WORKING PAPER 487

# REGIONAL LAND EVALUATION USING A PILOT GEOGRAPHICAL INFORMATION SYSTEM BASED ON LINEAR QUADTREES AND A RELATIONAL DATABASE

JAMES HOGG, MARK GAHEGAN*
AND NEIL STUART

* Current address: Department of Computer Studies, University of Leeds

School of Geography
University of Leeds
Leeds LS2 9JT

April 1987

## ABSTRACT

A pilot GIS based on linear quadtrees and a relational database is described. It consists of two parts: a spatial database of binary images represented as linear quadtrees; and a relational database of objects abstracted from the binary images and augmented by aspatial data provided by users. The two parts are integrally linked within the GIS. This allows users interactive access to explore spatial objects and relationships. The approach has wide potential application for analysis and management of both spatial and aspatial geographic data. To illustrate this, an example describes how to use the GIS for regional analysis of terrain. The authors conclude that linear quadtrees have several advantages for analysing spatial data from raster scanned maps and remote sensing systems. They also conclude that a relational database management system provides a powerful and convenient way to access and analyse objects abstracted from spatial data. The combination of linear quadtrees and a relational database management system provides acceptable levels of performance and offers flexibility and adaptability to the needs of a variety of users of the GIS. Work is continuing to evaluate the potential applications of the pilot GIS with larger and more diverse types of data.

# 1 INTRODUCTION

As part of a project to evaluate the potential of geographic information systems (GIS) for regional analysis of land resources, a pilot GIS based on linear quadtrees and a relational database has been developed. It consists of two parts: a spatial database of binary images represented as linear quadtrees; and a relational database of objects abstracted from the quadtrees and augmented by aspatial data provided by users. The two parts are integrally linked within the GIS to allow users interactive access to explore spatial objects and relationships.

This paper describes the structure and components of the pilot GIS and illustrates its potential for regional analysis of land resources. It focuses on the use of linear quadtrees for storing spatial data and a relational database management system (RDBMS) for storing information about objects abstracted from the spatial database. It serves to illustrate the theory and potential applications of the pilot GIS. A much larger database would be used in practice however to provide a means of interrogating satellite remote sensing images which had been classified and registered to digital map and field survey data.

The remainder of this paper is organised as follows. The second section reviews the nature of land resources evaluation, the growing need for facilities for analysis and management of geographic data in GIS and the concept linear quadtrees and relational databases. The third section outlines the structure and components of the pilot GIS. The fourth section provides an illlustration of its use with reference to a small area of the Peak District, Derbyshire. In the fifth section we discuss briefly the scope and limitations of the pilot GIS. Finally we draw conclusions from the results that we have been obtained so far.

## 2 REVIEW

### 2.1 Land resources evaluation

Land resources evaluation is concerned with making assessments about man's potential use of land for purposes such as agriculture, forestry, recreation, urban planning or engineering (Christian and Stewart, 1968). It involves analysis of the capabilities and constraints imposed by the physical characteristics of a region and is usually conducted in support of some decision-making process in land management. It is carried out by scientists from many different fields of study but, in many cases, they adopt a similar approach, though of course the level of detail and specific requirements and methods usually differ (Mitchell, 1973). Moreover they draw typically upon a common core of information about the land. The extent of this common core is usually substantial - all require basic information about topography, geology, soils, climate and lan use. While there are of course minor differences in specific requirements, the major difference often lies in the level of detail required.

In conducting land resources evaluations, land is usually characterised by a distinctive assemblage of attributes and interlinking processes in space and time (Townshend, 1983). The attributes include topography, geology, soils, water, climate, vegetation, and fauna as well as the results of human activity. Mitchell (1973) stresses that land evaluation is a broad term which encompasses analysis, classification and appraisal of information from a variety of sources for a potential land use. Analysis involves selecting characteristics which have importance for a particular application and compiling land characteristics. Classification relates to the organisation of characteristics which distinguish one area from another and which characterise each. Appraisal uses these characteristics, along with other properties, to assign a value to a piece of land, expressed either by a numerical value or by a judgement of its worth in qualitative terms.

A land resources evaluation system has several basic requirements. Mitchell (1973) identifies three: 1) a means of answering queries from users; 2) a means of acquiring, storing, analysing and displaying information about the land and its potential uses; 3) a means of retrieving and manipulating information. The traditional approach to fulfilling these has been by preparing manually various maps and transparent overlays showing features, such as slope, aspect, soils, drainage and other characterisitcs and by preparing statistical and other textual reports. Visual comparison and interpretation of maps and reports leads to an evaluation of regional land resources for a particular application. The basic source of information for all these maps has usually been aerial photographs, though other forms of remote sensing are increasingly being used to aid sub-division of the land, such as multi-spectral imagery from satellites such as Landsat and SPOT (Harris, 1986).

### 2.2 Geographic Information Systems(GIS)

The concept of a geographic information system or GIS has evolved gradually over the past twenty years. As Young (1986) has shown, it is a broad concept. It refers to all systems that are designed to store, retrieve, manipulate and display data which relate to locations on the Earth's surface (Marble & Peuquet 1983). Whereas in the past GIS were analogue systems comprised of collections of maps, statistics and reports, most modern GIS use computer technology. This has become increasingly necessary because of the need to handle extremely large volumes of geographic data and to study the complex inter-relationships which are present. Many computerised GIS were originally developed independently, on different computer systems and over varying periods and are often optimised for particular applications (Tomlinson 1984; Young, 1986). Today therefore there exists a diversity of GIS, though Dangermond (1986) noted a trend towards integration of hardware, software and applications within many organisations.

As the concept of a GIS has gradually changed with advances in computing technology, emphasis has shifted from storing, retrieving, manipulating and displaying data for automated cartography to much more sophisticated systems for modelling and decision-making in land management (Crain and MacDonald 1984). While these are just beginning to appear, they take advantage of the latest advances in computing theory and technology to allow interrogation of spatial and aspatial relationships within geographic data. The development of intelligent GIS in which the concepts and techniques of artificial intelligence and database systems are integrated represents a major new field of research (Smith and Pazner 1984; 1984; Smith and Peuquet 1985; McKeown at al. 1984). With optical disk technology recognised as a break-through in mass storage media, new opportunities for creating massive geographic databases have become a practical possibility ( Light, 1986a; 1986b ).

In designing a GIS, a critical decision is the choice of data model to represent spatial data. This is the abstraction that is used to represent land properties in the computer. Peuquet (1984) reviewed the different types of spatial data models that have been used in GIS and compared their performance. Spatial data have been represented using many different types of data models, but a basic difference is between vector and raster types of data model.

1 Vector data

In this type of data model, the logical unit in a geographical context corresponds to a line on a map. The line is recorded as a series of x-y coordinates with a heading describing the feature it represents. Vector data is widely used in digital cartography and is probably the most widely used data model today.

2 Raster data

In this type of data model, a fixed-sized square cell or raster represents spatial data in a binary array or grey-scale image. The development of data models based on rasters has been largely driven by advances in the technology of remote sensing and computing over the past decade (Marble and Peuquet 1983). The use of multi-spectral scanning systems in satellite remote sensing has been a major influence. At the same time, there have been significant advances in the technology of raster scanning, video digitising and parallel processors. These have greatly accelerated the whole process of digitising of maps and related documents. Because all these systems use a square cell or raster, this is probably the most practical tiling or tessellation. A number of other possibilities exist however which may be theoretically better than the regular tessellation or raster (Bell et. al., 1983).

Approaches to the design of spatial databases have been subjected to critical review by Peuquet (1984) who identified a number of limitations in existing approaches. These limitations have been amplified by Abel and Smith (1986), who pointed out that most designs to date have centred on the use of formatted database models and established database management systems which offer poor support for the distinctive aspects of spatial database. In particular, Abel and Smith (1986) stressed four critical deficiences:

1. There is no provision for a high-level view of spatial entities in the data model;
2. Spatial relationships such as containment and intersection are missing from the query languages and are difficult to express;
3. The physical methods of accessing data are not efficient for queries involving spatial relationships;
4. The query languages and the ancillary facilities do not provide for the graphic display of the retrieved data.

These factors restrict the potential of automated GIS based on the use of vector or raster data models to cope with the variety of different forms of geographic data and the anticipated massive volumes of spatial data that will be involved. For these reasons, attention has focused increasingly hierarchical data models to represent large volumes of spatial data in a compact form which allows efficient retrieval and processing using set logic. These data models are known as quadtrees.

**2.3 Regions as quadtrees**

Quadtrees have been studied extensively over the last decade and a large number of algorithms have been published for performing operations on them. Early work on the subject by computer scientists, like that of Klinger (1971) and Finkel and Bentley (1974), stimulated much of the later work on a large number of hierarchical data models, including quadtrees (Oliver and Wiseman, 1983; Rosenfeld et al. 1982, 1983; Samet et al. 1984, 1985; Abel, 1985,1986). Samet (1984) provides a comprehensive review of quadtrees and related hierarchical data models which are based upon recursive decomposition of an image. In a more recent paper, Samet et al. (1986) outline recent developments in their research into quadtree-based geographic information systems.

