平成16年度

筑波大学第三学群情報学類

卒業研究論文

題目 <u>地図に付随する注釈情報を動的に表示する</u> <u>インタフェースの研究</u>

主専攻 情報科学主専攻

著者 大谷裕昭

指導教員 田中二郎・志築文太郎

要 旨

日常我々は、カーナビゲーションシステムなどの GIS(geographical information system)を使う機会にしばしば遭遇する。

しかし、それらのシステムに使われている地図では建物等の情報は文字ラベルやアイコンで 示されているだけであり、その建物等についての詳細を地図上から確認することは難しい。そ こで、我々は地図に注釈情報を付加し、注釈情報を地図上へレイアウトする為のインタフェー スについて提案、実装を行った。実装されたシステムでは、地図上の道をユーザが自分で選 択することにより、その道の周りの注釈情報がレイアウトされる。地図に注釈情報を付加し、 それをレイアウトすることにより、地図を一覧するだけで前よりも多くの情報を得る事が可 能になった。

目 次

第1章	はじめに	1
第2章	既存地図の表示	3
2.1	情報の表示	3
2.2	詳細情報の表示	3
2.3	注釈情報付き地図の提案・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	3
第3章	注釈情報付き地図の表示	4
3.1	レイアウト場所の設計	4
3.2	処理	4
	3.2.1 レイアウト位置の処理	4
	3.2.2 ポイントの決定	6
	3.2.3 線の交差の除去	6
	3.2.4 配置変更の可視化	7
3.3	機能	8
	3.3.1 情報の追加	8
	3.3.2 情報の削除	9
	3.3.3 情報の変更	9
	3.3.4 情報の保存	11
	3.3.5 経路選択	11
	3.3.6 経路選択後の表示の変更	11
	3.3.7 範囲選択	13
第4章	システムの実装	14
4.1	言語、環境	14
4.2	それぞれのクラスの実装	14
	4.2.1 地図を表示するクラス	14
	4.2.2 注釈情報に関するクラス	14
	4.2.3 キャンバスの表示に関するクラス	16
	4.2.4 ポップアップ、ダイアログに関するクラス	19
第5章	アプリケーション例	20
5.1	電子的な携帯ガイドマップ.................................	20

5.2	旅行を計画するための地図.............................	20
第 6 章 6.1	まとめ 今後の展望	22 22
	謝辞	23
	参考文献	24

図目次

3.1	システム画面の縁...................................	5
3.2	経路沿い	5
3.3	経路沿い以外	6
3.4	初期状態....................................	7
3.5	地図のドラッグ	7
3.6	再レイアウト中	8
3.7	レイアウト完了	8
3.8	情報追加ダイアログ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	8
3.9	ポップアップの表示	9
3.10	削除後のレイアウト	9
3.11	ポップアップの表示	9
3.12	ダイアログ	10
3.13	変更前の表示	10
3.14	変更した情報を入力したダイアログ	10
3.15	変更後の表示	10
3.16	リストの保存	11
3.17	経路選択	12
3.18	経路選択後のレイアウト	12
3.19	見にくい例	12
3.20	移動前	13
3.21	移動後	13
3.22	範囲選択....................................	13
3.23	範囲選択後のレイアウト	13
4.1	サークルの描画	15
4.2	アニメーションの描画	15
4.3	情報の追加	16
4.4		17
4.5	情報の削除	17
4.6	経路・範囲選択	18
4.7	n 角形 (閉じた図形) の例	18
4.8	n 角形ではない例	18

4.9	サークルの描画	19
5.1	ガイドマップ利用イメージ................................	21
5.2	旅行の計画イメージ	21

第1章 はじめに

我々は、車やインターネットの普及により、カーナビゲーションシステムや、マピオン[1]、 グリーンマップ[2]のような地理情報システム(GIS:Geographical Information System)を多数 見かけるようになった。さらに現在では、GISを使った様々なアプリケーションが開発され ている[3][4][5][6][7]。

