

交通機関のアクセスポイントを考慮した Focus+Glue+Context マップシステムの実現

平見 祐樹[†] 山本 大介[†] 高橋 直久[†]

[†] 名古屋工業大学大学院工学研究科情報工学専攻 〒466-8555 愛知県名古屋市昭和区御器所町

E-mail: [†]hirako@moss.elcom.nitech.ac.jp, ^{††}{yamamoto.daisuke, naohisa}@nitech.ac.jp

あらまし Focus+Glue+Context マップシステム EMMA は、大縮尺の注目領域 (Focus) と小縮尺の周辺領域 (Context), および、縮尺差により生じる歪みを吸収して両者を結ぶ領域 (Glue) からなる視認性と一覽性に優れた Web マップシステムである。Web マップの使用目的の1つとして、遠方から交通機関を使用して目的地へ向かう時の行き方を調べることがある。その際ユーザーは、どのように交通機関を乗り継いで目的地の最寄りにあるアクセスポイント (駅やバス停等) へ行くのかと、そのアクセスポイントからどのような経路を歩いて目的地に向かうのかを考える。本稿では、このような使用状況を想定し、交通機関のアクセスポイントを考慮した Focus+Glue+Context マップシステムを提案する。Focus には目的地の周辺にあるバス停や駅などのアクセスポイント (ローカル AP) を、Context には新幹線や飛行機などの長距離移動で使われるような大きなアクセスポイント (主要 AP) と、そこからローカル AP までの路線を表示することで、ユーザーは主要 AP からローカル AP までの路線経路と、ローカル AP から目的地までの歩行経路を同一画面上で1度に閲覧することが可能となる。

キーワード Web マップ, 位置情報システム, ビジュアライゼーション, コンテキストアウェアネス, 地理情報システム

1. はじめに

近年, Google マップや Yahoo!地図等の Web マップシステムが広く普及し使われている。Web マップシステムを使うことで、ユーザーは縮尺や表示範囲を自由に選んで地図を閲覧することができる。しかし、これらの Web マップシステムでは、画面上に表示される地図は同一縮尺のため、縮尺を変更する際は画面全体の縮尺を変更する必要がある。例えば、交通機関を利用して目的地へ行くときの行き方を調べる場合を考えてみよう。目的地が遠方であれば、新幹線や飛行機等の長距離移動に適した交通機関を利用するため、空港やターミナル駅などのような広域においても目印となる大きなアクセスポイント (以下主要 AP と呼ぶ) を経由することになる。そして主要 AP からは鉄道やバスを乗り継いで目的地の最寄りにあるアクセスポイント (以下ローカル AP と呼ぶ) まで行き、そこから徒歩で目的地へ向かう。主要 AP からローカル AP への路線経路を決めるのには、目的地、主要 AP、ローカル AP、路線の経路を表す線が含まれる広域地図 (小縮尺の地図) が適しており、ローカル AP から目的地までの歩行経路を決めるのには、より詳細に道路が描画されている詳細地図 (大縮尺の地図) が適している。よって従来の Web マップを使用して行き方を決める場合、広域地図と詳細地図を切り替えながら使用しなければならない。

この問題を解決するため、我々は図 2 のような Focus+Glue+Context マップシステム EMMA [1] [2] [3] を提案した。EMMA は広域と詳細を同時に 1 枚の画面で閲覧可能な Focus + Glue + Context マップを提供する。Focus + Glue + Context マップは縮尺の異なる 3 つの領域によって構成される。注目する部分を大縮尺で表示する Focus 領域、周辺部分を小

縮尺により表示する Context 領域、2 つの領域の間で縮尺差によって発生する歪みを吸収する Glue 領域の 3 つである。このように、Focus と Context の間に可変縮尺の Glue を導入することで、Focus と Context を一定の縮尺で描画することが可能となり、視認性と一覽性を両立させたマップを実現している。

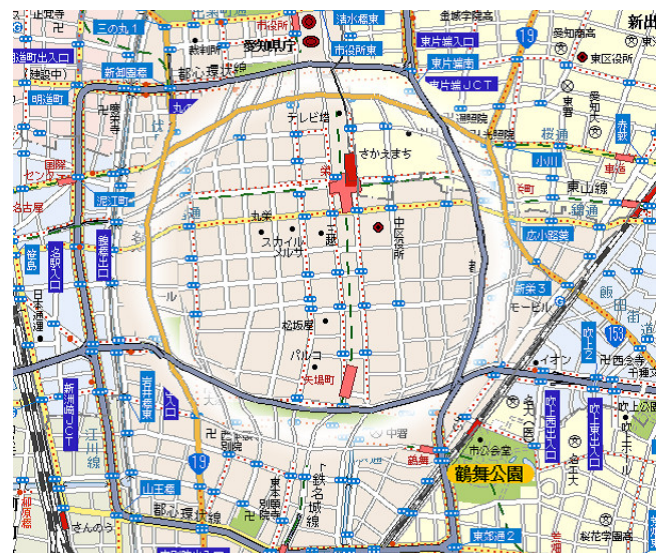


図 1 Focus+Glue+Context マップ EMMA

これにより、先程の例のような場合においても、目的地からローカル AP までの経路の詳細と、目的地と主要 AP の位置関係の把握を同一画面上で1度に行うことが可能となった。一方で、本当に有用なマップを実現する上で、以下に示すような問題が残っている。