A quadtree is constructed from a square binary array of pixels which represents an image. The set of connected black pixels in the image is referred to as a region, such as an area of forest (Fig 1). If we assume that an image is comprised of a $2^n \times 2^n$ binary array of pixels (Fig 2), then a quadtree encoding represents this image by recursively sub-dividing it into four quadrants until no further sub-division is necessary (Fig 3). This occurs when we obtain square blocks (possibly single pixels) which are homogeneous in value (i.e. either all black or all white) or when we reach the level of resolution that we require. This process is represented by a tree with four branches, in which the root node corresponds to the whole image and the four sons of the root node correspond to the four quadrants (Fig 4). The leaf nodes of a quadtree correspond to homogeneous blocks for which no further sub-division is necessary. Nodes at level n represent square blocks of size $2^n \times 2^n$. Further details of this process are described by Hogg and Gahegan (1986), Gahegan and Hogg(1986) and Samet (1984).

While a large number of data models for quadtrees have been proposed, that of Gargantini (1982) is probably most attractive because it is economical in its requirements for space. Gargantini(1982) proposed a data structure to represent quadtrees which is known as a linear quadtree. It is in the form of a linear list consisting of the quadtree nodes in some order of traversal of the tree (Table 1). A number of different forms of keys is available to map set of ancestors to a numeric key (Fig 5)(Morton, 1966; Gargantini 1982; Abel and Smith 1983). Each node possesses a unique key, which can be processed to show the level at which the node was formed and its x,y address in the original image. Two quadtree files can be processed together (to find the intersection or union) simply by reading the nodes one at a time. Alternatively, nodes can be matched to x,y addresses by decoding their keys in a bitwise fashion (Peuquet, 1984). This provides a flexible means of accessing the data represented by the quadtree.

**2.4 Object Management and Relational databases**

An object is an entity that exists in the real world and possesses attributes that we are interested in (Oxborrow, 1986). An object, such as a particular reservoir, wood or census district, can be related to its location in an image by its unique position in the linear quadtree. It may have many attributes which are specific to it alone, such as its size, orientation or shape. In addition it may have aspatial attributes such as volume of water in the reservoir, ownership of a wood or population of a census district. Quadtrees are generally unsuitable for storing aspatial attributes, so an alternative must be found.

One ready alternative is to use an existing formatted data base. A database can be regarded as a collection of regularly formatted data which can be accessed by more than one person and which can be used for more than one purpose (Frost, 1986). Typically it consists of a collection of similar data such as bank accounts or student records but it can equally consist of objects abstracted from the spatial database of linear quadtrees.

A data base management system (DBMS) is a series of computer programs which provide the essential services to implement and maintain a data base (Frost, 1986). It usually provides the following features:

(i) establishment of data relationships within the data base;

(ii) the facilities to implement and load it with data;

(iii) the facilities to maintain and update the data base;

(iv) the facilities to allow the user to interface with and use the data in the data base;

(v) data security and control.

There are many different types of DBMS currently in use but few offer an efficient data model of the geographer's abstraction of the real world. This requires a suitable spatial data model such as the linear quadtree and a suitable data model for handling attribute data. By combining the spatial data model based on linear quadtrees and the aspatial data model based on relational modelling both spatial and aspatial data can be analysed (Gahegan and Hogg,1986).

The relational approach to data modelling is based on the mathematical concepts of relations and sets (Oxborrow, 1986). The form of a relation is mathematically relatively simple yet it is capable of capturing many of the relationships represented by the more complex data structures that are in current use. A relation may be viewed as a table, where each row is a tuple or record and each column corresponds to one component or attribute. The columns are given names called attributes. The set of attribute names for a relation is called the relation scheme. If we had a relation called reservoirs, and its relation scheme has attributes, we write reservoirs (attribute_1, attribute_2, etc.). The collection of relation schemes used to represent information is called a relational database scheme (Date, 1986).

A relational database allows objects and their attributes to be organised in a way that shows their logical relationships. Furthermore, a relational database management system provides quick access to data along a variety of access paths, so many types of query can be satisfied. While detailed discussion of the theory and application of relational modelling is beyond the scope of the present study and the reader is referred to the literature on the subject (Date, 1986; Frost 1986; Stonebraker 1986), there has been considerable research into the subject over the past two decades. The concept of the relational model was introduced by Codd (1970), but many others have contributed to its development and extensions (Codd 1979, Ullman 1982, Stonebraker et. al. 1986).

The application of relational models for interactive processing of geographical data has recently been explored by several workers. These include the work of Abel and Smith (1986) and Steiner and Gilgen (1984).

## 3 STRUCTURE AND COMPONENTS OF PILOT GIS

The structure and major components of the pilot GIS, and the relationships between these, are shown in figure 6. The function of each component is described below.

### 3.1 Encoding binary images as linear quadtrees

The program for quadtree encoding accepts binary, square arrays or images which have been generated by vector to raster scan conversion, by raster scanning of maps or by remote sensing systems. A recursive routine is used to encode such images by progressively viewing smaller areas of the image until a homogeneous area or region is found (i.e.pixels have the same colour or value) and a node is formed. The linear quadtree that is formed is stored on disk.

### 3.2 Operations on linear quadtrees

A set of functions can be applied to the linear quadtrees to perform various operations. These include a set of low level functions which are often required by higher level operations, for example, functions for returning the X, Y address or colour of a node, the neighbour of a node or traversing a line across a binary image. The advantage of separating such low level functions is not only to avoid repetition in higher level functions but to allow the representation of the quadtree to be modified without affecting the validity of the higher levels.

Functions for set operations on linear quadtrees form the backbone of the GIS for regional analysis. They allow for intersecting two quadtrees to find areas common to both, for union of two quadtrees to find the amalgamation and for finding the complement of a quadtree.

More complex functions perform various operations on linear quadtrees. These include functions for windowing, where a relevant sub-section is extracted from a quadtree; for traversing the border of a quadtree to return the length of the border (Samet and Tamminen, 1985); and for labelling all the separate regions or components. The labelling function is an extension of the border program that recognises and labels the separate components of an image. A large number of useful functions for operations on linear quadtrees have been devised and reported in the literature. Many of these appear to be potentially useful for regional analysis of land resources but have yet to be added to the GIS.

Functions for operations on linear quadtrees have been implemented in the pilot GIS using the C programming language on a Vax 11/780 computer running Berkeley Unix 4.2.

**3.3 Relational database**

The relational database is used to store information about objects. By applying the border function, the separate components of in a quadtree image can be identified, labelled and stored in the database as individual objects. This provides an alternative way of forming a query i.e. by object as opposed to by region. Queries are commonly answered by processing both objects and regions and for this reason the two databases are integrally linked. The relational database management system manipulates object-based parts of the query and the quadtree processing in the spatial analysis parts of the query.

The relational database Ingres was used to store all information about objects in this study (Stonebraker et al. 1976; Stonebraker 1986). It was chosen for three major reasons: (i) Ingres is a relational database with a query language (QUEL) which is strongly based on relational calculus (Stonebraker et al 1976; Date, 1986). Searches to satisfy a query may proceed across several relations, searching on any attribute or combination of attributes. New relations can readily be added as new data is brought to light without damaging the integrity of the design. (ii) Ingres can readily be interfaced to C programs of the GIS by the preprocessor, EQUEL. This allows commands to Ingres to be integrated into programs at appropriate points, while keeping the database invisible to users. (iii) One of the authors has considerable knowledge and previous experience of using Ingres in earlier research and this was judged important for making rapid progress in the time available.

### 3.4 Query language

The structure of the query interface to the GIS is shown in figure 7. A query language forms the interface between the user and the GIS. It governs the operation of functions and the opening and closing of files holding linear quadtrees in the spatial database. Such files may be whole regions (i.e. region quadtrees), individual components of a region ( i.e. object quadtrees ) or label files identifying nodes in regional quadtrees. For queries involving the use of object information, the query language communicates with the Ingres relational database management system. Users enter queries by a single capital letter e.g. U (union); I (intersection); N (complement). For example, if we wish to find the intersection of grit rocks and gley soils between heights 500 - 700 feet, we enter the compound query:

```
grit I gley I (height500_600 U height600_700)
```

The inner-most expression in parenthesis is evaluated first. The results from this expression are then processed with the next expression. Results from the above query can be formed into a new quadtree by prefixing the query with a file name and an equals sign:

high_ground = (height900_1000 U height1000_1100)

or sent directly to the plotter to produce a map (Hogg, Gahegan & Stuart, 1986).

**3.5 Plotting software**

Output from the GIS is currently displayed on a Versatec electrostatic plotter which is capable of producing black and white maps up to A4 size. While this gives a high level of resolution, it places restrictions on size and colour of plots. All quadtrees that are to be plotted as maps are first processed by software for document production which is provided by the Unix operating system ("pic" and "vtroff"). This converts them to a form which is suitable for plotting.

**3.6 Database interface.**

The result of applying the component labelling function to a quadtree is a quadtree where each node is labelled to show to which separate object it belongs. Information about these separate objects, such as their location, size, shape and orientation, can be collected during the second pass of the component labelling function and stored in the database.

