step.3 の処理によって両 Focus とも相手 Focus と少しだけ重なっている状態であるので、重なりを解消する。相手 Focus の領域に入っている頂点を step.3 で求めた R の線分に沿って中心 avg に向かって少しだけ移動させる。この作業を i 番 Focus と j 番 Focus 交互に行い、全頂点が相手 Focus の領域から出るまで繰り返す。

step.5 上記 Step で変形した座標を相手 Focus の番号と共に Focus 情報管理部分に記録する .

step.6 j に 1 を加え , $j < N$ なら Step.2 へ移行する .

step.7 i に 1 を加え , $i < N - 1$ なら Step.2 へ移行する .

step.8 全 Focus は Focus 情報管理部分に複数の変形座標を持っているので , 各頂点ごとに一番中心に近い変形座標を選択する . その Focus 番号と座標データを描写システムに渡し , Focus 変形が行われる .

Focus の各頂点は , 中心との傾き R の直線に沿って移動するので , Focus は図 10 のように変形する . 変形している Focus 同士が離れる際に , step.3 の作業により , それぞれの Focus は変形した形状から元の形状に少しずつ復元される .

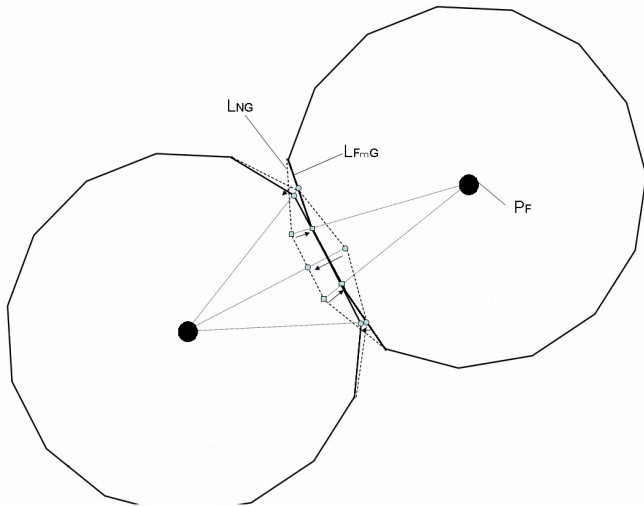


図 10 各頂点の移動による Focus 変形

4.4 Focus 結合機能

Focus 同士が一定距離まで近づいた時に , 両 Focus を削除し新しい Focus (結合 Focus) を生成することで 2 つの Focus が結合したかのように見える . 結合を行うかの判定は Focus を 2 つ 1 組で全ての Focus に対して小さい番号を持つものから全通り行うので , 3 つ以上の Focus が同時に結合することはない .

結合 Focus の L_{OF} は , 結合前のそれぞれの Focus の L_{OF} を含む必要がある . ここで , 図 11 では , 表示されている地図の縮尺が context 領域では $1/70000$, Focus 領域では $1/40000$ であり , 2 つの Focus が隣接している . 図 11 のそれぞれ 2 つの Focus の L_{OF} を表したのが , 図 12 の赤い丸の部分である . Focus 同士が隣接していても Focus 領域に表示される地図領域同士は離れており , Focus 領域と Context 領域の縮尺の差が大きくなるほど L_{OF} は小さくなる .

重なり浅い Focus 同士を結合させると , できた結合 Focus の L_{OF} に元の 2 つのそれぞれの Focus の L_{OF} をあまり含むことができず , 元の 2 つのそれぞれの Focus 領域に表示していた地図領域は結合 Focus 領域に表示されなくなってしまう . 図 12 の黒い影は , 図 11 の 2 つの Focus をその位置で結合させた際にできる結合 Focus の L_{FO} を表しており , 赤い丸で囲んだ領域は , 図 11 の 2 つの Focus の L_{OF} を表している . 黒い影に赤い丸で囲んだ領域があまり含まれていないことから , Focus



図 11 Focus 同士が隣接



図 12 図 11 の 2 つの Focus の L_{FO} と , 結合 Focus の L_{FO}

同士がある程度深く重なってから Focus 同士を結合させるべきであることがいえる .

