

一筆圃場管理のためのマッピングシステム

小林 一*・酒井美幸**

平成7年6月23日受付

The Mapping System for the Computer System on a Field Management of Paddy Farm

Hajime KOBAYASHI* and Yoshiyuki SAKAI**

The large-scale paddy farm's managers need to apply the computer system to the information management about a rice field, because they cultivate a lot of rice field under their control. But existing softwares have some problems about operation and price, so they don't spread regularly. Therefore in this research we developed the mapping system using Windows of a personal computer which is a part of the computer system on a rice field management of paddy farm. Firstly this system reads punctual coordinates from the original drawing of a farm picture, inputs them to a computer, and draws their blank map. Continuously this completes a wide farm picture which can manage many rice fields more widely by uniting plural blank maps. Thus we can visually indicate a rice field information according as we link pictures with the farm database. We can try to manage a rice field information efficiently and support to make a work planning by use of this system.

緒 言

水田作経営において、集团的土地利用を推進して生産性の高い農業経営の確立をはかるためには、もっとも基幹的な生産基盤となる一筆単位の圃場に関する情報を適切に管理することが重要である。とくに、大規模水田作経営にとっては、農用地の利用権集積や実際の利用をめぐって、わが国の土地所有構造の特徴である零細分散錯圃制に由来する諸条件からの規制が強く作用している。そのため、土地利用の基礎単位となる一筆ごとの圃場に

関する権利や利用状況について、基本的な情報を収集・整理し、水田高度利用の推進に向けて、それを農家間での農用地利用調整や、経営の運営管理に活用していくことが課題となる。

ところが現実には、耕作規模が20から30haを上回るような大規模な水田作経営になると、対象にしなければならない圃場枚数が多くなり、それらについて緻密な圃場情報を収集し、管理することが容易ではなくなる。水田作の個別大規模経営や営農集団の経営運営において、農用地に関する合理的な情報管理の仕組みを構築すること

* 鳥取大学農学部農林総合科学科情報科学講座

* *Department of Agricultural Information Science, Faculty of Agriculture, Tottori University*

** 鳥取大学大学院農学研究科

** *The Graduate School of Agricultural Science, Tottori University*

が要請されているのである。

このような必要性から、農家レベルでも普及が進んできたパーソナルコンピュータ（以下、パソコンと略称）を利用して、大規模な水田作経営の農地管理を支援するためのソフトウェアの開発が手がけられるようになってきた。しかしこれまでのところ、実用に供されるようになったソフトウェアは少数であり³⁾、しかもそれらは機能や操作性、価格面において普及上の問題をもつものが多く、本格的な実用段階に達していないのが現状である。

とくに、水田作経営の圃場情報の利用効果を高めるためには、コンピュータの利用システムのなかに作図機能を備えて、地図情報と圃場情報を重ねて用いるような機能をもたせることが必要である。この点で、作図機能の側面からとらえて、従来の提供ソフトウェアには扱える面積範囲や作図処理速度などについて限界をもつものが多かった。生産現場で幅広く活用できるようなコンピュータ用の圃場管理システムを開発するためには、農業集落などの小地区ごとに分割作成された圃場図を連結して、連続した広域情報として処理できるようにしたり、ユーザーにとって操作性や価格面からみて使いやすい簡易なシステムに仕上げることが大切である。

そこで本研究では、パソコンによる一筆圃場管理システムの開発にむけて、その基礎となる圃場図を用いたマッピングシステムの開発を行った。その際、とくに力を注いだのは以下の点である。

一般に圃場図は、土地基盤整備事業の進行にあわせて、農業集落を単位に作成される場合が多く、作成年次や地区が異なると地図の縮尺や方位などが違ってくることが多い。そこで、これらの種類の異なる圃場図をコンピュータ上で連結させ、広域情報として一括、高速処理できるように工夫した。また、圃場区画の変更や農道の新設などによって図面に変更が生じた場合には、簡単に修正が加えられるようにした。そして、圃場の属性や権利、利用に関わる圃場情報のデータベースが、できるだけわかりやすく効果的に利用できるよう、作図及び作表機能の充実を努めた。