Suppose a user wishes to study all areas of urban land that occur within a selected study area. He can use the component labelling function to identify each separate area by tracing its perimeter. A relation can then be created automatically (called "urban_land"), and a tuple appended automatically to this corresponding to each individual urban area, as it is recognised by the component labelling function. A structure for this relation is given below:

| label | x1 | x2 | y1 | y2 | size |
|---|---|---|---|---|---|
| 1 | 114 | 135 | 86 | 117 | 206 |
| 2 | 188 | 245 | 177 | 233 | 873 |

'label' corresponds to the number assigned automatically to that particular object by the component labelling function. The four figures X1, X2, Y1 and Y2 are the diagonally opposite coordinates of the rectangle that neatly surrounds the object (Fig 9). They are used as a means of streamlining a search by area: a relation is scanned once and any objects lying outside of the area of interest are ignored. The figures, along with the area of the object (i.e. the number of pixels it covers), are easily collected during the second pass of the component labelling function. Hence any queries about the location or size of objects can now be answered directly by referring to the database.

Objects can now be considered as separate entities and processed individually. For example, "urban_land_1" can be extracted from the urban_land quadtree by taking only those nodes whose label is 1 . If a labelled version of the quadtree is no longer available, (it may have been deleted to save space), then the two pairs of diagonal coordinates denoting opposite corners of the rectangle that surrounds it can be used as an index to the relevent parts of the quadtree . The quadtree has of course to be reprocessed to locate the object in question but this can be achieved efficiently especially if the quadtree can be indexed using a B-tree memory management system (Abel, 1984).

By forming a database of objects, it becomes possible to store aspatial or attribute data related to individual objects. This unique feature of the GIS represents a significant advance over current GIS and provides a major increase in its functionality as much geographic data is inherently aspatial in character. Aspatial data can be supplied from various sources. For example, if census districts are stored in the spatial database, then census data pertaining to each district or region can be added. The type of information added depends on the nature of the objects, but there is no reason why several relations, all holding different aspatial data, cannot be added to the database as a need arises. In this way, users can build up the relations that they require to solve problems in their particular field of geographical research. Thus, the facilities of the relational database can equally be used for regional modelling of flows between regions for research in human geography such as population studies, urban and regional modelling and planning or transport studies or for research in physical geography such as catchment hydrology, land evaluation or ecological research.

The relational database can be used to store various definitions of objects which can then become a part of the query language. For example, the definition of 'high ground' might be:

high_ground = height800_900 U height900_1000 U height1000_1100

This could be stored in a relation to enable the query parser to relate the term 'high_ground', when it is employed by the user, to a specific definitionthat has been entered previously. This provides a means of formalising ideas. At the same time, this illustrates how the GIS supports multiple user views, as each user may define selected objects for a given application.

Other information relating to the region as a whole, again obtained from the component labelling function, can readily be added along with these definitions. This might include, for instance, for each occurrence of an object, its area and the length of its perimeter, its shape, orientation and other relevant geometrical variables. Hence information at a higher level can be obtained from the relational database without further reference to the spatial database of quadtrees.

### 3.7 Help system

An interactive help system allows users to remind themselves about the commands and operators at their disposal. The seventeen operators that form part of the query language are shown in fig 10. Users access the help system by pressing key H. Each user has the option to modify the help system for his own use, with the type and level of help geared to his current needs. Thus, the default help statements may be aimed at beginners but each user may over-ride it to give the kind of help that they feel that they require.

## 4 USE OF THE GIS

To illustrate the potential of the GIS for regional evaluation of land resources, a data base was created by digitising geographic data from maps. This represents a small geographical area but is useful for illustrative purposes.

### 4.1 Description of Study Area

A study area was selected close to Matlock, Derbyshire (Fig. 11). This was chosen because of the variety of geology, relief, soils and land use within a relatively small area. The area lies on a boundary of Carboniferous Limestone and Millstone Grit. There is a range of relief from valleys below 100 m to summits over 350 m. An interesting change in land use occurs from the relatively fertile alluvial valleys which support intensive use (orchards, arable, settlements., etc), to the upland areas characterised by extensive uses, such as rough grazing and forestry. Soils change from alluvium on the valley floor, through surface water gleys to brown earths on valley sides, with podsols occurring on the highest ground. The area represents therefore a relatively rich source of geographic variation and contains many identifiable sub-regions suitable for illustrating land resources evaluation by GIS.

The area of approximately 25 sq km was chosen to coincide with existing map sheets and so that the resolution after digitising would be sufficient for the type of queries envisaged. The square area selected corresponded to O.S. map sheet SK 26 SE at 1:10560 scale. This was one of the older O.S. maps produced in 1971. Heights were recorded in feet. The resulting raster grid of 256 x 256 pixels lead to a ground resolution of about 20 x 20 m.

### 4.2 Data Capture

Two different methods were used to build arrays of spatial data for quadtree encoding:

1. Manual Digitising

A GTCO digitiser linked to an IBM PC/XT was used to digitise selected features from the O.S. map at a scale of 1:10560. A series of functions for polygon to raster conversion form part of the GIS. These allow vector data to be mapped on to an array of a specified size. Functions for filling regions then produce binary array images. A set of 10 binary array images of size 256 x 256 were made for images of geology, soils, land use and climatic data.

2. Video digitising.

A video digitising system was used to produce 256 x 256 8 bit images of contours from the O.S. map. The contours were first traced onto transparent overlays and each contour interval was coloured by hand to produce a grey scale image. This was then captured by the video digitiser and frame grabber then transformed to fit a square 256 x 256 array. The result was 10 binary array images of relief.

All the binary array images were then quadtree encoded to produce a database for the area for testing various operations on quadtrees and land evaluation.

### 4.3 Illustration using GIS

Examples of the response of the GIS to queries entered by users are given in two parts. The first shows how operations may be carried out using set theory on the spatial database. The second shows how spatial and relational databases may be linked to provide a means of interrogating the GIS. Both parts start from a menu which is displayed on running the GIS program (Fig 11 ).

#### 4.3.1 Operations on spatial database using set theory

Figures 12 to 18 show selected results from a land resources evaluation for the Matlock area. Parameters considered were geology, elevation, soils and land use. Figure 12 shows a quadtree image of areas where grit is the parent material. It highlights the way in which the quadtree representation can save space by storing maximal blocks at different levels within the tree. The northern quarter of figure 12 is stored as only 6 quadtree nodes for the 6 maximal blocks. The complex boundary of the grit requires considerably more nodes to be accurately represented. A characteristic feature of quadtree encoded images is the tendency for smaller blocks at the edges of regions and alrger ones towards the centre of homogeneous areas.

The ten binary array images of contours were quadtree encoded and processed by an edge finding routine to produce a contour plot resembling the original contour map (fig 13). Figure 14 shows land below 800 feet above sea level. This map was produced by the union of all height maps for intervals above 800 feet to produce the map 'height_above800'; this was then complemented to produce areas not above 800 feet. Following each union operation maximal blocks are formed since four small blocks with a common ancestor may be produced.

Figure 15 indicates areas of acid brown earth soils, which occur in two regions on the valley sides. Note that four smaller blocks are not always merged into a single large block, if they do not have a common ancestor in the quadtree structure.

Figure 16 indicates woodland areas. This is a fairly complex image since it contains a large number of small polygons. A relatively large number of nodes are required to store this image, many of which are nodes at the lowest level. This results in the quadtree structure being less efficient at storing many small regions rather than a few extensive areas.

Figure 17 shows only those areas which possess all the above characteristics: it displays only regions with grit geology and brown earth soils which are within the height limit and which are not covered by woodland. Figure 17 was produced by the GIS query:

P brown_earth I grit I height_below800 I N woods

where the P operator stands for plot map. The N operator produces the comoplement of the woods map since we are interested in those areas not covered by woods; this is then intersected with the height map, the results of this is intersected with the grit geology map and finally an intersection is made with the brown earth soils map. At each stage those areas which do not share all the attributes are excluded. One of the major advantages of using linear quadtrees is the speed at which set operations such as these are performed; the above query took only a few seconds on a multi-user computer system, though the plotting was carried out separately.

Figure 18 shows land in the study area between 300 and 400 feet generalised by omitting some details. In this case, the smallest size blocks, present in other images, have been omitted. Figure 18 shows the same area after further generalisation by omitting the two smallest size blocks. This illustrates the variable resolution of the quadtree structure: it is equivalent to a descent of the tree which stops one or two levels above the terminal or 'leaf' nodes. Generalisation of quadtree images is therefore a strightforward operation. As the storage required for generalised images is usually much less than for the complete images, generalised images can provide a quick facility for browsing.

#### 4.3.2 Operations involving the relational database

Requests to the relational database are formed by the query parser at run-time as a response to a user query, and are passed to the relational database. The results obtained from the relational database can then be processed by other parts of the GIS. As an example of this, consider the case when a user requests that all objects of type "woods" are discovered and placed in the database.

Firstly the GIS searches the relational database for an object called "woods" and retrieves the definition of this object. Secondly, the definition is passed to the query parser which, with the aid of the set operations, processes the query and forms a quadtree as a result (Fig 16). Thirdly, a new relation is formed, if one does not already exist, to store the new information on the objects of type 'wood'. Fourthly the new quadtree is passed to the component labelling function which recognises and automatically labels the separate parts of the quadtree (Fig 20). Fifthly, a tuple is appended to the new relation that contains data corresponding to each individual part or object encountered (fig 21). Finally, information about that class of objects (e.g. number, total size, perimeter length, etc.) is added to the relational database and the initial processing is complete.