しかし、このようなシステムに使われている地図は以下の点において問題がある。

情報の表示方法

カキコまっぷ[6]における情報は文字ラベルを用いて表示されるが、このような表示方法であると文字ラベルが重なってしまう状態が多くみられる。また、場log[7]における情報は写真で示されているが、これでは地図上の写真同士が重なってしまい、せっかくの情報が台無しになってしまう。

一覧した時得られる情報量

カキコまっぷのようなものであれば、地図を一覧して得られるのは文字ラベルやアイ コンのある位置にどのような名前のものがあるのかという情報しかつかめない、また WebGIS を用いたバリアフリーマップ[8]のようなものであると、何処に障害があるの かという情報しか分からない。

詳細情報の表示方法

また詳細情報は、紙に書いてある地図では書いてないものも多く、書いてあるとしても 別のページにしか記載されていない。また電子的な地図であっても詳細情報は別ウィン ドウで表示という地図が多く見られる。

本研究では GIS に使われる地図に注釈情報を付加し、その注釈情報をレイアウトするこ とで問題を解決しようと試みている。地図上に注釈情報をレイアウトしている研究は土屋ら により行われている [9] が、そのレイアウト手法は手動、静的な方法である。このようなレイ アウト方法の場合、レイアウトすべき注釈情報が多数あった場合非常に労力がいる。また、静 的なレイアウト方法であるとユーザが見たいと思っていない注釈情報まで表示されてしまう ため、ユーザが見たいと欲している注釈情報を探すのに時間がかかる。このようなシステム の地図の場合、注釈情報を探すのに急いでいない状況ならば問題はないが、早急に探し出し たいといった状況の場合、注釈情報を探すのに時間がかかるため不便である。そこで、ユー ザの手を煩わせず地図上で注釈情報を閲覧出来るようなシステムが必要であると考えられる。 また、そのために注釈情報を見易く配置するべきであるということも考えられる。本研究で は PC 内の処理により自動で注釈情報がレイアウトされ、さらにユーザの状況によって動的に 注釈情報をレイアウトしている。自動でレイアウトされることでユーザ側のレイアウト時の 負担が減り、また動的にレイアウトすることで、ユーザが欲している注釈情報をレイアウト することが可能になる。

本論文の構成は2章で既存の地図の表示方法の問題の記述、3章では注釈情報付き地図を 用いて作ったシステムについての説明、4章でシステムの実装、5章でシステムのアプリケー ション例を述べ最後6章でまとめを記述し今後の展望を述べる。

第2章 既存地図の表示

この章では、既存のシステムに使われている地図の表示方法とその問題点について述べる

2.1 情報の表示

既存のシステムに使われている地図における情報の表示方法は文字ラベルやアイコンで表 されているものが多い。しかしこれらの表示方法によって地図上から得られることは、その 位置にあるものがどのような属性であるかという情報だけであり、その位置にあるものがど のような雰囲気なのか・何が売っているところなのかという詳細情報を読み取ることは出来 ない。

2.2 詳細情報の表示

紙に描かれている地図では詳細情報を得ることは出来ない。しかし、WebGIS などの電子 的な地図であれば、アイコンやラベルをクリックすることで、詳細情報が記されているウィ ンドウが別に現れ、それを見てユーザは詳細情報を確認することが出来る。ただ、別ウィン ドウが出るような詳細情報の表示方法であると、複数の情報を見比べることが難しい。また、 コンピュータの画面上に並べて見比べたとしても、見終わった時にウィンドウを多数閉じな ければならないのは煩わしい作業である。

2.3 注釈情報付き地図の提案

上記のような問題を解決するため、我々は注釈情報付きの地図を使い、それを表示するため のインタフェースについて研究を行った。注釈情報は最初データとして地図に埋め込まれてい る形で用いて、システムに読み込まれた時には、注釈情報は閲覧出来ないようにする。ユー ザからの入力を受け取ることで注釈情報が表示される。本研究では注釈情報として写真を利 用している。写真を使うことでその場所の雰囲気を地図上から得ることが出来る、また地図 を一覧するだけでラベルやアイコンのある位置の情報を前より多く得ることができ、さらに は注釈情報を写真以外にすることでも地図上でより詳しい情報を見比べる事も可能になる。