問題 1 Glue は Focus の生成によって隠れてしまう領域を限られたスペースに圧縮する。そのため、Glue 上には重要な道路を選別して描画する [4] [5]。また、地名や建物名等の地図オブジェクト情報も描画しない。Focus が大きくなると、Glue に圧縮される領域も増大し、地図から得られる地図オブジェクト情報が減少する。

問題 2 Focus + Glue + Context マップは 3 つの領域から構成されるため、拡大して閲覧したい領域に合わせて、それぞれの領域に以下の操作を行う必要がある。

Focus : 拡大したい領域を含むように大きさを変更
Glue : 道路の圧縮具合に合わせて領域幅を調整
Context : 閲覧したい領域が全て画面内に収まるよう地図移動、縮尺変更

本稿では、これらの問題を解決するため、交通機関のアクセスポイントとユーザーの目的地を考慮した Focus+Glue+Context マップシステムを提案する。

2. 関連研究

この章では、本研究に関連する研究について述べる。

2.1 Fisheye View map

広域地図と詳細地図を同じ画面上で表示する手法として FisheyeView マップ [6] [7] がある。地図全体に円形の魚眼レンズを当てたような変形を施すことで、全体の位置関係と目的地の詳細を同時に把握することを可能としている。この地図では、中心付近の様子は詳細に知ることができるが、周辺部は道路が密集してしまい道のつながりが視認しにくいという問題がある。また、地図全体にわたって大きな歪みがあるため、距離感の把握も難しい。EMMA では、このような Fisheye View マップに Glue を加えた、Focus+Glue+Context 型マップを実現している。Glue に地図の歪みを全て集中させることで、注目領域 (Focus) と広域 (Context) を歪み無く一定の縮尺で描画することが可能となる。また、各地図座標に動的に再配置関数を作成し、それに沿って Focus を生成することで、自由な形状の Focus を生成可能にしている。

2.2 EMMA における任意形状 Focus

準凸多角形 Focus 生成手法 [8] を用いることで、凹多角形 Focus の生成が可能となる。それまでは、円形や矩形のような凸多角形の形状の Focus しか生成できなかったが、この論文では、Focus の形状を準凸多角形へと変形することで凹多角形の Focus も生成可能となることを明らかにしている。しかし、この手法では、Glue における歪みについては考慮されていないため、不動点が観測点領域の端に定められた場合、Glue の歪みが大きくなり視認性が低下してしまう。さらに Focus が鋭角を含んでいる場合、その周辺の道路や鉄道などが Glue において大きく歪んで視認性が著しく低下するという問題も存在する。また、この手法では、Focus の形状はユーザーが頂点座標を全て入力することによって行っているが、本研究では、ユーザーが

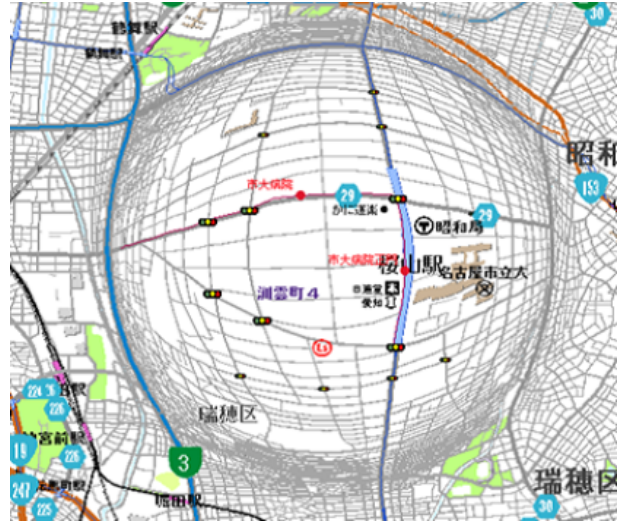


図 2 Fisheye View Map

指定した地点から徒歩圏内にあるローカル AP をもとに Focus の形状を決定する。

3. 提案システムの概要

この章では、提案システムの目標と、システムの構成について述べる。

3.1 提案システムの目標

(1) Access-Points-Based map

提案システムでは、Focus+Glue+Context マップシステム EMMA を用いて、図 3 のような Access-Points-Based map の実現を目指す。Access-Points-Based map は Focus, Glue, Context の 3 領域で構成される。Focus は目的地を中心とし、目的地の近傍と目的地からローカル AP までの道路を大縮尺で表示する。Context は主要 AP と、主要 AP とローカル AP を結ぶ路線を小縮尺で表示する。Glue は可変縮尺で道路を表示し、Focus と Context の道路を滑らかにつなぐ。ユーザーはこのマップを用いることで、主要 AP からローカル AP までの路線経路と、ローカル AP から目的地までの歩行経路を 1 つの画面で閲覧することができ、目的地まで行くために使用するローカル AP や、その際に使う路線を柔軟に選択することが可能となる。