After the initial processing of geographic data, users can employ the GIS to answer queries about objects by referring only to the relational database, thus avoiding the need to reprocess quadtree images. For example, the queries:-

find all woods covering an area > 3000

find all woods in the area x1, x2, y1, y2 find total and average size of all woods

could be answered immediately from the relational database.

Aspatial data can be included in the relational database, allowing greater flexibility in the types of geographic data that can be used and in the types of queries that can be addressed (Fig 22). For example:-

find the size of holly wood

what percentage of the woodland areas is coniferous

In this way, two sets of data exist within the GIS. The first is a set of spatial quadtrees, which are required to begin with. The second is a set of relational data which corresponds to the various objects contained within the quadtrees and is generated gradually as the quadtrees are used. It can then be used as an alternative, faster means of satisfying some types of query, since it is already in a processed form. Aspatial data can be added to this, to enhance the flexibility of the system.

## 5 DISCUSSION

Individual users can share data in both databases, or have their own dedicated set of quadtrees and relations or some combination of shared and individual databases as required. This allows users from different disciplines, for example to share the same satellite remote sensing data but to include different types of ground data to meet individual requirements. It offers a flexible approach.

In building the GIS, we have adopted a pragmatic approach in order to get a working system within a relatively short time. We recognise however that the structure of the database is not ideal and represents a balance between various options. For example, it would probably be better to express the relationship between information on a single object and information on a class of the same object as a hierarchy, as in the extended relational model of Codd (1979). The use of such an extended relational database for applications of this nature is currently being investigated as a part of a separate research project in the Department of Computer Studies, University of Leeds.

The need for a colour graphics display to reap the full benefits from the GIS is apparent to users. The speed of access only becomes advantageous when users have interactive access and can see the response to their queries as a colour map on a high resolution display. Such a display would allow us to make far better use of functions such as those for windowing and generalisation of maps which are major benefits of the quadtree representation.

After encoding binary images as linear quadtrees, set theory provides an efficient means of spatial analysis of inter-relationships between various regional components in a given area of study. Relatively complex compound queries may be built as macro instructions and recalled repeatedly to explore patterns and relationships between selected features, using the facilities for combining geographic data from diverse sources. For example, atmospheric, topographic, remote sensing and geophysical data may be analyzed spatially. Socio-economic data pertaining to census districts, residential regions or other areas may be analyzed in relation to topography, remote sensing data or agricultural land use. While the process of encoding binary images as linear quadtrees is relatively slow, access to the quadtree images is relatively fast and efficient.

The function for component labelling allows each object in a quadtree image to be automatically given a unique label. Objects may then be held in a relational database and given attributes. This novel feature allows aspatial data to be added to the GIS such as characterisitcs of a soil, cropping history of a field, population of a census district or whatever. The full potential of the relational database may be used to answer a range of queries regarding objects and their attributes. While the feasability of doing this for regional land evaluation in a small study area has been illustrated, the work has been carried out on a multi-user university computer and there are various constraints on its use which restrict the size of area that may be studied. Much larger areas have been used however to test the efficiency of the GIS and have shown that its performance remains relatively good with larger files.

Although the software has been developed and is currently installed on a VAX 11/780 computer, it is written in the C programming language and could be ported to a range of super micro computers relatively easily. This would provide facilities for local processing by either using a workstation attached to a mini-computer orby using a stand-alone system with a relational database management system.

## 6 CONCLUSIONS

The pilot study described here demonstrates the feasibility of building a GIS using linear quadtrees to store spatial data and a relational database to store objects to enhance flexibility and increase the level of performance of the GIS. Linear quadtrees provide rapid access to multiple binary images and give the GIS performance characteristics which are satisfactory for interactive analysis in geographical research. Use of an existing relational database provides a means for storing objects and is comparatively easy and quick to implement. It gives acceptable levels of performance though it is stressed that tests have so far been carried out on a relatively small set of data. Full evaluation of the GIS will require construction of a larger database of images and objects.

One advantage of using an existing relational database is the comparative ease and speed with which a flexible and powerful pilot GIS can be developed to handle spatial and aspatial data. The pilot GIS provides opportunities for evaluating the potential of GIS in a range of different applications and has thus enabled geographers and other field scientists to further their knowledge and appreciation of GIS. This advantage must be considered in relation to inherent limitations in adapting existing relational technology to specific needs and in particular to how closely it can be adapted to achieve satisfactory levels of performance for various users of the GIS. At this stage, the benefits appear to outweigh the limitations at least in the short term. In the long term, it may prove to be more efficient to build special object databases which are tailored explicitly to the needs of users of geographic data.

## 7 ACKNOWLEDGEMENTS

Maps used in this study were based upon the 1971 Ordnance Survey

1/10560 map with the permission of the HMSO, Crown Copyright Reserved. We thank the University of Leeds for providing a one year research assistantship for M. Gahegan and a three year post-graduate studentship for N. Stuart for this work.

## 8 REFERENCES


Abel, D.J. 1984 A B-tree structure for large quadtrees. Computer Vision, Graphics & Image Processing, Vol 27, pp 19-31.

Abel, D.J. 1985 Some elemental operations on linear quadtrees for geographic info systems. Computer Journal, vol 28, No1 pp73-77.

Abel, D. J. 1986 Bit-interleaved database access keys. In: B.M. Diaz & S.B.M. Bell (eds) Spatial data processing using tesseral methods. Natural Environment Research Council, Swindon. pp 163-178

Abel, D.J.& Smith, J.G. 1983 A data structure and algorithm-based on a linear key for a rectangular retrieval problem. Computer Vision, Graphics & Image Processing, Vol.24,pp1-13

Abel, D.J. & Smith, J.G. 1986 A relational GIS database accommodating independent partitioning of the region. In Proceedings Second International Symposium of Spatial Data Handling, Seattle,Washington, July 5-10. 213-224

Bell, S. B. M. Diaz, B. M. Holroyd F. C. & Jackson M. J. 1983 Spatially referenced methods of processing raster and vector data. Image and Vision Computing 1(4):211-220

Christian, C. S. & Stewart, G. A. 1968 Methodology of integrated surveys. In: Aerial surveys and integrated studies. Proceedings of the Toulouse Conference, UNESCO, Paris, pp 233-280.

Codd, E.F. 1970 A relational model of data for large shared data banks. Comm of ACM Vol 13 no 6,pp 377-387.

Codd, E.F. 1979 Extending the database relational model to capture more meaning. ACM Trans. Database Systems Vol 4 No 4 pp 397-434.

Crain, I.K. & MacDonald, C.L. 1984 From land inventory to land management : the evolution of an operational GIS. Cartographica 21, 40-46.

Date, C.J. 1986 Database systems. Addison-Wesley,(4th edition),pp 209-228

Dangermond, J. 1986 GIS trends and experiences. In: Proc. Second International Symposium on Spatial Data Handling, Seattle, Washington, July 5-10, 1986 pp 1-4.

Finkel, R.A. & Bentley, J.L. 1974 Quad trees: a data structure for retrieval on composite keys. Acta Inf. 4, 1, pp 1-9

Frost, R.A. 1986 Introduction to knowledge base systems. Collins,London. pp 677.

Gahegan, M & Hogg, J. 1986 A pilot geographical information system based on linear quadtrees and a relational database for regional analysis. In: B.M. Diaz & S.B.M. Bell (eds) Spatial data processing using tesseral methods. Natural Environment Research Council, Swindon. pp 213-231.

Gargantini, I. 1982 An effective way to represent quadtrees. Communications of A.C.M. vol 25 No. 12 pp 905-910.

Harris, R. 1986 Satellite remote sensing: high spatial resolution.

Progress in Physical Geography, Vol 10 No 4 p579-586

Hogg, J. Gahegan, M. & Stuart,N. 1986 Evaluation of regional land resources using geographic information systems based on linear quadtrees. International Archives of Photogrammetry and Remote Sensing, Vol 26, Part 7/2, pp 917-925

Hogg, J. & Gahegan, M. 1986 Regional analysis using geographic information systems based on linear quadtrees. Intenrational Archives of Photogrammetry and Remote Sensing, Vol 26, Part 4, pp 113-124.

Klinger, A. 1971 Patterns and search statistics. In: Optimising Methods in Statistics by J. S. Rustagi (ed) Academic Press, London, pp 303-337.

Light, D. L. 1986a Mass storage estimates for the digital mapping era. Photogrammetric Engineering and Remote Sensing. Vol 52, No 3, pp419-425.

Light, D.L. 1986b Planning for optical disk technology in digital cartography. Photogrammetric Engineering and Remote Sensing. Vol 52,No 4, pp 551-557.

Marble, D & Peuquet, D 1983 Geographic information systems and remote sensing. Chapt 22 in Manual of Remote Sensing, American Society of Photogrammetry, Falls Church, Va. 2nd Ed. Vol 1.

Mitchell, C. 1973 Terrain evaluation. Longman Group, London, pp 220.

Morton,G. 1966 A computer oriented geodetic data base and a new technique in file sequencing. IBM Canada Ltd, unpublished report.