第3章 注釈情報付き地図の表示

この章では、注釈情報を付加した地図を用いて実装したシステムについて述べる。 単に注釈情報を文字ラベルやアイコンのように置くだけでは、地図は逆に見にくくなって しまう。それを解決するため注釈情報をユーザが見やすいようにレイアウトした。

レイアウトする場所、レイアウトするにあたっての規則や工夫などを考え、それに従って レイアウトに関する実装を行った。

3.1 レイアウト場所の設計

図 3.1 のように注釈情報が重なることなく、また対応関係が一目でわかるようなレイアウト 場所として、システムウィンドウの縁に注釈情報をレイアウトするようにした。レイアウト する場所は図 3.1 以外に図 3.2 のような自分が通ろうとしている道の道沿い、図 3.3 のような 道沿い以外を考えた。図 3.2、図 3.3 における、通ろうとしている道は図中の S の字の青色の 太い破線の事を示している。図 3.2 を見ると注釈情報同士が重なってしまい、さらには対応関 係が非常に分かりにくい。図 3.2 は注釈情報の重なりというものがないが、レイアウトする時 に情報が存在する位置を覆い隠してしまい、やはり注釈情報と情報の存在する位置との対応 関係が確認しにくい。

3.2 処理

3.2.1 レイアウト位置の処理

次に、システムウィンドウの縁のどの位置に注釈情報をレイアウトするのかを述べる。 まず、注釈情報の大きさ(160*160)と同じスペースを確保する。次に、システムウィンドウ の大きさにあわせて縦横共にどれだけそのスペースが入るかを計算する。計算をするときに、 なるべく多くのスペースを入れられるようにするため、2割ほどのマージンを設けて計算を する。システムウィンドウの大きさ、先ほどの計算結果、スペース同士の間隔を定め、それ らに合わせてスペースを縮小する。そして、縮小されたスペースをシステムウィンドウに並 べたときのスペースの中心がそれぞれ注釈情報がレイアウトされる位置(以下 ポイント)に なっている。



図 3.1: システム画面の縁



図 3.2: 経路沿い



図 3.3: 経路沿い以外

3.2.2 ポイントの決定

次に、各注釈情報のポイントの決定の仕方について述べる。

それぞれの注釈情報に対応する情報の位置(以下 サークル)から各ポイントへの距離を計 算し、短い順にレイアウトを行っている。これはサークルとポイントがなるべく近いほうが 見やすいのではないかという考えである。もし、ポイントが重なってしまった場合、後にポイ ントを決定したほうは、次に最も近いポイントへとレイアウトされる。そうすることで、注 釈情報同士が同じポイントにレイアウトされるケースがなくなる。また、スライダーを動か す・地図の大きさを変更する・地図をドラッグするなどをユーザが行うと、ポイントの決定 を再び行い注釈情報をレイアウトする。

3.2.3 線の交差の除去

以上までの処理が終わったら、次に注釈情報とサークルとを結ぶ線(以下 線)に関する処 理を行う。

注釈情報とサークルの対応関係をさらに明確にするためには線の交差が無ければ、よりわ かりやすいのではないかと考えた。これは、証明等を行っていないので実際に出来るかどう かは未確認だが、以下の方法で線の交差の除去を実装した。まず、線の交差をチェックし、交 差があったのならばポイントを入れ替えてキャンバスに表示させる。ここで注意することは、 入れ替えたときに別の線と交差している可能性があるので、総当りで交差のチェックを行い ポイントを入れ替えるということを繰り返し行っている。これも、ポイントの決定と同じく、 ユーザがスライダーを動かす、地図の大きさを変更する、地図をドラッグするなどを行うと、 ポイントの決定をしてから、線の交差の判定・除去が行われ、注釈情報が再びキャンバスへ レイアウトされる。