なお、ソフトウェアの利用対象者として想定しているのは、一筆圃場管理システムに関心の高い、水田作の個別大規模経営や地域営農集団である。

ハードウェアの構成

本システムの開発環境は、以下の通りである。

機種：NEC PC-9821Xp(メインメモリ640KB+プロテクトメモリ13MB、ハードディスク340MB内蔵)

CRT：ICM LS-2617FN II

解像度：1024×768・256色

OS：MS-Windows Ver.3.1

言語：MS-VISUAL BASIC Ver.2.0 PROFESSIONAL EDITION²⁾、開発ツールとしてSpyWorks-VBを使用

デジタイザ：グラフィテック KT4300

利用環境は、NEC PC-9801シリーズでWindowsが動作可能な環境であり、解像度が1024×768・256色の表示が可能であれば、ほぼ動作する。

圃場図を利用したシステムを考える際には、地図上での圃場選択やデータ表示が必要であるので、できるだけ、GUI (Graphical User Interface：マウス操作やマルチウィンドウなどの視覚的な操作を重視したインターフェイス)を意識しなければならない。WindowsではGUIを前提としているため、開発者や利用者にとって利用しやすい環境を提供できるので、システム開発をするうえで、利用するうえでも有効なOSであるといえる。

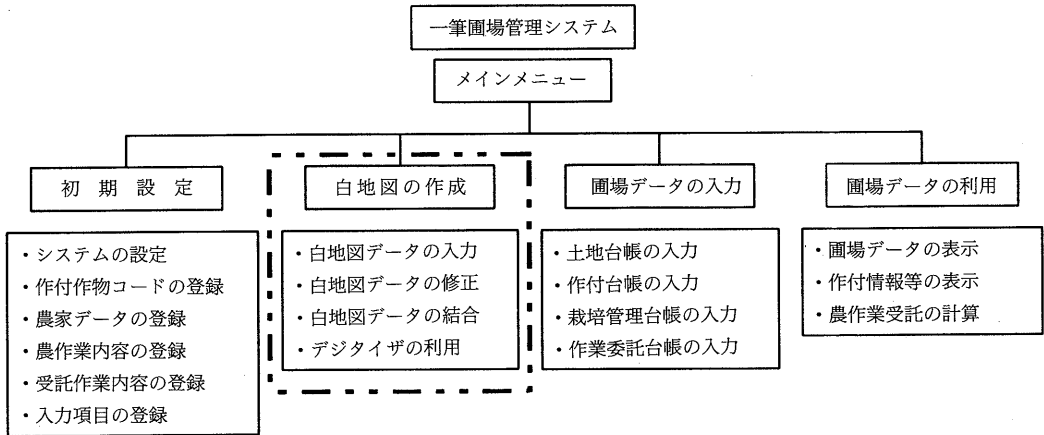
さらにWindowsでは、機種依存性が少ないため開発作業を行いやすく、利用環境にも多様に対応できる。現在のパソコンの発展状況を見ると、高性能機種の低価格化が進行しており、将来的にはWindowsが標準OSとして普及することが考えられる。

マッピングシステムの構成

本システムでは、圃場図の原図をもとに圃場情報を活用するための白地図を、パソコンによって作成する。白地図は、原図に示された圃場の角部分の端点の座標を入力することによって作成し、さらに地図上に何らかの変更があった場合には、必要な修正を加えることができる。また、旧字あるいは町村の範囲内で、圃場図は地区ごとに細分化されて複数の枚数によって作られているのが一般的である。その場合には、それぞれ別々にデータ入力して作成した白地図を、コンピュータ上で結合させることによって、地域内の圃場図を1枚の図面にまとめることが可能である。そして、このマッピングシステムによって作成された白地図と、圃場データベースを連動させて利用することにより、各種の管理情報の処理にあたる。マッピングシステムは、「白地図データの入力」「白地図データの修正」「白地図の結合」「デジタイザの利用」の4部構成からなっており、それぞれの部分で次のような作業を行う(第1図)。