Oliver, M.A.& Wiseman,N.E. 1983 Operations on quadtree encoded images. The Computer Journal, Vol.26, No 1, pp 83-91.

Oxborrow, E. 1986 Databases and database systems: concpets and issues. Chartwell-Bratt Ltd, Bromley, Kent, U.K. p254

Peuquet, D.J. 1984 A conceptual framework and comparison of spatial data models. Workshop W2, International Symposium on Spatial Data Handling, 1984, Zurich, pp.80.

Rosenfeld, A. Samet, H Shaffer, C. Webber, R.E. 1982 Application of hierarchical data structures to geographical information systems. TR-1197 Computer Vision Laboratory, University of Maryland. pp 160

Rosenfeld, A. Samet, H. Shaffer, C.A. Webber R.E. 1983 Application of heirarchical data structures to geographical information systems: Phase II. TR-1327 Computer Vision Laboratory, University of Maryland, pp76

Samet, H. 1984 The quadtree and related hierarchical data structures. ACM Computing Surveys 16, 2:187-260.

Samet,H. Rosenfeld, A. Shaffer, C.A Nelson, R.C. Huang, Y.G. 1984 Application of hierarchical data structures to geographical information systems: Phase III CS-TR-1457 Center For Automation Research, University of Maryland.

Samet H. Rosenfeld A. Shaffer,C.A. Nelson,R.C. Huang,Y.G. Fujimura, K.

1985 Application of hierarchical data structures to geographical information systems: Phase IV CS-TR-1578 Computer Science Department, University of Maryland, pp 61.

Samet, H. & Tamminen M. 1985 Computing geometric properties of images represented by linear quadtrees. IEEE Trans. Patern Analysis and Machine Intelligence, 7,1

Samet, H., Shaffer, C.A., Nelson, R.C. Yuan-geng Huang, Kikua Fujimura & Rosenfeld, A. 1986 Recent developments in quadtree-based geographic information systems. In Proc. Second International Symposium on Spatial Data Handling, Seattle, Washington, July 5-10, 1986 pp.15-32.

Smith,T.R. & Pazner,M.I. 1984 A knowledge based system for answering queries about geographical objects.Proc. 9th William T. Pecora Memorial symposium on spatial information technologies for remote sensing today and tomorrow. pp 286-289.

Smith, T.R. & Peuquet, D.J. 1985 Control of spatial search for complex queries in a knowledge-based geographic information system. Proc. International Remote Sensing Society/CERMA Conference on Advanced Technology for Monitoring and Processing Global Environmental Data, London, 1985, pp 439-453.

Steiner, D & Gilgen, H. 1984 Relational modelling as a design and evaluation techique for interactive geographical data processing. Proc. Int.Symp. on Spatial Data Handling, Zurich, Switzerland. pp 61-77

Stonebraker, M. (ed) 1986 The Ingres Papers: Anatomy of a relational database system. Addison-Wesley Publishing, Reading, Mass.

Stonebraker, M. Wong, E. Kreps, P. & Held, G. 1976 The design and implementation of Ingres. ACM Trans. Database Systems. Vol 1 No 3 pp 189-222.

Tomlinson, D.J. 1984 Geographical information systems - a new frontier. Proc. Int. Symp. Spatial Data Handling, 1984, Zurich.

Townshend, J. R. G. (ed) 1981 Terrain evaluation and remote sensing. George Allen and Unwin, London, pp 231.

Ullman, J.D. 1982 Principles of database systems. Computer Science Press, London. p 483.

Young, J.A.T. 1986 A U.K. geographic information system for environmental monitoring, resource planning and management capable of integrating and using satellite remote sensing data. Monograph, Department of Geography, University of Edinburgh.

## 9 CAPTIONS FOR FIGURES

Figure 1 A region such as an area of forest

Figure 2 A binary array of pixels to represent the region in Figure 1.

Figure 3 Recursive sub-division of binary image into four quadrants.

Figure 4 A quadtree of the region in which the root represents the whole image and the four sons of the root node correspond to the four quadrants.

Figure 5 Arrangement of coordinates and linear keys used in this study.

Figure 6 The structure and major components of the pilot GIS showing the relationships between each.

Figure 7 The structure of the query interface to the GIS.

Figure 8 The minimum and maximum X and Y coordinates to define a rectangle that neatly surrounds an object

Figure 9 The 17 operators that form part of the query language.

Figure 10 The location of the study area, Matlock, Derbyshire.

Figure 11 The menu from which users select the various options.

Figure 12 Quadtree image of areas of grit rock in the study area.

Figure 13 Contour plot produced by finding the edge of the quadtree image for each contour and copying them to the same file.

Figure 14 Land below 800 feet above sea level in the study area.

Figure 15 Land covered by acid brown earth soils in the study area.

Figure 16 The "woods" quadtree showing all areas of woodland on O.S. map SK 26 SE.

Figure 17 Brown earth soils below 800 feet over grit and not covered by woodland.

Figure 18 Land in the study area between 300 and 400 feet generalised by omitting the smallest pixels.

Figure 19 Land in the study area between 300 and 400 feet generalised by omitting the smallest and the second smallest pixels.

Figure 20 The result of the component labelling operation on the "woods" quadtree.

Figure 21 Data corresponding to each object in the "woods" quadtree is held as a tuple in the relational database with X and Y coordinates and number of pixels within each object (i.e. area).

Figure 22 Aspatial data may be added to the tuple as required by the user, such as type of woods, name, ownership.

Table 1 A list of linear keys produced by encoding the binary image of a region shown in figure 1 as a linear quadtree.

--------end--------

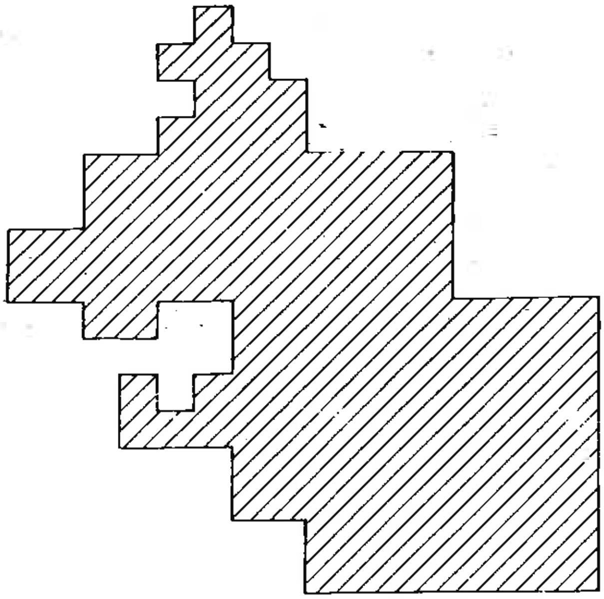

Figure 1 A region such as an area of forest

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

Figure 2 A binary array of pixels to represent the region in Figure 1.

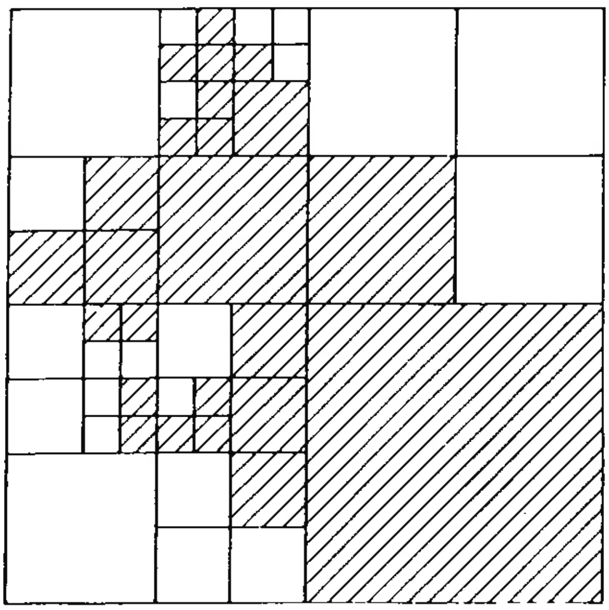

Figure 3 Recursive sub-division of binary image into four quadrants.