3.2.4 配置変更の可視化

ポイントの決定、線の交差の除去を再び行い注釈情報を表示させるとき、元あった場所か ら次の場所へとポイントが移ってしまった場合、その変化を確認するのは難しい。そこで、そ の変化をアニメーションにして可視化することで、ユーザは前に見ていた注釈情報が何処に 移ったのかを簡単に把握することが出来るようになる。

以下に図で説明を加える。図 3.4 が注釈情報をレイアウトした図で、これを初期状態とす る。次に図 3.5 がマウスをドラッグし地図をドラッグしている状態で、再レイアウトを行う前 の状態である。この状態でマウスボタンを離すとポイントの決定・線の交差の除去が行われ、 レイアウトが始まる。図 3.6 は再レイアウト中の状態でアニメーションの途中をあらわしてい る。図 3.7 が再レイアウト完了の状態である。



図 3.4: 初期状態

図 3.5: 地図のドラッグ



図 3.6: 再レイアウト中

図 3.7: レイアウト完了

3.3 機能

3.3.1 情報の追加

ユーザがこのシステムを使っている時に新たに情報を発見した場合、地図上への情報の追 加が必要になる。そこで情報を追加するための機能を実装した。

システム内の地図上でマウスの左ボタンをダブルクリックすると図 3.8 のように情報を追加す るためのダイアログが表示される。そのダイアログに、タイトル、コメント、注釈情報を入 力することで地図上に情報が追加される。追加された情報は、経路選択 (3.2.5 参照)、範囲選 択 (3.2.6 参照)をすることでレイアウトされる。



図 3.8: 情報追加ダイアログ

3.3.2 情報の削除

元に情報があった場所に情報が無くなってしまった場合、ユーザは地図上の情報の削除が 必要になる。そこで情報の削除機能を実装した。

情報上で右クリックをすると図 3.9 のようにポップアップメニューが表示される。そこで delete をクリックすることで、その情報が削除される。図 3.9 の操作で情報が削除されて再び レイアウトが行われた後の状態を図 3.10 に示す。

3.3.3 情報の変更

元に情報があった場所の情報が変更されていた場合、地図上の情報の差し替えが必要になる。そこで情報の変更機能を実装した。

情報の上で右クリックをすると図 3.11 のようにポップアップメニューが表示される。そこ で change をクリックすることで、図 3.12 のようなの情報を変更するためのダイアログが開 く。このダイアログに表示されているのは元から注釈情報として入力されていたデータであ る。図 3.13 に情報変更される前の状態を示す。情報を変更するためには図 3.14 のように以前 とは違ったデータを入力すればよい。図 3.15 に情報変更後の状態を示す。



図 3.9: ポップアップの表示

図 3.10: 削除後のレイアウト



図 3.11: ポップアップの表示



図 3.12: ダイアログ

アログ

図 3.13: 変更前の表示



図 3.15: 変更後の表示

3.3.4 情報の保存

上記の操作を行っただけでは次回実行時に変更が適用されないので、この操作が必要になる。

システムメニューバーの File をクリックするとダウンメニューで listsave が表示される。それ をクリックすることで上記の変更が保存され、次回プログラム実行時に変更が適用される。



図 3.16: リストの保存

3.3.5 経路選択

地図上に付加された注釈情報全てを見易くレイアウトすることは困難であると考えられる。 また、ユーザが全ての情報に対する注釈情報を把握、閲覧することは考えにくい。つまりユー ザが見たいと思う情報に対する注釈情報だけをレイアウトすればよい。ユーザが地図上で見 たいと思う情報は自分が通る道などの周りの情報であると考えた。そこで、そのための経路 を選択する機能を実装した。

システムの地図上で右ドラッグをすると図 3.17 のように青い破線が表示される。この線を 用いてユーザは自分が通る予定である経路を選択し、その破線の周りにある情報の注釈情報が レイアウトされる。マウスボタンを離すと、経路選択は終了し図 3.18 のようにシステムウィ ンドウの縁に注釈情報がレイアウトされる。