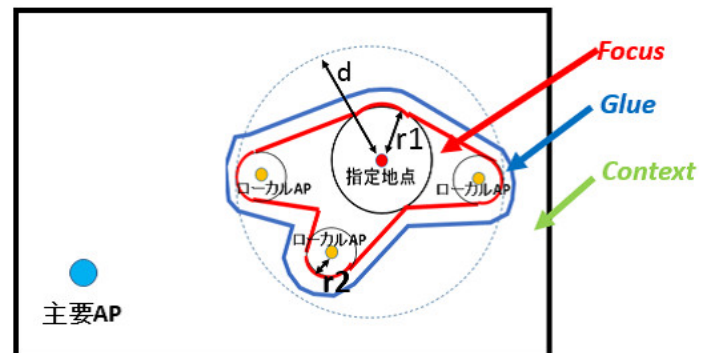


図 3 Access-Points-Based map

(2) 準凸多角形 Focus の生成

EMMA では図 4 に示すように、Focus を形成する各頂点を不動点と呼ばれる基準点と逆方向に遠ざけるように再配置することで Focus を生成する。この際、不動点は Focus を形成する多角形の外部を通ることなく全頂点と線分を結ぶことができる領域を持つ必要がある。このような領域を持つ形状を準凸多角形という。たとえば、図 5 において、左の形状は Focus を形成する多角形の外部を通ることなく不動点と全頂点を結ぶことができるため準凸多角形であるが、右の形状では不動点をどの位置に配置しても、多角形の外部を通ることなく不動点と多角形の全ての頂点を結ぶことができないため準凸多角形ではない。準凸多角形でない形状の Focus を生成すると、道路の繋がりが途切れてしまったり、地図上の各地点の位置が実際とは異なる位置で表示されてしまう。そのため、EMMA で生成する Focus の形状は必ず準凸多角形にする必要がある。

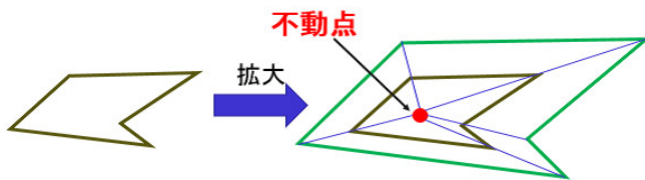


図 4 EMMA における Focus の生成

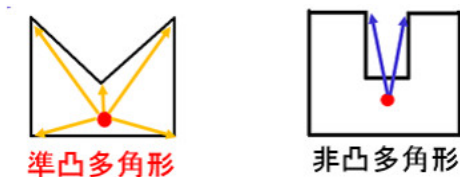


図 5 準凸多角形の例

3.2 提案システムの構成

提案システムの構成を図 6 に示す。提案システムは、Focus 形状算出機能、Glue 画像サーバー、Context 補正機能によって構成される。ユーザーはブラウザ上の地図で地点の指定を行い、その座標が提案システムへの入力となる。Focus 形状算出機能では、入力された地点の座標をもとに Focus の形状を決定する。まず、アクセスポイントデータベースより、目的地から徒歩圏内にあるローカル AP の情報を取得する。それらの駅を Focus 内に含み、かつ必要無い領域を除いた形状を求め、それを Focus の形状とする。そして、Glue 画像サーバーによって、求めた Focus 形状に合わせて Glue 画像を生成する。Context 補正機能では、アクセスポイントデータベースから取得した主要 AP の位置情報をもとに、Context に主要 AP が含まれるような地図の縮尺と表示範囲を計算し Context をその結果に合わせて補正する。そして最後に、地図描写システムによって Context 画像の上に Glue 画像、Focus 画像を適切な配置で重ね合わせることによって Focus+Glue+Context マップを生成して出力する。

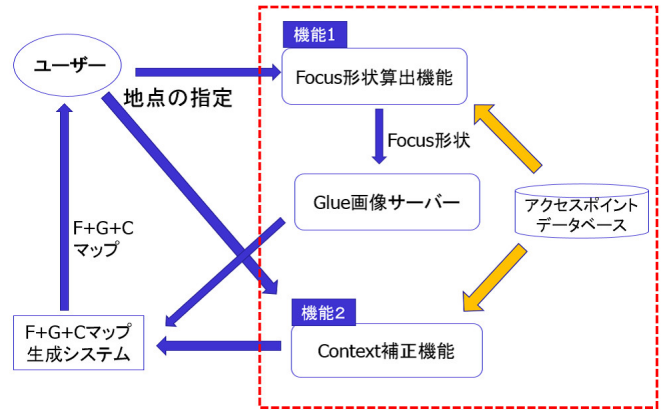


図 6 システム構成図

4. 提案システムの実現法

この章では、提案システムを実現するための各機能について述べる。

4.1 Focus 形状算出機能

4.1.1 Focus 形状の要件

以下に示す領域を Focus 内に含むことを、Focus 形状の要件とする。

- 指定地点の近傍
指定地点の近傍は、ユーザーが閲覧したい可能性が最も高いため Focus に含むべきである。今回は近傍を指定地点から半径 500m と定めた。
- 指定地点から徒歩圏内にあるローカル AP
今回、ローカル AP を経由して指定地点へ行くことを想定した地図の生成を目的としているため、指定地点から徒歩圏内のローカル AP を Focus に含める。今回は徒歩圏内を指定地点から半径 1.5km と定めた。
- ローカル AP の周辺
ローカル AP から指定地点に行く過程で別の地点に寄り道する可能性を考慮すると、ローカル AP 周辺のエリアも、ユーザーにとって重要であるため Focus に含めるようにする。今回は周辺範囲をローカル AP から半径 100m と定めた。
- 指定地点からローカル AP までの経路確認に必要な領域
ローカル AP から指定地点への経路は必要不可欠であり、それらの道路は Focus 内に含むべきである。