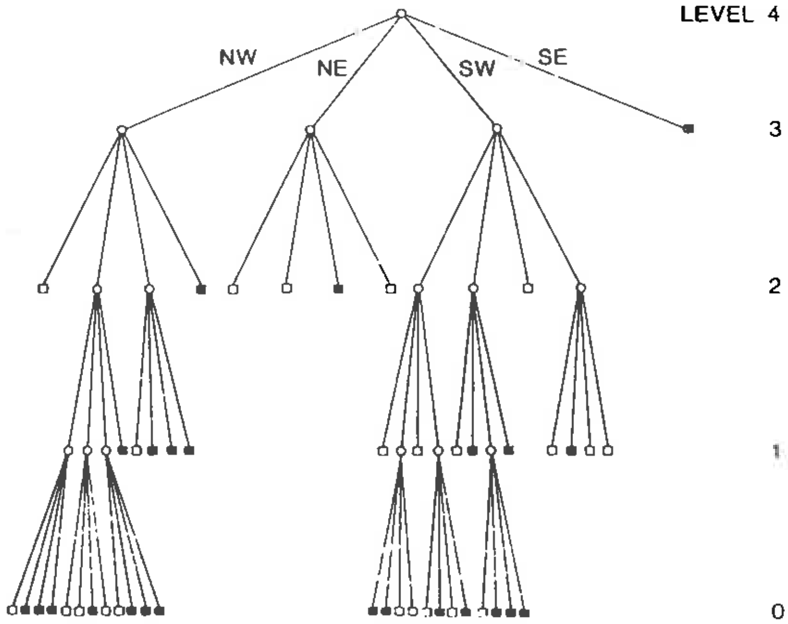


Figure 4 A quadtree of the region

x-axis

| NW | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NE |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 111 | 112 | 121 | 122 | 211 | 212 | 221 | 222 | |
| 1 | 113 | 114 | 123 | 124 | 213 | 214 | 223 | 224 | |
| 2 | 131 | 132 | 141 | 142 | 231 | 232 | 241 | 242 | |
| 3 | 133 | 134 | 143 | 144 | 233 | 234 | 243 | 244 | |
| 4 | 313 | 314 | 321 | 322 | 411 | 412 | 421 | 422 | |
| 5 | 315 | 316 | 323 | 324 | 413 | 414 | 423 | 424 | |
| 6 | 331 | 332 | 341 | 342 | 431 | 432 | 441 | 442 | |
| 7 | 333 | 334 | 343 | 344 | 433 | 434 | 443 | 444 | |
| SW | | | | | | | | | SE |

y-axis

Figure 5 Arrangement of coordinates and linear keys used in this study.

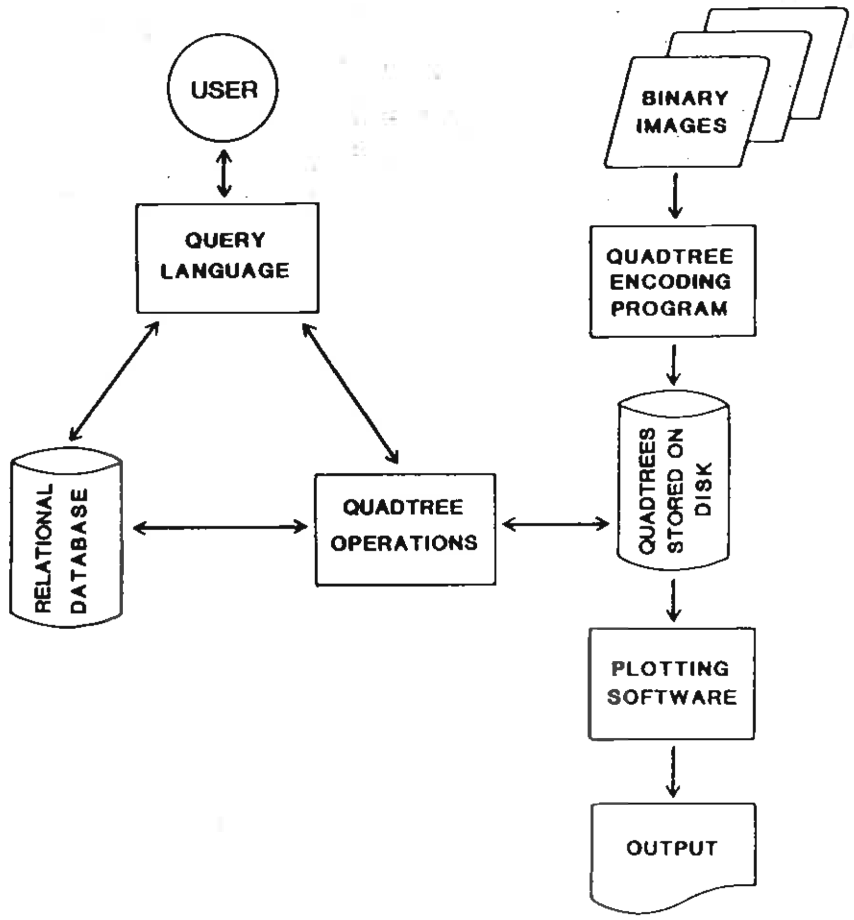


Figure 6 The structure and major components of the pilot GIS showing the relationships between each.

## QUERY STRUCTURE

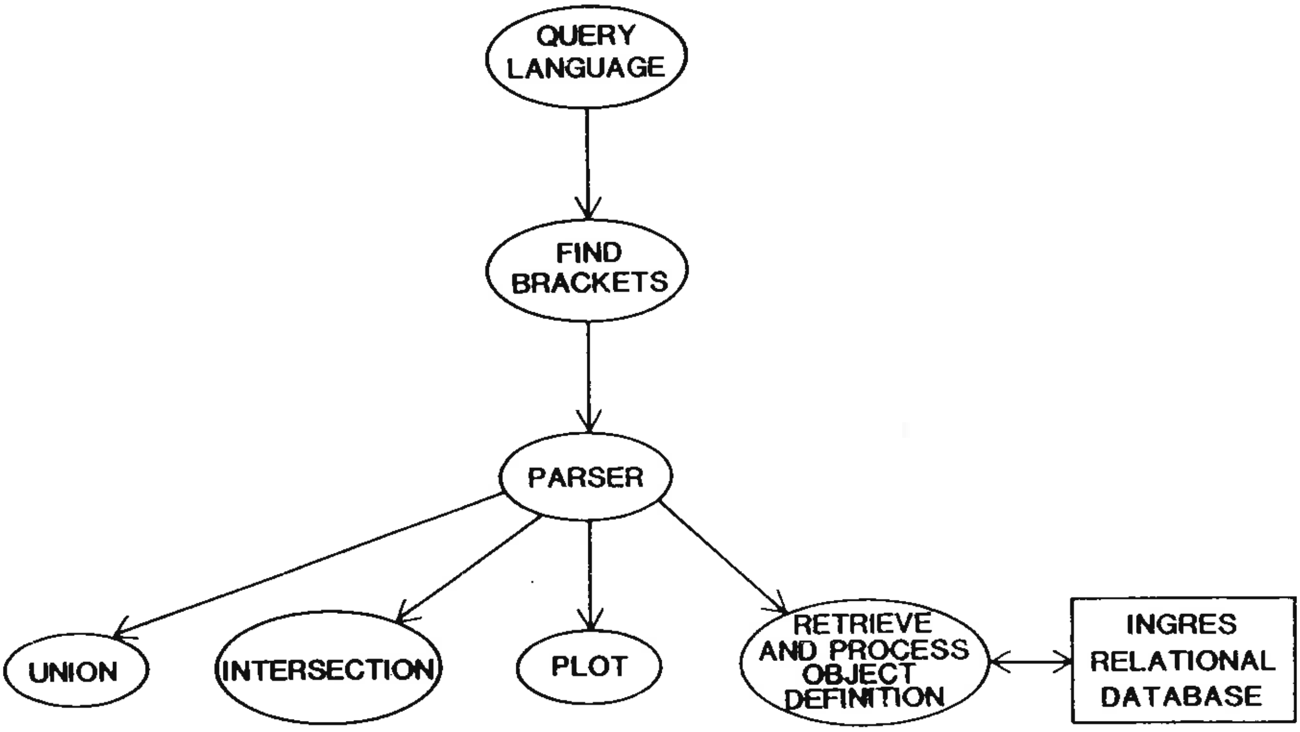


Figure 7 The structure of the query interface to the GIS.

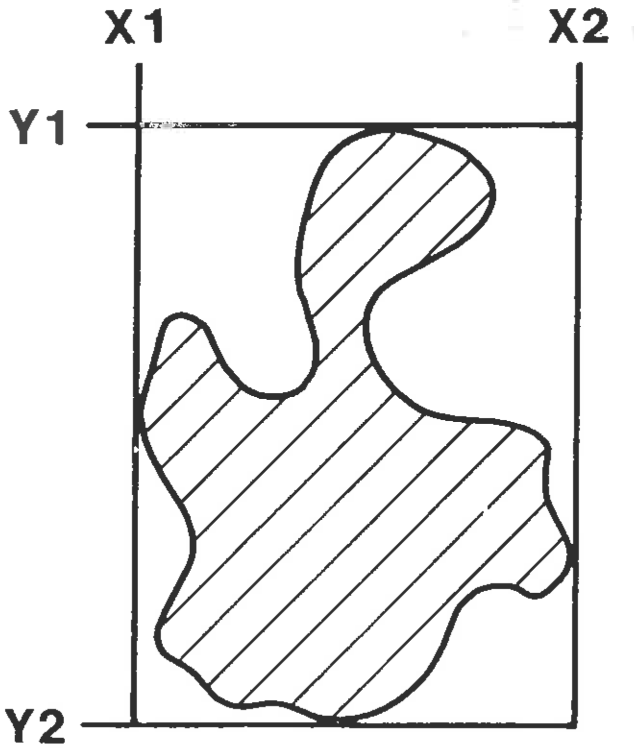


Figure 8 The minimum and maximum X and Y coordinates to define a rectangle that neatly surrounds an object

List of operators available:

---

A ...To ALTER PROTECTION of a quadtree
B ...To calculate the length of the BORDER of a quadtree
C ...To CHANGE the WORKING DIRECTORY
D ...To DELETE a quadtree
E ...To change the scale of resolution of the system
H ...To get detailed use of help facility
I ...To form the INTERSECTION of two quadtrees
L ...To LIST the quadtrees in the current working directory
N ...To form the COMPLIMENT of a quadtree (NOT)
O ...To declare an OBJECT DEFINITION
P ...To send the result of a query to the PLOTTER
Q ...To QUIT from the query language
R ...To RETRIEVE an object quadtree
S ...To provide STATISTICAL INFORMATION on the query
U ...To form the UNION of two quadtrees
W ...To make a WINDOW into a quadtree
X ...To enter objects into a RELATION

Press return to continue

Figure 9  The 17 operators that form part of the query language.