3.3.6 経路選択後の表示の変更

経路を選択した時、その選択した場所が図 3.19 のようにシステムウィンドウの縁に近い場 合非常に見難くなってしまう。

そこで、選択した経路をシステムウィンドウの中央へ移してから注釈情報をレイアウトし ようと考えた。図 3.20 を見ると、薄い赤色の円で囲まれた内部に経路選択をする青い破線が 表示されている。その状態からマウスボタンを離した後の状態を図 3.21 に示す。今度は、薄



図 3.17: 経路選択

図 3.18: 経路選択後のレイアウト



図 3.19: 見にくい例

い青色の円で囲まれた内部に先ほど選択した経路が表示されている。図 3.19 と見比べてみる と明らかにこちらのほうが見やすいということがわかる。図 3.21 の地図の中心は選択した経 路の重心である。



図 3.20: 移動前

図 3.21: 移動後

3.3.7 範囲選択

上記に加え、ユーザが地図上で見たいと思う情報は自分の活動範囲内の情報も調べたいこ ともあるのではないかと考えた。そこで、範囲選択を可能にする機能を実装した。

上記の破線を閉じた図形にすることで実行される。破線の周囲にある情報ではなく、閉じ た図形の内部にある情報の注釈情報もレイアウトされる。



図 3.22: 範囲選択

図 3.23: 範囲選択後のレイアウト

第4章 システムの実装

この章ではシステムの細かい実装について述べる。

4.1 言語、環境

システムに使用した言語は Java2SDK ver1.4.1_06。使用した地図は国土地理院が発行して いる数値地図 2500(空間データ基盤)を使用している。動作テストは OS に WindowsXP をプ ロセッサに pentium4 を CPU が 2.53GH z の PC を 1 台と OS に WindowsXP、プロセッサに pentium4、CPU が 2.00GHz の PC を 1 台の計 2 台で行った。

4.2 それぞれのクラスの実装

本システムでは Road クラス、PartOf Maps クラス、ListItem クラス、ListItem Mover クラス、 ListDialog クラス、LLXYConverter クラス、MyCanvas クラス、Popup クラスの計 8 つのクラ スで構成されている。

4.2.1 地図を表示するクラス

地図の表示は PartOfMaps クラスで行われている。PartOfMaps クラスではキャンバス上への地図の位置の決定を行っている。

使用地図をそのまま表示すると x 座標・y 座標が逆であるので、その変換が必要になる。つま り普通なら x 座標が左右、y 座標が上下という関係になっているが、使用地図では、x 座標が 上下・y 座標が左右という形で作られている。

4.2.2 注釈情報に関するクラス

注釈情報の表示、レイアウトに関するクラスは ListItem クラス、ListItemMover クラス、 LLXYConverter で行っている。ListItem クラスでは、地図上へのサークルの位置の決定、ラ ベルの表示、アニメーションの表示を行っている。ListItemMover クラスでは、注釈情報のレ イアウト位置の決定、線の交差の除去を行っている。LLXYConverter クラスでは、世界測地 系から直交座標系への変換を行っている。

地図上へのサークルの位置の決定は、MyCanvas クラス、LLXYConverter クラス、ListItem

クラスによって決定される。本システムでの情報の保存は[情報 ID、タイトル、コメント、緯 度、経度、注釈情報]という形で保存されている。位置の決定は緯度・経度が使われ、これら を LLXYConverter クラスに渡し地図上の直行座標系に変換、その座標情報を ListItem クラス が受け取り MyCanvas クラスで実際にペイントする。ラベルの表示方法は、まずタイトルの 長さを計算し、その長さに見合った四角形の領域を作り、その中にタイトルを書き込む。ラ ベルの初期位置はサークルの右隣に表示される。アニメーションの表示では、ListItemMover クラスで元に注釈情報があった座標と、再レイアウトされたときに決定されたポイントの座 標を ListItem クラスに渡し、ListItem クラス内でその座標の差分をとり、少しずつずらしてア ニメーション表示させている。