4.1.2 形状の決定に必要なローカル AP 情報の取得

交通機関のアクセスポイントに関する情報を格納したアクションポイントデータベースを用いて、Focus の形状を決めるのに使用するローカル AP を選別する。今回はローカル AP として、鉄道の駅を用いる。以下に、選別方法を示す。

step1 指定地点の半径 1.5Km 以内にある鉄道のアクセスポイントを、ローカル AP の位置情報として取得する。

step2 以下に示すような、Focus 形状の決定に不必要なローカル AP を、取得した情報の中から削除する。

- 複数の路線があるローカル AP
複数の路線を持つ駅は、データベース上では路線ごとに別々の駅として登録されているが、Focus の形状を求めるにあたって必要な AP の位置情報は、同地点に対して 1 つであれば良い。
- 指定地点から 500m 以内にあるローカル AP
指定地点から 100m 以内の領域は、4.1.1 で設定した要件を満たしており必ず Focus に含まれるため、その領域にあるローカル AP は形状を求めるのには不必要である。
- 他のローカル AP と指定地点の経路上に存在するローカル AP
指定地点からローカル AP までの経路にあたる領域は 4.1.1 で設定した要件を満たしており必ず Focus に含まれる。よってその領域に存在するローカル AP は Focus の形状を求めるのには不必要である。

4.1.3 形状算出方法

Focus の形状を求めるアルゴリズムを以下に示す。

- step1** 4.1.2 で求めたローカル AP をを中心とする半径 100m の円と、指定地点を中心とする半径 500m の円を結ぶ共通接線（基準線）を引く
- step2** 隣り合う頂点の基準線同士が交点を持つ場合は交点を Focus の新頂点として加え、持たない場合は接点の間を円に近似する

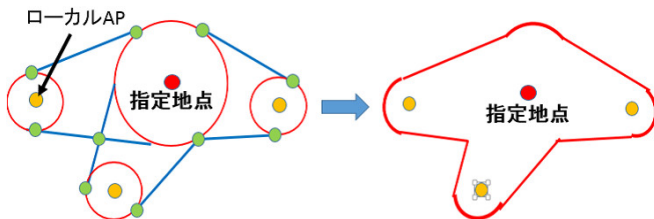


図 7 形状決定までの流れ

4.2 Context 補正機能

目的地から最も近い主要 AP と、そこからローカル AP までの路線経路が Context に表示されるように、Context の縮尺と表示範囲を補正する。今回は主要 AP として、空港と新幹線の駅を用いる。

- step1** アクセスポイントデータベースより、ユーザーの指定した地点から最も近い主要 AP の位置情報を取得する。
- step2** step1 で取得した主要 AP の位置情報と指定地点の位置情報を基に両者の距離を算出する。
- step3** step2 で求めた距離と地図を表示するブラウザのサイズを基に、指定地点、主要 AP、ローカル AP を全て画面内に含み、かつ Context の縮尺が最大になるよう縮尺を定める。

- step4** 指定地点、主要 AP、ローカル AP が全て画面内に収まるように Context の表示範囲を調整する。

5. プロトタイプシステム

提案システムの機能を全て持つプロトタイプシステムを実装した。サーバサイドに Java、クライアントサイドに Javascript を使用した。また、Web マップシステムとしてのインターフェースを実現するために、Javascript ライブラリである Lealet [9] を活用した。地図データに関しては、OpenStreetMap [10] から取得した。ローカル AP データと主要 AP データの検索には PostgreSQL と PostGIS [11] を用いて構築したデータベースを使用した。また、ローカル AP として使用する鉄道駅のデータは駅.jp [12] から取得した。

5.1 アクセスポイントデータベース

交通機関のアクセスポイントに関する情報を格納したアクセスポイントデータベースを構築した。表 1 にアクセスポイントデータベースのデータ構造を示す。今回の提案システムでは、ローカル AP として鉄道駅を、主要 AP として空港と新幹線の駅を用いた。表 1 に示すようなデータを用意することで、高速道路の IC やバス停をローカル AP とすることも可能である。

表 1 アクセスポイント DB テーブル

カラム	データ型	説明
AP_id	int	アクセスポイントを識別する ID
AP_type	varchar	交通機関の種類
AP_name	varchar	アクセスポイント名
line_cd	int	路線コード
lon	int	アクセスポイントの経度
lat	int	アクセスポイントの緯度