Location of study area. Matlock, Derbyshire. O.S. sheet SK 26 SE


Figure 10 The location of the study area, Matlock, Derbyshire.

```
GEOGRAPHIC INFORMATION SYSTEM
=============================

Based on the pointerless quadtree approach.

This version for 512 x 512 resolution images.


         OPTIONS AVAILABLE:

  0.  Select a QUADTREE for study
  1.  Get COLOUR of a NODE
  2.  Get FATHER of a node
  3.  Get SONS of a node
  4.  Get NEIGHBOUR of a NODE
  5.  WALK in a DIRECTION
  6.  WALK in a DIRECTION -OBSERVE
  7.  To use the QUERY LANGUAGE
  8.  To enter INGRES and use QUEL
  9.  To QUIT

ENTER CHOICE....7
```

Figure 11 The menu from which users select the various options.

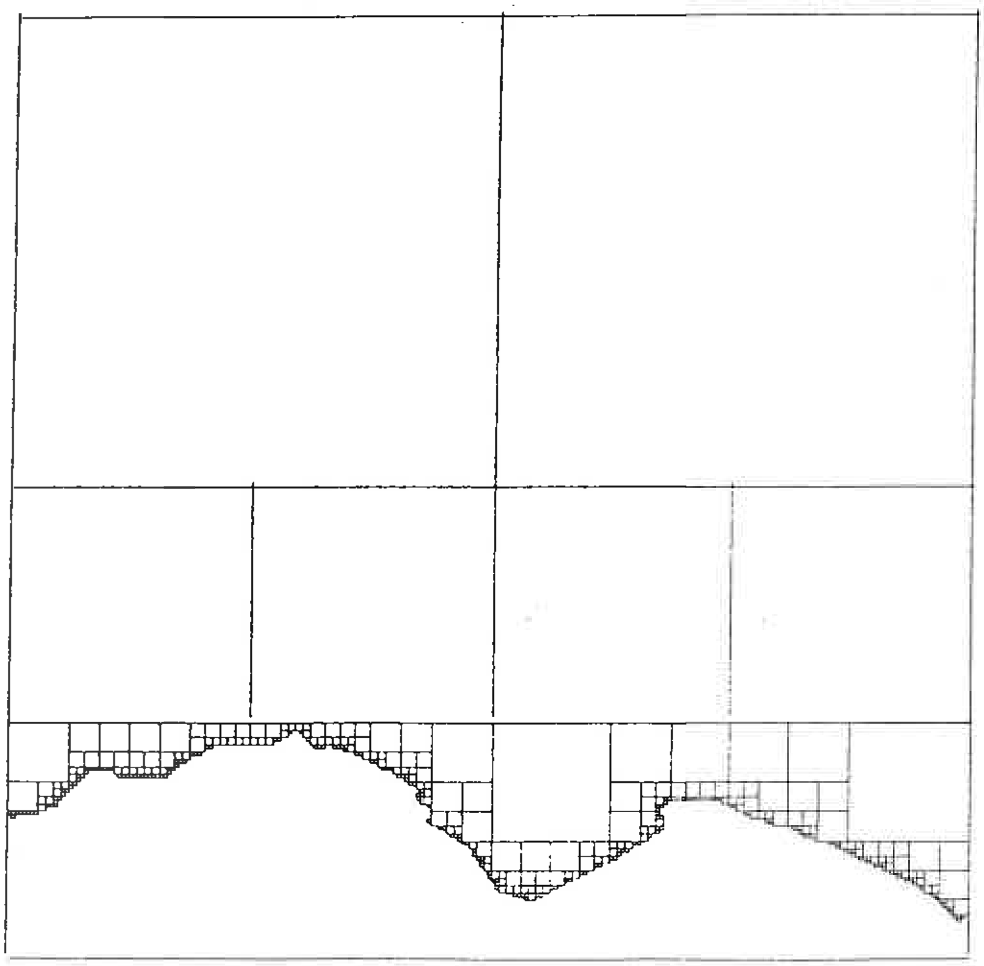

Figure 12 Quadtree image of areas of grit rock in the study area.

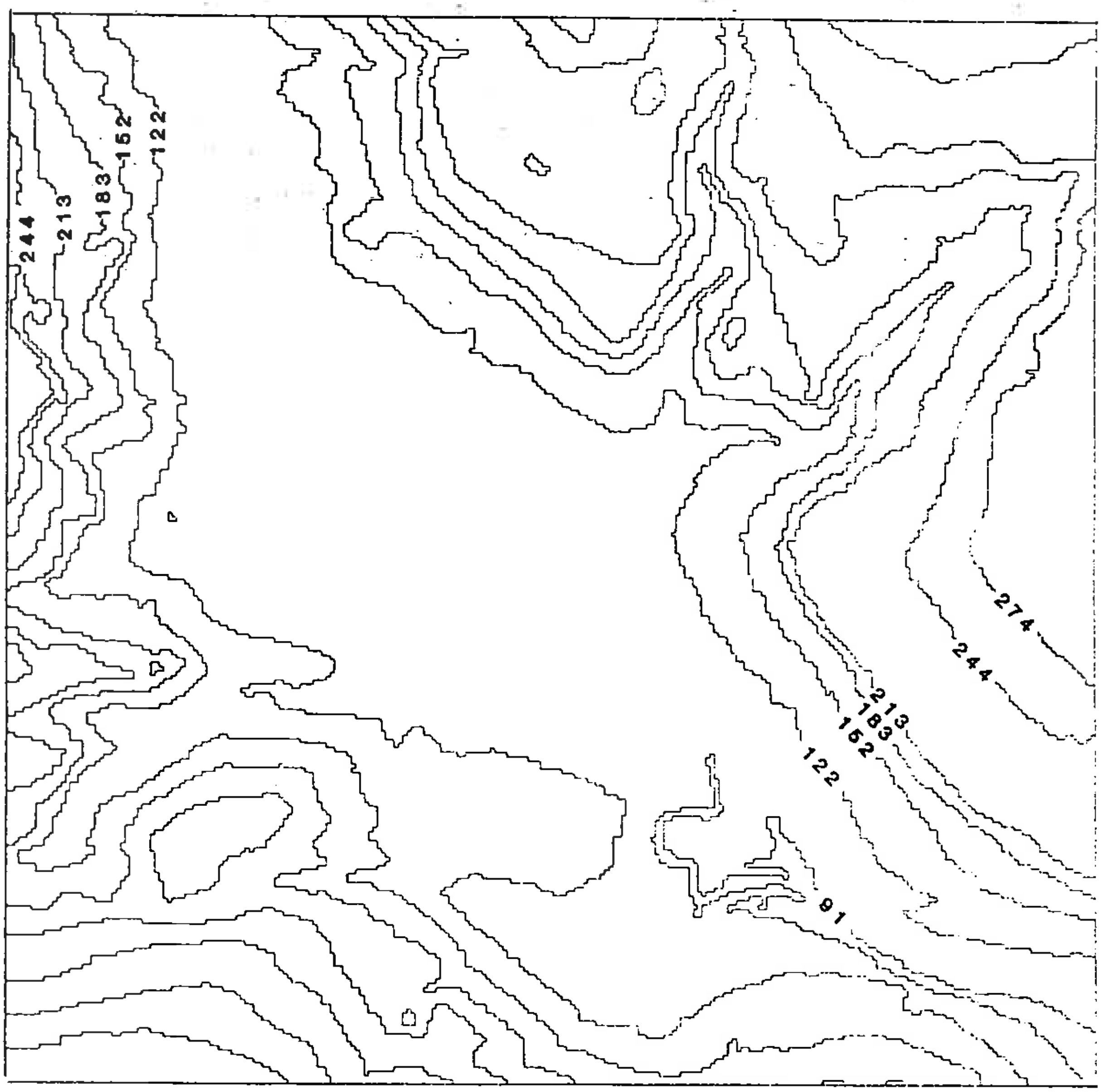


Figure 13 Contour plot produced by finding the edge of the quadtree image for each contour and copying them to the same file.