1) 白地図データの入力

ここでは、コンピュータによって圃場の角部分にあた



第1図 一筆圃場管理におけるマッピングシステムの構成図

る端点の座標データを入力し、それをを用いて白地図の画像を作成する。

作業手順としては、もっとも単純な方法として、まず、原図の上に透明な方眼フィルムをのせて、圃場の端点ごとに線を引くためのx, y座標を人力によって読み取る。次に、その座標データをコンピュータに入力し、画像情報として白地図データに変換する。その場合、原図である圃場図の縮尺と方位は限定しないので、点座標の読み取り作業を行いやすいように、必要に応じて図面を拡大、縮小しておくことと便利である。

座標の読み取りを行う際の地図上の原点(0, 0)は、原図の左下の端に設定する。点座標のデータについては、x座標とy座標を取得するとともに、その座標点が終点であるかどうかを示すF値をあわせて取得する。F値は、1ならば始点または連続点であることを示し、2ならば終点であることを示す。そのx, y座標とF値をコンピュータに入力したのち、圃場を囲む点を結んで線を描き、白地図を作成する。

また、作成した白地図が間違っている場合は、ディスプレイ上でそれを拡大したり、入力した順に端点を結んで線を引かせていくことによって、誤った数値を探し出して修正することができる。

白地図は、Windows上で一般的に利用される画像フォーマットであるBMP (Bitmap) 形式で保存する。これによって応用範囲が広まることになり、たとえば、ワープロソフトで資料等を作るときには、文章中に図面をそのままの形で挿入することができる。

2) 白地図の結合

ここでは、原図にそって別々に入力した白地図データ

を結合する(第2図, 第3図, 第4図)。

圃場図の原図は、土地改良の実施時期にあわせたりして、農業集落などの小地区単位に分割して作成されていることが多く、旧字あるいは町村の範囲で連続的に一括して圃場図の表示を行おうとするときには、地図の連結作業が必要となる。しかし、地図を結合する際に、図面間で縮尺や方位が異なることが多く、その場合には単純に地図を接合することはできない。

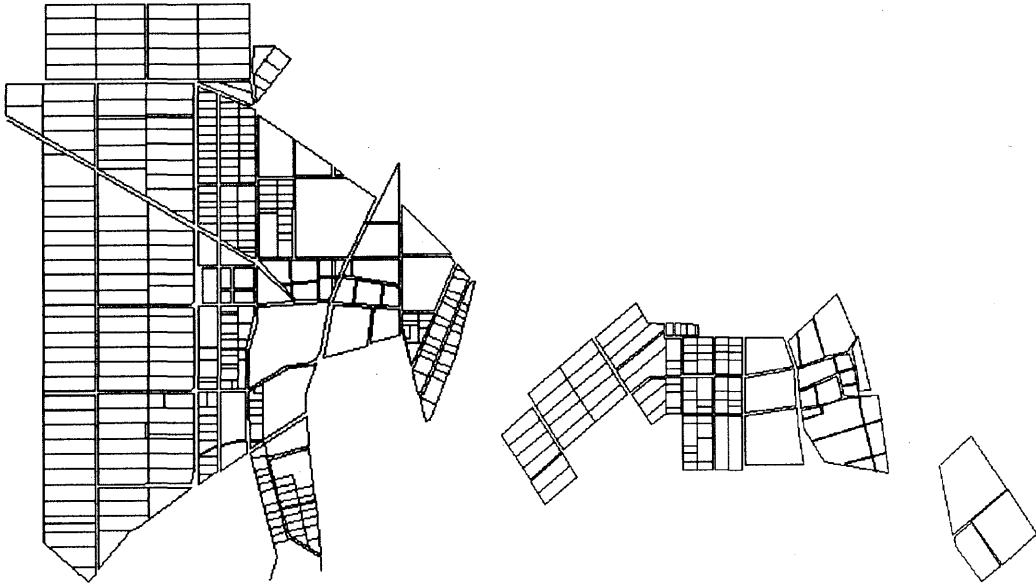
そこで本プログラムでは、つなぎ合わせようとする2枚の地図の縮尺を以下の数式によって同率にし、方位も回転させることによって、作業が容易に行えるようにした。そのため2枚の地図は、たとえば3000分の1と2500分の1のように縮尺が違っていても、北向きの方位がずれていたとしても、作業には大きな支障はない。