Focus 結合のアルゴリズムを以下に示す .

step.1 $i = 0, j = 1$ と初期値を設定する .

step.2 i 番 Focus の P_F と j 番 Focus の P_F の距離が一定値 D_1 より小さければ次の step へ移行し , 大きければ Focus を結合させず結合機能を終了する .

step.3 生成される結合 Focus の L_G を黒線で表示する . i 番 Focus の L_{NG} の頂点と j 番 Focus の L_{NG} の頂点の全ての組において最も距離が遠い頂点の組みを算出する . その算出した 2 つの頂点と , それら 2 つの頂点からそれぞれ両隣 $(M - 2)/4$ 個の頂点を取得する . それらの頂点で構成したポリラインを結合 Focus の L_G とし , 黒線で表示する .

step.4 i 番 Focus の P_F と j 番 Focus の P_F の距離が一定値 D_2 (ただし $D_2 < D_1$) より小さければ , i 番 Focus と j 番 Focus を削除し , 結合 Focus を生成する . 結合 Focus の L_G は step.3 と同様の方法で求める . 結合 Focus 変形機能のために i 番 Focus の L_{NG} と j 番 Focus の L_{NG} を保存する .

図 13 は , step.3 で示した方法により求めた結合 Focus の形を太い黒線で表示している . 図 13 の丸は , 2 つの Focus の各頂点間で一番距離が大きい頂点の組み合わせであり , それらの頂点とその両隣 3 個の頂点で結合 Focus の L_{NG} を構成している .

4.5 結合 Focus 変形機能

Focus 同士が結合する瞬間からマウスボタンを離さずにマウスドラッグを続けて行くと , マウスポインタの移動に沿って

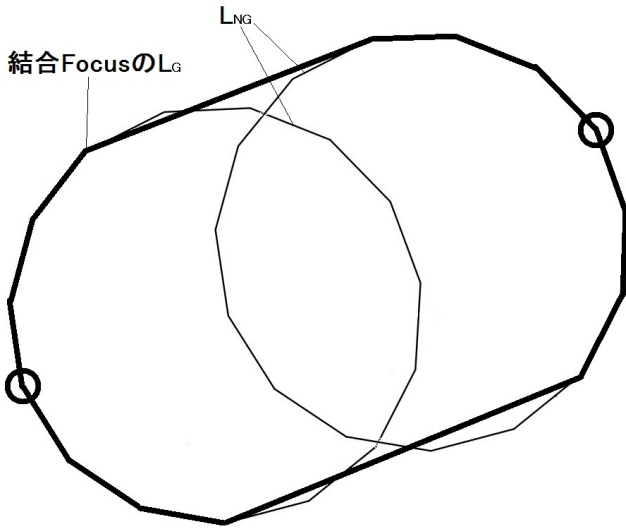


図 13 結合 Focus の形の決定法

結合 Focus が変形する．マウスボタンを離れた時の形に結合 Focus の形を固定する．

結合 Focus 変形機能のアルゴリズムを以下に示す．

step.1 4.4 章の step.4 で結合 Focus 生成時に保存した 2 つの Focus の L_{NG} を読み出す．

step.2 2 つの L_{NG} のうちユーザがマウスドラッグで操作している方の Focus の L_{NG} を，マウスポインタの動きに合わせて移動させる．その移動した L_{NG} を Focus 分裂機能のために保存しておく．

step.3 その 2 つの L_{NG} に対して 4.4 章の step.3 と同様の方法を用いて，結合 Focus の L_G を再計算して更新する．

4.6 Focus 分裂機能

結合 Focus を削除し，お互いに変形させ合っている 2 つの新しい Focus を生成することで Focus が分裂したかのように見せる．結合前の 2 つの Focus の L_G を保存しておき，結合 Focus が分裂する際にその情報を反映させる．分裂は Focus が結合した時からマウスドラッグを続けた場合にのみ発生し，分裂した 2 つの Focus がそれぞれ表示する地図領域は重なっていない．以下に Focus 分裂機能のアルゴリズムを示す．

step.1 結合前の 2 つの Focus のうち，ユーザが移動操作をしていない方の Focus の P_F と，4.5 章の step.2 で保存しておいた Focus の L_{NG} から計算した P_F ，2 つの P_F の距離を計測する．