5.2 Glue 画像サーバー

OpenStreetMap の地図画像を用いた Glue 画像を生成する Glue 画像サーバの実装を行った。基本的な構造は、文献 [1] と同じである。必要なパラメータを組み込んだ URL をサーバにリクエストすることで、様々な形状の Glue 画像を返すことが可能である。リクエストパラメータは以下の 5 つである。

- centerLngLat : Focus 中心の緯度経度
- focus_zoom_level : Focus の縮尺
- context_zoom_level : Context の縮尺
- shape_inner_polygon : Glue 内側の形状 (Focus 中心からの相対座標)
- shape_outer_polygon : Glue 外側の形状 (Focus 中心からの相対座標)

このようなリクエストパラメータを持つ URL を Glue 画像サーバにリクエストすることで、図 8 のような Glue 画像を取得することができる。一番内側の枠は拡大前の Focus 形状を表す。その 1 つ外側の枠が拡大後の Focus 形状を表す。1 番外側の枠は、Glue 部分を含めた形状を表す。そのため、中間の枠と外側の枠の間の部分が Glue として使用される。

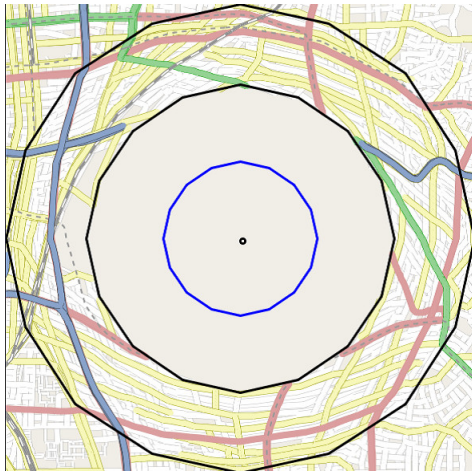


図 8 Glue 画像

5.3 Focus+Glue+Context マップシステム

図 9 のように、ベースとなる Context 領域の上に、Glue 画像サーバーから取得した Glue 用の画像、その上に、Focus の地図画像を重ねて表示することで、Focus+Glue+Context マップを実装した。従来の Focus+Glue+Context マップでは、ユーザーが各領域に対して操作を行っていたが、今回実装した提案システムでは、ユーザーが指定した目的地と交通機関を考慮して Focus の形状と Context の表示範囲、縮尺をシステム側で決定し、Focus+Glue+Context マップを生成する。

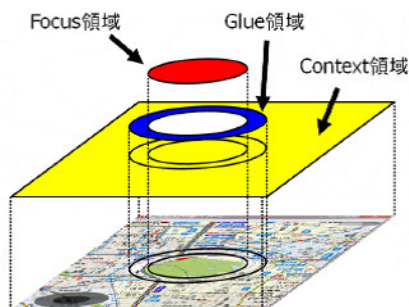


図 9 Focus+Glue+Context マップの構成

5.4 システムのインターフェース

実装したプロトタイプシステムの概観を図 10 に示す。プロトタイプシステムは地図画面、地図縮尺ボタン、Glue 道路選別手法選択ボタン、円形 Focus 生成ボタン、AP_Focus 生成ボタン、Clear ボタン、Context 補正ボタンによって構成され、これらのボタンとマウスによって操作を行う。各ボタンを押した際の挙動を以下に示す。

● 地図縮尺ボタン

地図の縮尺を変更することが可能。縮尺の変更はマウスのホイールを操作によっても行うことができる。

● Glue 道路選別手法選択ボタン

Focus を生成する際、Glue に描画する道路の選別方法を選択することが可能。選択項目は以下のとおりである。

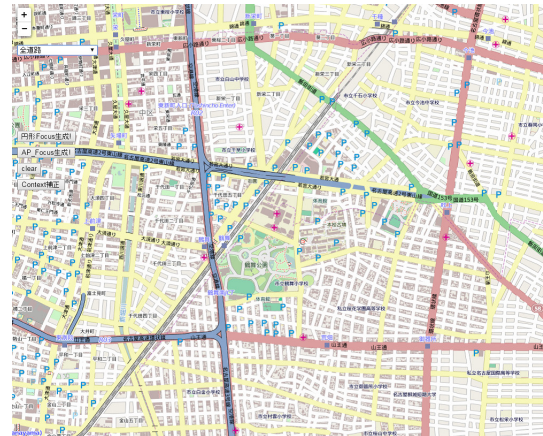


図 10 プロトタイプシステムの概観

－ 全道路

全ての道路を使用する。

－ 車

車が通れる道路を使用する。

－ ストローク

ストロークと呼ばれる道なりに続く道路を使用する。

－ コネクティビティ優先

Focus と Context の道のコネクティビティを考慮し、Focus に描画されている道路を優先的に使用する。

－ 小関らによる選別手法 [1]

接続する道路のなす角の大きさと道なりかどうか判別し、道なりと判定された道路を優先的に使用する。

－ 村瀬らによる選別手法 [5]

ストロークの長さや道路周辺に位置する施設情報に基づいてストロークの重要度を決定し、重要度の高いストロークを優先的に使用する。Focus 付近における道路の偏りが少ない。