結合の方法としては、まず、2枚の圃場データを読み込み、それぞれの地図で一致する2点を設定して、その座標データを入力する。次に、それらの図面上の一致する点の座標を用いて、異なる白地図の2点間の距離が一致するように倍率を求め、続いてx, y座標のずれを求めて回転角度を算出し、移動させようとする側の地図上の点をすべて変換して、その後連結作業を行う。

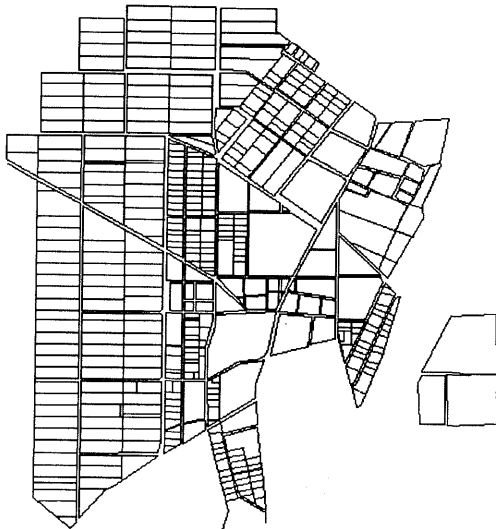
白地図の結合は、具体的には以下のようなアルゴリズムによって行う。

まず、変換しない側の地図の2点を(x1, y1), (x2, y2)とおき、変換を行う側の2点を(x3, y3), (x4, y4)とおく。

次に、変換のための倍率を求めるために、2点間の距離を等しくさせるaの値を求める。



第2図 入力した白地図(結合前)



第3図 結合した白地図

$$\sqrt{(x1-x2)^2+(y1-y2)^2}=\sqrt{(ax3-ax4)^2+(ay3-ay4)^2}$$

そのとき、等倍にした点座標 $(ax3, ay3)$, $(ax4, ay4)$ を $(x3', y3')$, $(x4', y4')$ とおく。

続いて、 x , y 座標のずれを求める。これは、次式によって、 $x0, y0$ として算出する。

$$x0=x1-x3' \quad y0=y1-y3'$$

次に、回転角度を求める。点を θ 回転させる式は次のようになる。

$$x2=x4'\cos\theta-y4'\sin\theta$$

$$y2=x4'\sin\theta+y4'\cos\theta$$

最後に、点座標を変換する式は次のようになる⁹⁾。

$$X=(x-x0)\cos\theta-(y-y0)\sin\theta+x0$$

$$Y=(x-x0)\sin\theta+(y-y0)\cos\theta+y0$$

また、変換後、数値の丸め誤差などによって、図面上に微妙なずれが生じるので、マウスによって2枚の地図を上下左右に動かして、適切な位置に補正する。

3) 白地図データの修正

ここでは、前述の「白地図データの入力」部分で作成した白地図に対して、必要な修正作業を行う。

圏場図の原図から点座標を読み取る際、曲線や細かい点を正確にひらうことが難しく、それをコンピュータ上で数値化するときに微妙なずれが生じやすい。そのため、読み取った座標データをそのまま用いると、小区画圏場

や不整形圃場、道路などを正確に画像表現することが難しい。また、圃場が再整備されたり、農道が新しく設置されたりして、白地図に修正を加えなければならないことが少なくない。このような理由から、図面の修正機能がどうしても必要となる。