図 4.1: サークルの描画



図 4.2: アニメーションの描画

4.2.3 キャンバスの表示に関するクラス

キャンバスの表示に関するクラスは MyCanvas クラスで行っている。MyCanvas クラスでは、 情報の追加・変更・削除などの細かい機能や、注釈情報、サークル、ラベルなどの描画を実 際に行っている。

情報の追加では、情報追加ダイアログ(3章参照)でタイトル、コメント、注釈情報を入力 し、OKボタンを押すと、保存されている情報の一番最後に追加されるように情報 ID を付随 し、一時ベクトルへ保存される。すると、追加された情報が repaint される。(この時、3章で も書いてある通り保存をしないと次回に反映されない。)情報の削除は情報が保存されている ベクトルからその要素だけを削除し repaint を行っている。情報の変更は情報変更ダイアログ に変更したい部分を入力し OK を押せば repaint されるようになっている。情報の保存は所定 のファイルに [ID、タイトル、コメント、緯度、経度、注釈情報]を書き込むように実装をし ている。

経路選択に関する機能はマウスイベントを使い実装している。まず、mousePressed が呼び 出されたらその座標を保存、そして mouseDragged が呼び出されたらそのつど座標を保存し てということを繰り返して経路選択用の青い破線が描画される。青い破線の周囲の情報に関 する注釈情報が表示されるようにしているが、実世界で 50m 以内のものについて表示される ようになっている。実装の仕方については、経路を選択していき、その範囲内に入ったもの から順にベクトルに保存され、そのベクトルを ListItemMover クラスへ渡しそのなかで表示さ れる位置が決定される。範囲選択の実装は青い破線が閉じた図形になっていた場合、それを 多角形としてとらえて実装している。つまり API の Polygon クラスを使い、破線の状態が図 4.7 のような n 角形の図形として成り立っていたら、その n 角形内部にある情報の注釈情報を ベクトルに保存し、後は経路選択した時の処理と同じことをする。選択された経路を画面の 中央に持っていくための実装は、破線の重心を求めその座標をシステムウィンドウの中心に アニメーションで移動させるという実装方法で行っている。



図 4.3: 情報の追加



図 4.4: 情報の変更



図 4.5: 情報の削除



図 4.6: 経路・範囲選択



図 4.7: n 角形 (閉じた図形) の例



図 4.8: n 角形ではない例

4.2.4 ポップアップ、ダイアログに関するクラス

情報の削除・追加・変更などに使うポップアップやダイアログは Popup クラス、ListDialog クラスで実装されている。Popup クラスでは、ポップアップメニューのどのメニューが選ば れたかなどを判定しその時の処理を行う。ListDialog クラスでは、フィルタの設定など簡単な ダイアログ表示の実装を行っている。



図 4.9: サークルの描画

第5章 アプリケーション例

ここではシステムのアプリケーション例について記述する

5.1 電子的な携帯ガイドマップ

身近に見る案内地図などは、2章でも記述したとおり、パンフレットなどに地図が載ってい てその地図上にアイコンや文字で建物等の名称を記述、そして別ページにそれらの詳細情報 などを記述している。こういったものでは、詳細情報と位置との対応関係、詳細情報の確認 などを何度もページをめくって確認しなくてはならないので面倒である。

しかし、このシステムを PDA やノート PC などに移し、GPS と組み合わせれば今歩いてい る場所の近くにあるものの注釈情報が表示されるので対応関係や詳細情報の確認を何度も見 返すという必要がなくなってくる。

例えば、初めて学校に訪れる人に学内を案内する場合、その人は学内の情報に関しては一 切知らない。そこで情報を知る手段が先に挙げたパンフレットのようなものであった場合、ま ず自分の位置を把握することが困難であるためパンフレット内の情報を頼りに自分の位置を 把握しないといけない。自分の位置を把握出来て、目的の場所に辿り着けたとすると、今度 は詳細情報を確認するために別ページを開かなければならない。しかし、このシステムを使 い、GPS を組み合わせることで、自分の位置がすぐ把握出来るため目的の場所へスムーズに 移動が可能になり、また、一画面で詳細情報が確認出来るので、ページをめくるなどの煩わ しい作業がなくなると考えられる。