● 円形 Focus 生成ボタン

ユーザーは右クリックにより地図上の地点にピンを立てることができる。その状態で円形 Focus 生成ボタンを押すことで、ピンを立てた地点を中心とする円形の Focus を生成する。生成した Focus は、マウス操作によって Focus 範囲の移動や、大きさの変更が可能である。また、Focus のみ縮尺を変更するといった操作を行うことも可能である。

● AP_Focus 生成ボタン

提案システムとして述べた機能を実現した部分。ユーザーは右クリックにより地図上の地点にピンを立てることができる。ピンを立てた状態で AP_Focus 生成ボタンを押すことで、ピンの地点を目的地として交通機関のアクセスポイントを考慮した Focus+Glue+Context マップを生成する。Focus は目的地から徒歩圏内にある全てのローカル AP とそこまでの経路を表示し、Context は目的地から最も近い主要 AP を含み、かつ最大の縮尺で表示する。

● Clear ボタン

生成された Focus や右クリックで立てたピンを消去し、地図表示範囲、地図縮尺を全てデフォルトの状態に戻す。

● Context 補正ボタン

ユーザーが指定指定した地点から最も近い主要 AP を地図上に含み、かつ縮尺が最大となるよう地図の表示範囲と縮尺を補正する。AP_Focus 生成ボタンとは異なり、Focus は生成しない。

5.5 システムの動作

ユーザーはブラウザ上の地図で、マウスの右クリックによって図 11 のようにピンを立てて地点の指定を行う。その後、画面の左側に位置する AP_Focus 生成ボタンを押すことで、Focus+Glue+Context マップの生成を行う。図 12 が生成された Focus+Glue+Context マップである。4.1.1 で定めた領域を含む形状の Focus を生成している。図の例では、指定地点の徒歩圏内に 2 つのローカル AP(桜山駅、瑞穂区役所駅)があり、それら周辺と、指定地点からそこまでの経路を含む Focus が生成されていることが確認できる。また Focus 生成前の地図画面には主要 AP が含まれていなかったが、Focus 生成後の地図画面は、Context の縮尺と表示範囲が調整され、指定地点から最も近い主要 AP(名古屋駅)を画面上に含んでいることが確認できる。

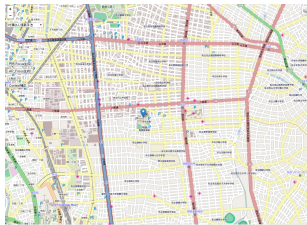


図 11 Focus 生成前

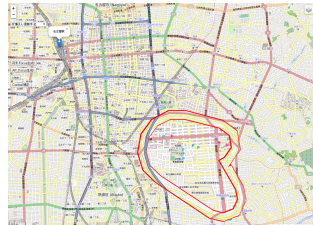


図 12 Focus 生成後

6. 提案システムの評価

提案システムの有効性を実験的に評価する。

6.1 評価実験の目的

遠方の目的地へと行く場合、新幹線や飛行機等の長距離移動に適した交通機関を使用して主要 AP へ行き、そこから鉄道でローカル AP へ行き、そしてローカル AP から目的地まで徒歩で移動する。このような場合における提案システムの有効性を検証する。

6.2 評価方法

提案システムの有効性を評価するために実験を行う。以下に実験内容についてまとめる。

評価対象

- 通常の OpenStreetMap

Focus 生成機能を持たず、全体を同一縮尺で表示する

- 従来の Focus+Glue+Context マップ

円形等の単純形状の Focus のみを生成する Focus+Glue+Context マップ

- 提案システム

交通機関のアクセスポイントを考慮し、目的地に合わせた Focus + Glue + Context マップを生成する

実験内容

被験者は上記の評価対象である 3 つの Web マップシステムを使用してタスクを行う。その後、ユーザビリティに関するアンケートに回答する。以下に、被験者が行うタスクの詳細な手順を述べる。

- step1** 使用する 3 つの Web マップシステムの使用方法に関する説明を受け、使い方を理解する
- step2** こちらで予め出発地点と目的地を設定する。被験者は事前に配布された名古屋市内の路線図(図 13)と Web マップシステムを用いて、出発地点から病院(何の病院でも構わない)を経由して目的地へ行く経路を決める
- step3** 出発地点から目的地に行くのに使用する路線とローカル AP、経由する病院名、ローカル AP から目的地までの歩行経路を紙に図示する

目的地点を変えて、このタスクを計 2 回行う。

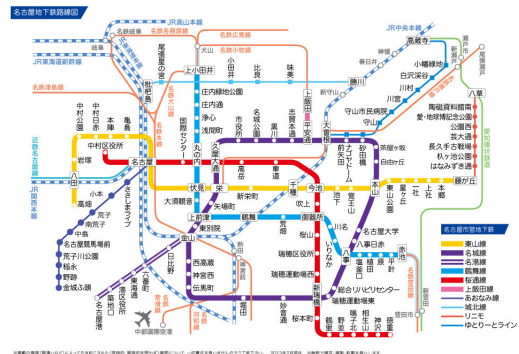


図 13 配布した路線図

評価尺度

- タスク完了までに費やした作業時間
- タスク完了までの操作回数
 - 縮尺の変更(2段階変更した場合も1回と数える)
 - 地図表示範囲の移動
 - Focus の拡張、縮小

ユーザビリティ(アンケート)