修正作業においては、マウス操作によって白地図上に点を打ったり、直線を引いたり、あるいは線を消すといった動作を繰り返して、より正確な図面を作成する。白地図の点座標データには、ドット単位の精度が要求されるので、修正作業は細かい仕事となる。そこで、作業精度を高めるために、ディスプレイに表示した図面を部分的に拡大表示し、白地図の修正を容易に行うことができるように工夫した。

また前述したように、白地図はBMP形式のファイルとして保存されているので、それをさらに細かく修正したいときは、市販のグラフィックソフトを利用することも可能である。

4) デジタイザの利用

パソコン用の周辺機器であるデジタイザを利用して、白地図データの入力作業の効率化を進めることができる。本システムでは、この機能を装備して作業の円滑化をはかるようにした。

ここでは、「白地図データの入力」部分で解説したのと同様な考え方で、人力の代わりにデジタイザを利用して、原図からx、y座標とF値を取ることににより、白地図を作成する。デジタイザを利用して圃場の点座標を取る際には、その数値が正しい位置を示しているかどうかを、コンピュータ画面を見ながら確認することができるので、正確かつ迅速に作業を行うことができる。ただし注意を要するのは、デジタイザでは微妙な手ぶれによって、ドット単位の誤差が生じやすい点である。できるだけ鮮明な白地図を作成するためには、画像表示によって座標数値を確認することが必要である。

圃場データベース

圃場データベースは、マッピングシステムを利用して、一筆圃場管理を行うために必要なデータを集積したものであり、作成した白地図と連動させて圃場情報を表示するためのデータベースである。圃場データベースの内容としては、圃場の所有と利用に関する権利関係や圃場条件のデータをまとめた土地台帳、作付状況をまとめた作付台帳、作業実施状況をまとめた栽培管理台帳、地域の受委託作業状況をまとめた作業受委託台帳を取り扱う。

データベースに組み込まれる各台帳では、以下のよう

な情報が扱われる。

「土地台帳」

個々の圃場について所有と利用の権利関係や圃場条件に関するデータを扱う。具体的な項目は以下の通りである。

- ・圃場コード、地番、地名、地目、所有者、耕作者、地積面積、耕作面積、賃貸借権設定、地価、小作料、土地利用計画、土地利用権設定年数、土壌、用水、排水、水系、農道、機械作業の難易、土地改良を行った年度

「作付台帳」

圃場に作付けされた作物に関する基本情報を扱う。具体的なデータ項目は以下の通りである。

- ・作付作物名、品種、品質、単収、土壌中に含まれる窒素・リン酸・加里の量、作付来歴

「栽培管理台帳」

各圃場に作付けされた作物に対して実施された農作業に関する一連のデータを扱う。具体的な入力項目は以下の通りである。

- ・作業日時、作業内容、使用資材、使用資材量

「受託作業台帳」

各圃場で行われた受託作業に関するデータを取り扱う。具体的な項目は以下の通りである。

- ・受託作業日時、受託作業内容、作業者、その他

以上の各種の圃場データを白地図と連動させて表示させることにより、圃場情報をわかりやすく効果的に利用することができる。その際の結果の表示方法として、以下のようなものを用意した。

1) 圃場の色分けによる表示

地図と連動させて、条件に適した圃場を色分けする。具体例として、水稻の品種ごとにあるいは所有者別に、圃場に色分けをしたりすることなどをあげることができる(第5図)。