step.2 その距離が一定値以上になれば，結合 Focus を削除し，その 2 つの L_{NG} をそれぞれもつ 2 つの Focus を生成する．

分裂した Focus は重なっているが，後で Focus 変形機能が呼び出されるので重なったまま描写されることはない．

5. 評価実験

提案手法を評価するために，実験 1 と実験 2 の 2 つの実験を行った．

5.1 実験の目的

実験 1

提案システムが，従来システムよりも道筋とチェックポイントを同時に確認しやすいことを明らかにする．

実験 2

提案システムにより形状が変更された Focus が，地図上の情報を読み取るのに適していることを明らかにする．

5.2 実験方法

実験 1

提案システムと従来システムを用い，被験者が以下の課題を行う．その後，アンケート結果，操作回数，所要時間を比較する．以下に課題の詳細な手順を述べる．

step.1 各システムの動作に慣れるため，従来システム（従来の偏り Focus 重なり防止機能をもつ Emma）と提案システムをそれぞれ 3 分ずつ練習する．

step.2 図 14 で示すように，設定した開始地点と 2ヶ所のチェックポイントにそれぞれ Focus を配置しておく．被験者は，開始地点の Focus を指定された道筋に沿って移動させながら，チェックポイント付近の指定した建物を見つける．これらの作業を両システムを用いてそれぞれ行う．このとき被験者の操作回数（マウスのボタンを押してから離すまでを 1 回とする）と時間をそれぞれのシステムごとに測定する．

step.3 被験者は，道筋と建物を含む略地図を step.2 の作業中に読み取った情報を思い出して描く．

step.4 被験者は，ユーザビリティに関するアンケートに答える．

step.3 で用いる地図を図 14 に示す．開始地点は一番左の Focus の中心であり，黒線矢印は Focus を移動させる道筋である．

被験者は，指定した道筋の周辺地域を Focus に表示させて詳細情報を確認しつつ，全体地図を見ながら，チェックポイントとチェックポイント付近の指定した建物の位置関係を確認する．なお，チェックポイント付近の指定した建物は小さいので，見つけるためには Focus 領域内に表示して確認する必要がある．

アンケートでは，以下の各項目についてそれぞれ，5:非常に良い，4:良い，3:普通，2:悪い，1:非常に悪い，の 5 段階で回答する．

- 道路のつながりの認識性
- 自然に感じたか
- 処理速度
- おもしろさ
- 快適に使えたか
- 使いやすさ
- 注目地域を目で追いやすかったか

実験 2

被験者に，従来システムにおける偏った Focus（図 15）と Glue が消滅した Focus（図 16）と，提案システムにおける変形した Focus（図 17）と結合した Focus（図 18）の 4 つの Focus を提示する．被験者は提示された各 Focus に対して以下のアンケート項目にそれぞれ，5:非常に良い，4:良い，3:普通，2:悪い，1:非常に悪い，の 5 段階で回答する．



図 14 評価実験の Focus 配置と道筋

- 道路のつながりの認識性
- Focus 内地域と Focus 周辺地域の位置関係の把握しやすさ
- 自然に感じたか



図 15 偏った Focus

図 16 Glue が消滅した Focus



図 17 変形した Focus

図 18 結合した Focus

5.3 実験結果と考察

実験 1

操作回数と作業時間の平均値は、従来システムでは 7.1 回と 63.8 秒であり、提案システムでは 1.0 回と 49.1 秒であった。操作回数と作業時間はどちらも提案システムの方が低くなっており、特に操作回数に関しては全ての被験者が従来システムでは 5 回以上操作したが、提案システムでは 1 回だけしか操作していない。操作回数は移動させる Focus を切り替えた回数である。従来システムと提案システムに対するアンケート結果の平均値のグラフを図 19 と図 20 に示す。グラフの縦軸は各項目の当てはまる度合いであり、棒が長いほど項目が当てはまることを示す。

を示す。

「処理速度」の項目以外は全て提案システムの方が高い数字であり、実験 1 の目的「道筋とチェックポイントを同時に確認しやすい」に適しているシステムであることが確認できた。また、アンケートの「道路の繋がり認識性」の項目が提案システムの方が高かった。これらのことから、提案システムでは Focus 間の道路の繋がり認識できるので移動させる Focus を切り替える必要が無かったことが分かる。よって、提案システムは 2.2 章の問題 1 を解決すると考えられる。

また、「注目地域を目で追いやすかったか」の項目については、提案システムは従来システムの 2 倍近い値を得ており、このことは、提案システムがユーザが注目する地図情報を常に Focus 領域内に表示し続けたことを表している。よって、提案システムは 2.2 章の問題 2 を解決すると考えられる。