アンケートは 5 段階で行い、1(全く当てはまらない)~5(非常に当てはまる)のうち、最も当てはまるものを選択してもらう形式とした。アンケートの項目を以下に示す。

- 使いやすかった
- ストレスを感じなかった
- 歩行経路の確認を行いやすかった
- 路線経路の確認を行いやすかった
- 使っていて楽しいと感じた
- これからも使いたいと感じた
- 地図オブジェクト情報が見やすかった

被 験 者

大学生 5 名

6.3 結 果

タスク完了までの操作回数の平均を表 2 に、タスク完了までに費やした作業時間の平均を表 3 に、アンケート結果のグラフを図 14 に示す。

表 2 操作回数の比較

	縮尺の変更	地図範囲の移動	Focus の拡張, 縮小	操作回数合計
通常の OpenStreetMap	8.3	4.6	0	12.9
従来の F+G+C マップ	3.3	8.2	5.3	16.8
提案システム	3.1	2.8	1.2	7.1

表 3 作 業 時 間

	作業時間 (秒)
通常の OpenStreetMap	253
従来の F+G+C マップ	416
提案システム	234

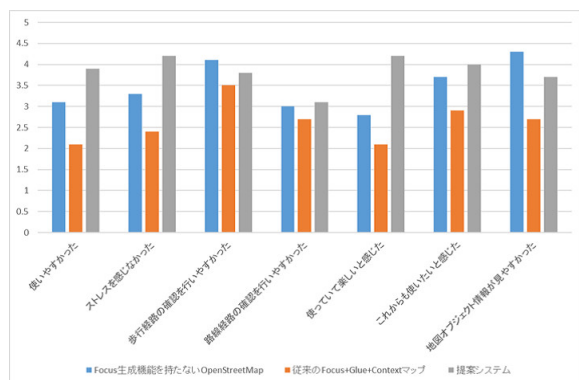


図 14 アンケート結果

6.4 考 察

操作回数の平均は、提案システムが最も低く、従来の Focus + Glue+Context マップシステムが最も多いという結果になった。通常の OpenStreetMap は画面内の地図全てが同一の縮尺で描画されるため、歩行経路と路線経路の確認を同時に行うことが出来ない。そのため、縮尺の変更を行う回数が多くなっている。従来の Focus + Glue+Context マップで生成される円形 Focus の大きさはデフォルトで決まっている。その大きさが今回のタスクを行うには小さすぎたので、被験者は Focus の大きさを拡張したり、Focus の表示範囲を何回も動かす必要があった。そのため、従来の Focus + Glue+Context マップシステムの操作回数が 3 システムの中で最も多くなったと考えられる。提案システムは、主要 AP を画面内に含むよう Context の補正を行うので地図範囲の移動を行う必要がない。Focus に関しても、指定した地点に合わせた形状をシステム側で算出するため、従来の Focus+Glue+Context マップより Focus の拡張、縮小を行う回数が少ないことが確認できる。

作業時間の平均を見ると、通常の OpenStreetMap と提案シ

ステムが同程度の時間で、従来の Focus+Glue+Context マップのみ長くかかるという結果になった。Focus の表示範囲の拡張を行うには、Glue 表示範囲と Focus 表示範囲それぞれに対して拡張の操作をする必要があり、その操作に時間がかかるため、Focus の拡張、縮小の操作回数が多い従来の Focus+Glue+Context マップの作業時間が長くなってしまったと考えられる。提案システムでは指定地点から徒歩圏内のローカル AP を含むように Focus を生成するが、生成した Focus が病院を 1 つも含まない場合、Focus の表示範囲を拡張する必要があるため作業時間が増加した。しかし、それを踏まえても、3 システムの中で最も作業時間の平均が短かったため、提案システムは従来の Focus+Glue+Context マップが持つ「操作に時間を要する」という問題を改善しているといえる。

次に、アンケート結果について述べる。グラフの縦軸は、各項目に当てはまる度合いを表し、数値が高いほどその項目に当てはまっていることを示す。提案システムは「使いやすかった」、「使っていて楽しいと感じた」、「これからも使いたいと感じた」の使用感に関する項目において他の 2 システムより良い評価を得ている。少ない操作回数で、ユーザーの求める情報を提供していることが、使用感の良さに繋がっていると考えられる。

また、「地図オブジェクト情報が見やすかった」という項目において、通常の OpenStreetMap が最も良い評価を得た。通常の OpenStreetMap は画面内の縮尺が全て一定で表示されるため、詳細な歩行経路を知りたい場合は画面全体の縮尺を大きくする。そのため、画面全体から詳細な地図オブジェクト情報を多く得ることができ、画面の一部分のみを拡大する従来の Focus+Glue+Context マップや提案システムより良い評価になったと考えられる。

また、従来の Focus+Glue+Context マップで生成される円形 Focus は、Focus 内に歩行経路を確認するのに不必要な領域も多く含む。そのため、提案システムより Glue に圧縮される領域が大きい。提案システムは、形状の算出により、無駄な領域を含まない必要最低限の大きさの Focus を生成している。従来の Focus+Glue+Context マップと提案システムで「地図オブジェクト情報が見やすかった」の項目を比較すると提案システムのほうが良い評価を得ていることから、提案システムは Glue に圧縮されて失われる地図オブジェクト情報の量を従来の Focus+Glue+Context マップより抑えているということがわかる。よって、提案システムは従来の Focus+Glue+Context マップが持つ「地図オブジェクト情報の減少」という問題を改善しているといえる。