2) 表・グラフの表示

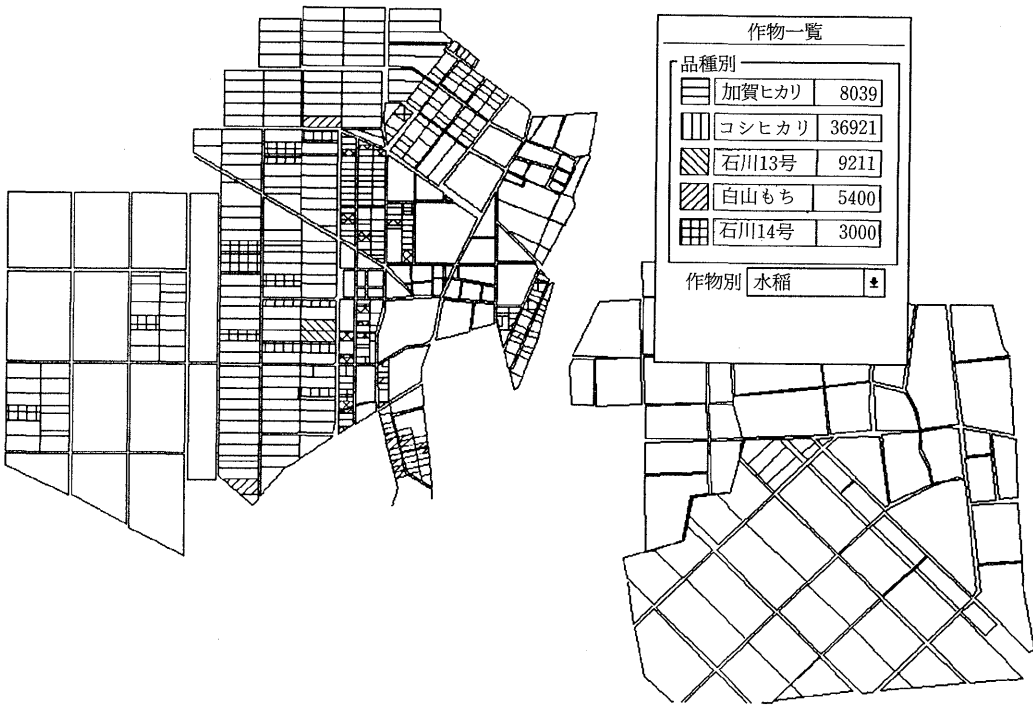
圃場単位の情報を集計整理して、表やグラフにまとめて表示する。たとえば、水稻品種別の作付面積や圃場数、借地の種類別の面積表示などがその例である。

3) 受託作業の料金表示

各圃場で実施した受託作業について、積み上げ計算を行い、委託者別、オペレータごとに料金表を作成する。

4) 各種計画の立案

圃場単位に色分けを行ったりして、作付計画や作業計画の検討作業を支援する。



第5図 水稲の品種別作付図

ファイル構造

マッピングシステムで利用される圃場のx, y座標とF値の数値データは、ランダムファイルの形式で保存される。そこでは、数値データの構造として、地図のx, y座標とF値を1つの構造体として扱っており、x, y座標は整数値、F値は2バイトの文字型変数として設定している。構造体の形態を用いることで、同じ変数型ではない関連のある情報をまとめることができるので、データを利用しやすくなる。

実際の地図のファイル構造は、以下のような形式になっている⁴⁾。

pt(i)	x	y	F
1	525	323	1
2	652	323	1
:	:	:	:
:	:	:	:
812	325	123	2

pt(i)はi番目の構造体であり、i番目のx, y座標及びF値はpt(i).x, pt(i).y, pt(i).Fとして表される。

圃場データベースは土地台帳、作付台帳、栽培管理台帳、受託作業台帳に別れているため、各台帳についてファイルを作成する。それぞれの台帳については、圃場コードがインデックス値となっており、台帳の圃場コードを読み込むことで、1つの圃場に関する4つの台帳を連結することができる。

圃場データベースも地図の座標データと同じく、構造体を用いることで情報をまとめている。

例えば、土地台帳に関するファイル構造は以下のような形式になっている。

t(i)	code	x	y	number ..
1	1001	252	234	2- 1 ..
2	1002	524	253	2- 3 ..
:	:	:	:	:
:	:	:	:	:
812	3025	354	234	4-12 ..

t(i)はi番目の圃場の土地台帳情報の構造体を示している。codeは圃場コードであり、x, y座標は圃場コードで表された圃場内部の任意の座標である。この座標はデータベースとの連動に使用する。numberは地番であり、以