5.2 旅行を計画するための地図

我々は旅行の計画を立てる時、ロードマップなどで経路に関して計画を立て、Web 上など で、その経路沿いの情報について調べ何処で食事をするか・何処で遊ぶかということをを決 定する。しかし、これでは経路沿いの情報について一つ一つ調べ、またそれに関して情報を 吟味していかなければいけない。しかし、このシステムを用いれば、経路を選択するだけで 周りの情報の注釈情報が表示されるため、一つ一つ調べ吟味する、という行為が非常に楽に なると考えられる。

例えば、A 地点から B 地点への旅行を計画し、経路の設定も済んだ場合、その道の途中で 食事がしたいと思いレストランがあるかどうかを調べたら 10 件のレストランが見つかったと する。旅行を計画する人は、この 10 件について色々と調べる必要がある。その情報について



図 5.1: ガイドマップ利用イメージ

1件1件メモしながら情報を見比べるのは労力がいる。このシステムを導入すれば、すぐにその10件の詳細情報が確認でき、しかも1画面に表示されるので、その労力は非常に少なくてすむと考えられる。



図 5.2: 旅行の計画イメージ

第6章 まとめ

本研究では、既存の地図の表示方法では地図を一覧しただけでは詳細情報が読み取りにく い、という問題点を挙げた。そして、この問題を解決するために注釈情報付きの地図の提案 を行い、これを使ったシステムの実装について述べた。また、このシステムを使ったアプリ ケーション例についても述べた。

6.1 今後の展望

今後の展望としては、サークルが一箇所に集まっていたとき注釈情報とサークルの対応関 係がわかりづらいのでその処理をどうするか、また、システムウィンドウの縁にレイアウトす る注釈情報の数が許容量を超えたときその処理を実装していないのでどのようにするか、と いうことを考えていきたい。

謝辞

本稿作成にあたり、ご指導くださった田中二郎教授、ならびに三末和男助教授、志築文太郎講師、高橋伸講師には心から感謝致します。

また田中研究室のみなさんにはゼミ、ミーティングを通じ貴重なご意見を頂きました。特にSWATチームのリーダー佐藤大介さんには自身がお忙しい中、議論や指導をしていただき 色々な意見を頂きました。ここに感謝の意を表します。

参考文献

- [1] マピオン: http://www.mapion.co.jp/
- [2] グリーンマップ: http://www.greenmap.jp/
- [3] 株式会社ジャスミンソフト: ケータイ日記 地図連動型電子日記システム http://w3land.mlit.go.jp/nrpb-gisbox/dl/jasminesoft/index.html
- [4] 沖電気工業株式会社: デジタルアルバム (整理箱) http://w3land.mlit.go.jp/nrpb-gisbox/dl/oki/ksj.html
- [5] 社団法人 日本リサーチ研究所地図ぽん 道路情報による地域コミュニケーションシス テムhttp://w3land.mlit.go.jp/nrpb-gisbox/dl/j-ric/index.html
- [6] 上田紀之, 中西泰人, 真鍋陸太郎, 本江正茂, 松川昌平: 時空間ポエマー + カキコまっぷ -GPS カメラケータイを用いた WebGIS の構築 - 電子情報通信学会 第6回 S-IT ワークショッ プ,2003
- [7] 上松 大輝, 徳永 徹郎, 沼 晃介: 場 log: 位置情報に基づく情報整理意システム, インタ ラクション 2004 論文集,pp.73-74,2004
- [8] 硴崎研究室: WebGIS によるバリアフリーマップの研究, 国土交通省、国土計画局のGI Sモデル地区実証実験参加者からの最終報告書,2002
- [9] 土屋 雅人: GIS を用いたポータルサイトのインタフェース研究, 情報処理学会研究報告,2004-HI-111,pp.49-54,2004