今回の実験は Web マップシステムと路線図を用いて行ったが、地図上での路線の確認が難しいというコメントが被験者から得られた。OpenStreetMap の地図データでは JR などの地上を走る鉄道の路線は常に地図上に描画されているが、地下鉄の場合は駅のみが地図に描画されており、路線は縮尺をかなり大きくしないと描画されない。駅だけではどの路線がどの駅を通るのか判断できないため、Web マップだけで経路確認を完結させるには、地下鉄等の路線も地図上に描画する必要があると感じた。また、路線経路を決める際、地図からは得られない情

報が欲しい時がある。地図を見ることで、使用する路線や乗換駅を知ることができるが、運賃や所要時間等の情報は地図からは得られない。1つのシステムで歩行経路と路線経路の確認を完結させるには、乗り換え情報を詳しく検索できる機能を設ける必要があると感じた。また、提案システムでは、Contextは主要APを含み、かつできるだけ縮尺が大きくなるよう補正を行ったが、図15のように、Contextから路線経路が外れてしまうことがあった。このような状況では路線経路の決定を正しく行うことができないため、路線経路もContext内に含めるよう改善する必要がある。

また、今回の実験では想定よりも多くの経路が被験者によって決定された。病院という経由地を設けたことで、ただ最短経路を求めるのではなく様々なローカルAPを使うことが可能になったため、経路の選択枝に幅が生まれたと考えられる。被験者によって何に重きを置くかが異なり、乗り換え回数が少なくなるようルートを選択する者もいれば、歩行距離が最も短くなるルートを選択する者もいた。ユーザーによって重きを置く条件が異なるため、Focus内に含める領域についても、ユーザーの好みに合わせて選択することができればより良いシステムになると感じた。

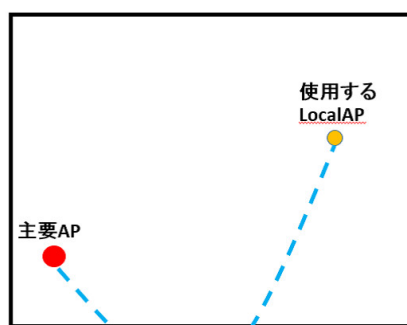


図 15 路線経路に関する問題

7. 終わりに

本稿では、ユーザーの目的地と交通機関のアクセスポイントを考慮した Focus+Glue+Context マップシステムを提案し、その実現法について述べた。ユーザーが指定した目的地に合わせて、Focusの形状、Contextの縮尺、Contextの表示範囲を決めることで、主要APからローカルAPまでの路線経路と、ローカルAPから目的地までの歩行経路を同一画面上で1度に閲覧することが可能な地図を生成した。また、提案システムに基づいたプロトタイプシステムを実装し、評価実験を行った。実験結果により、提案システムは従来システムより約44%少ない操作時間と約58%少ない操作回数で、地図利用者の要求に応じた Focus + Glue+Context マップを生成することを示した。そして、従来の Focus+Glue+Context マップシステムの持つ課題である、地図オブジェクト情報の減少を抑えることができた。今後の課題として、地図上への路線の描画がある。地図上に路線を描画することで、路線経路の把握がより容易に行うこ

とが可能になると考えられる。

謝 辞

本研究は JSPS 科研費 26330136, 25700009 および、科学技術振興機構 CREST の助成を受けたものです。

文 献

- [1] 小関章太郎, 高橋直久, focus+glue+context マップの glue 領域における道路ネットワーク総描法, 第6回情報科学技術フォーラム講演論文集, D-025(2007)
- [2] Naohisa. Takahashi, An Elastic Map System with Cognitive Map-Based Operations, International Perspectives on Maps and Internet Vol.1, M. P. Peterson and J. Liu (Eds), Lecture Notes in Geoinformation and Cartography, Springer-Verlag, 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, pp.101-110, 2009.
- [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, pp. 1914-1925, 2010.
- [5] 村瀬正樹, 山本大介, 高橋直久, Focus+Glue+Context マップのためのストロークに基づく道路ネットワーク総描システム, DEIM Forum 2016 H7-3.
- [6] George W. Furnas, Generalized Fisheye Views, CHI'86, pp. 16-23, 1986.
- [7] Manojit Sarkar, Marc H. Brown, Graphical Fisheye views of graphs, CHI '92 Conference Proceedings, pp. 83-91, 1992.
- [8] 加藤史也, 山本大介, 高橋直久, 地図オブジェクトの形状を考慮した Focus 生成システム EMMA テムの実現, DEIM Forum 2014 E9-1.
- [9] Leaflet
<http://leafletjs.com/>
- [10] OpenStreetMap
<http://www.openstreetmap.org/>
- [11] PostGIS
<http://postgis.refractory.net/>
- [12] 駅.jp
<http://www.ekidata.jp/>