下は前節で表した土地台帳のデータが順番に保存されることになる。

データベースと地図の連動

本システムの大きな特徴は、一筆単位の圃場データベースを地図と連動させることによって、視覚的に効率よく作業ができるようにしたところにある。その場合、地図と圃場情報の連動について、地図上で圃場領域の認識を正確に実行させることが技術上のポイントとなる。圃場の領域認識は、以下のようなカラーマッチングの方法で行うようにした。

まず、圃場データとして、線で囲まれた番号 i 番目の圃場内部の任意の点 $(x(i), y(i))$ をとって圃場コード $code(i)$ を設定する。

次に、圃場情報表示のために j 番目の圃場上のある点 $(x(j), y(j))$ を選択したとする。その $(x(j), y(j))$ を内部に持つ、周囲を線で囲まれた図形を塗りつぶす。そのとき、点 $(x(i), y(i))$ が点 $(x(j), y(j))$ と同領域にあれば、 $(x(i), y(i))$ は一緒に塗りつぶされたことになり、もし、違う領域であるならば、 $(x(i), y(i))$ は塗りつぶされていないことになる。

つまり、塗りつぶしを行った後、すべての圃場内部の圃場コードを付した任意の点 $(x(i), y(i))$ (i は、1から最大圃場枚数までの値)と $(x(j), y(j))$ の点上の色を検索し、 i 番目と j 番目の色が同じであるならば、同一圃場であると認識する。そこで、 j 番目の圃場コードは $code(i)$ となる。その後、 $code(i)$ の圃場データ呼び出すことで、その内容を確認することができるのである。なお圃場コードには、町村や集落が識別できるような形で数値を付しておく、より迅速な作業処理が可能となる。

ただし、注意しなければならないのは、コンピュータ上で白地図を作成する際に、圃場を確実に線で囲む必要があることである。これは、一枚の圃場が正確に線で囲まれていないと、カラーマッチングを行うときに他の圃場を塗りつぶしてしまうことになり、圃場データベースと地図が正しくリンクできないからである。

適 応 効 果

本システムを利用することにより、以下のような適用効果が期待できる。

①縮尺や方位の異なる地図でも結合できるため、同一の地域内にあって地区ごとに別々に作成された図面でも、

それらをコンピュータ上で、一枚の圃場図として利用することができる。そのため、圃場情報を広域に及んで一括的に利用することができる。

②作成した白地図を座標データとしてではなく、WindowsのBMP形式の画像データとして処理することにより、画像の表示や拡大、移動といった処理を高速に行うことができる。

③圃場データベースとの連動によって、地図上で土地利用に関する実態分析や計画作成を行ったりすることが容易になる。

利用上の留意点

本システムを利用する際には、以下のような点に留意しなければならない。

①白地図作成のためにとる圃場角の端点の上限数は、メモリーの制限から20,000個になっている。1枚の圃場図を作成するのに入力する点座標は、多くても1,000個程度であるので、20枚以上の圃場図を描くことができる。

②圃場情報については、最大2,000枚分を扱うことができる。水田作の個別大規模経営や地域営農集団が、通常扱っている圃場枚数をやや多く見積もっても、十分対応が可能であると推察される。ただし、利用する圃場図の面積領域が広がる場合には、画像データの情報量が大きくなるため、圃場情報を扱える圃場枚数が制限を受けることになる。

小稿は、平成6年度文部省科学研究費補助金（一般研究(C)）の助成を受けて実施した研究成果の一部である。

文 献

- 1) 小林 一・向井俊忠：水田作経営の一筆圃場管理システム。鳥大農研報 45 145-151 (1992)
- 2) Microsoft VISUAL BASIC リファレンスマニュアル。Microsoft社、東京 (1994)
- 3) 日本ソフトウェア協会編：農業ソフトウェアブック。楽遊書房、東京 (1994) pp.214-215
- 4) 農業研究センター・農業計画部・経営管理部：営農情報簡易地図化プログラム。農業経営研究資料 16 13-20 (1990)
- 5) 八木伸幸・井上誠喜・林 正樹・中須英輔・三谷公二・奥井誠人・鈴木正一・金保明：C言語で学ぶ実践画像処理。オーム社、東京 (1992) pp.170-193