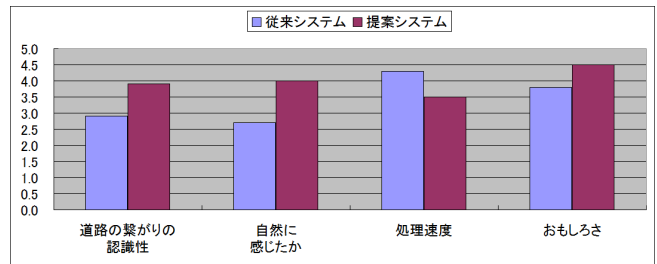


図 19 実験 1 のアンケート結果 1

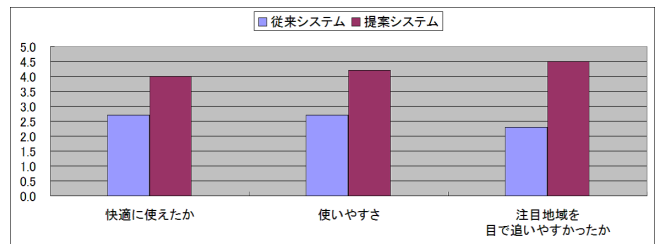


図 20 実験 1 のアンケート結果 2

実験 2

偏った Focus (図 15)、Glue が消滅した Focus (図 16)、変形した Focus (図 17)、結合した Focus (図 18) の 4 種類の Focus に関するアンケート結果の平均値をグラフ化したものを図 21 に示す。変形した Focus は、全てのアンケート項目に対して偏った Focus 以上の評価が得られた。全てのアンケート項目に対して、結合した Focus が他の Focus より評価が高く、かつ 4.5 を超える高い数字を示した。また、Glue が消滅した Focus は、全てのアンケート項目に対して最も評価が低く、特に「道路の繋がり認識性」の項目に対しては結合した Focus の半分以下となっている。これらのことから、提案システムにおける Focus 変形・結合手法は 2.2 章の問題 1 を解決すると考えられる。

6. おわりに

本研究では、Focus+Context+Glue マップ Emma における

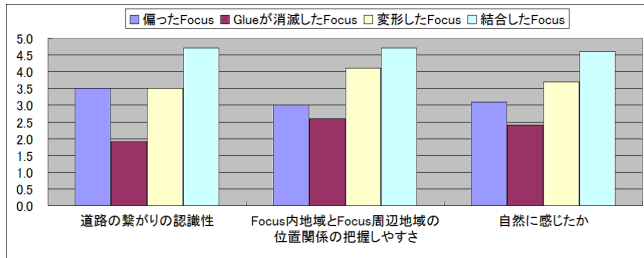


図 21 実験 2 のアンケート結果

Focus 変形・結合・分裂手法を提案した。今後の課題は、以下の 3 つである。

- 結合する 2 つの Focus の縮尺に大きな差がある場合や両 Focus の縮尺が大きい場合に、結合 Focus の大きさと縮尺をどのようにするか検討すること
- カーナビなどのナビゲーションシステムと比較し、評価・考察を行うこと
- プロトタイプシステムが散歩などの移動中に使用できるようなデバイスとインターフェースを考案すること

文 献

- [1] GoogleMaps. <http://maps.google.co.jp>
- [2] Naohisa Takahashi: An Elastic Map System with Cognitive Map-based Operations, International Perspectives on Maps and the Internet, Springer-Verlag, pp.73-87, Feb. 12, 2008 .
- [3] Daisuke Yamamoto, Shotaro Ozeki, Naohisa Takahashi, Focus+Glue+Context: An Improved Fisheye Approach for Web Map Services, Proceedings of the ACM SIGSPATIAL GIS 2009, Seattle, Washington, pp.101-110, 2009.11
- [4] 山本大介, 小関章太郎, 高橋直久, 道なり道路選別手法に基づく Fisheye View マップ総描手法, 電子情報通信学会論文誌, システム開発論文特集号, Vol.J93-D, No.10, pp.1914-1925, Oct. 2010.
- [5] アクションスクリプト 3.0 http://livedocs.adobe.com/flex/3_jp/langref/index.html
- [6] "Google Earth," 2010. <http://earth.google.com>
- [7] "Yahoo! 地図," 2010. <http://map.yahoo.co.jp>