第4図 「白地図の結合」部のメインプログラム・リスト

```

Sub Command3D1_Click ()
  c3 = -1 / Sqr(((a1 - a2) ^ 2 + (b1 - b2) ^ 2) / ((a3 - a4) ^ 2 + (b3 - b4) ^ 2))
  a1 = a1 * c3: a2 = a2 * c3: b1 = b1 * c3: b2 = b2 * c3
  bx1 = a2 - a1: bx2 = a4 - a3
  by1 = b2 - b1: by2 = b4 - b3
  If bx1 = 0 Then bx1 = .0000001
  If bx2 = 0 Then bx2 = .0000001
  If by1 = 0 Then by1 = .0000001
  If by2 = 0 Then by2 = .0000001
  b = (by1 * by2 + bx1 * bx2) / (bx1 * bx1 + by1 * by1)
  a = (-bx2 + b * bx1) / by1
  dx = a3 - a1: dy = b3 - b1
  i = 1
  x1 = pt(1).x * bai: y1 = pt(1).y * bai
  Do While Val(pt(i).f) <> 0
    pt(i).x = pt(i).x * bai: x2 = pt(i).x
    pt(i).y = pt(i).y * bai: y2 = pt(i).y
    i = i + 1
  Loop
  j = i - 1: i = 1
  Do
    x2 = ((c3 * Pt2(i).x - a1) * b - (c3 * Pt2(i).y - b1) * a + dx + a1) * bai
    y2 = ((c3 * Pt2(i).x - a1) * a + (c3 * Pt2(i).y - b1) * b + dy + b1) * bai
    pt(i + j).x = x2: pt(i + j).y = y2: pt(i + j).f = Pt2(i).f
    i = i + 1
  Loop While Val(Pt2(i).f) <> 0
  x1 = pt(1).x - xmin: y1 = ymax - pt(1).y
  For i = 1 To j
    x2 = pt(i).x - xmin: y2 = ymax - pt(i).y
    Link_Pic!Picture2.Line (x1 + 2, y1 + 2)-(x2 + 2, y2 + 2)
    If Val(pt(i).f) <> 2 Then
      x1 = x2: y1 = y2
    Else
      i = i + 1
      x1 = pt(i).x - xmin: y1 = ymax - pt(i).y
    End If
  End For
  Next
  fx = 0: fy = 0
  x1 = pt(j + 1).x - xmin: y1 = ymax - pt(j + 1).y
  i = 1
  Do While Val(pt(i + j).f) <> 0
    i = i + 1
    x2 = pt(j + i).x - xmin: y2 = ymax - pt(j + i).y
    Link_Pic!Picture4.Line (x1 + 2, y1 + 2)-(x2 + 2, y2 + 2)
    If Val(pt(j + i).f) <> 2 Then
      x1 = x2: y1 = y2
    Else
      i = i + 1
      x1 = pt(j + i).x - xmin: y1 = ymax - pt(j + i).y
    End If
  End Do
  Loop
  wrt=StretchBlt(Link_Pic!Picture3.hDC,0,0,Link_Pic!Picture3.ScaleWidth,Link_Pic!Picture3.ScaleHeig
ht,Link_Pic!Picture2.hDC,0,0,Link_Pic!Picture2.ScaleWidth,Link_Pic!Picture2.ScaleHeight, SRCCOPY)
  wrt=StretchBlt(Link_Pic!Picture3.hDC,0,0,Link_Pic!Picture3.ScaleWidth,Link_Pic!Picture3.ScaleHeig
ht,Link_Pic!Picture4.hDC,0,0,Link_Pic!Picture4.ScaleWidth,Link_Pic!Picture4.ScaleHeight, SRCAND)
  Link_Pic!Picture1.Refresh
End Sub

```