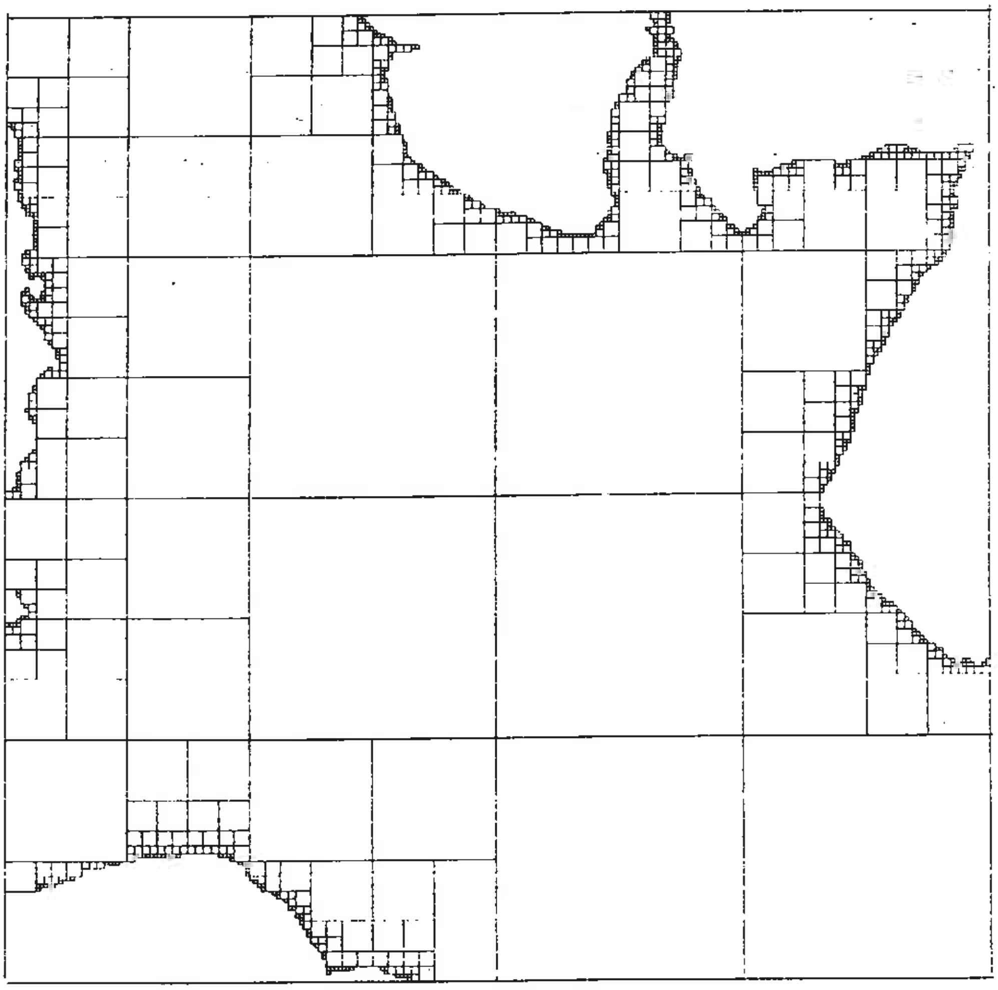

Figure 14 Land below 800 feet above sea level in the study area.

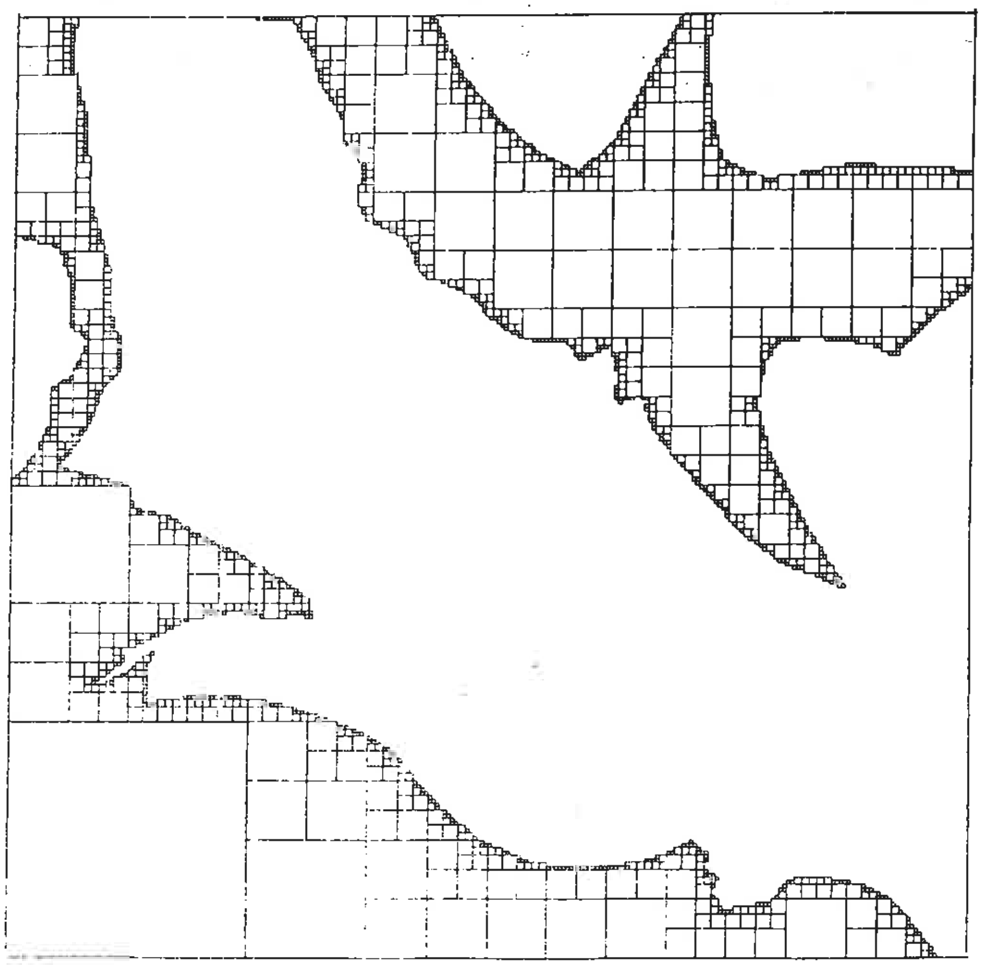

Figure 15 Land covered by acid brown earth soils in the study area.

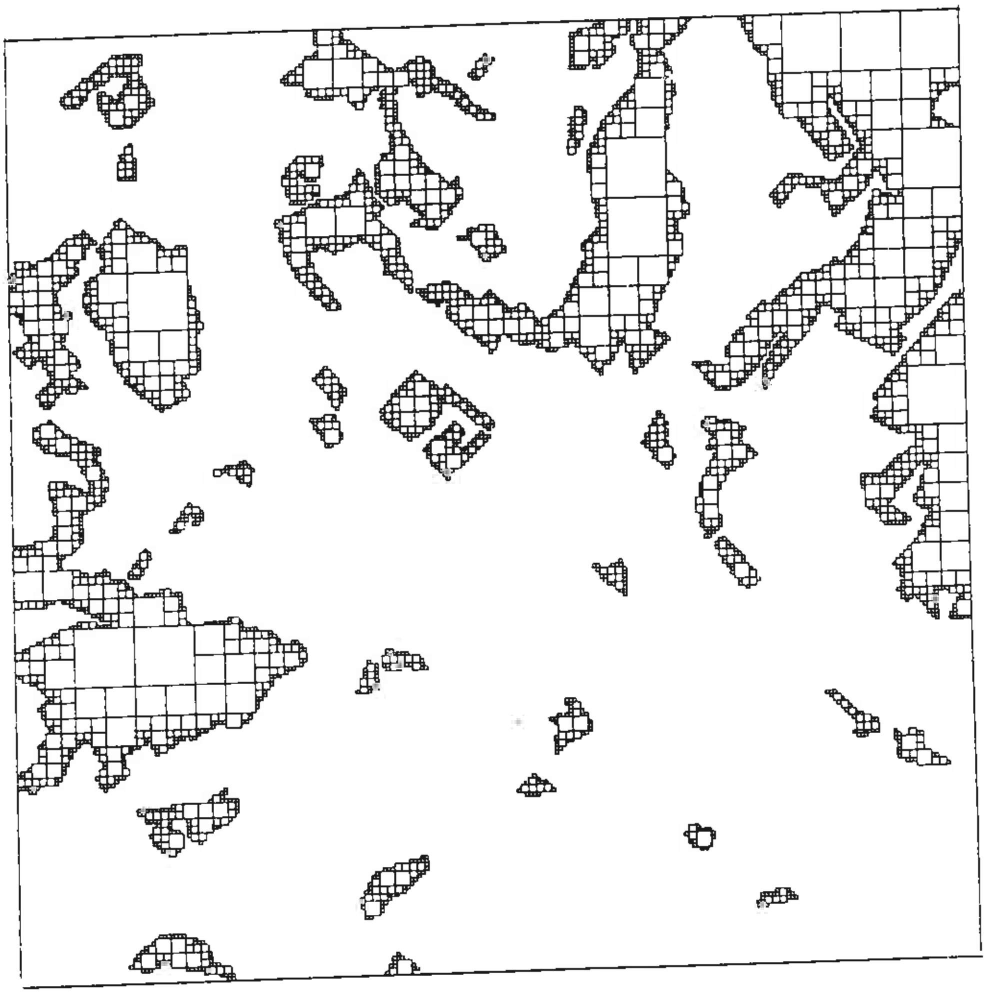

Figure 16 The "woods" quadtree showing all areas of woodland on O.S. map SK 26 SE.

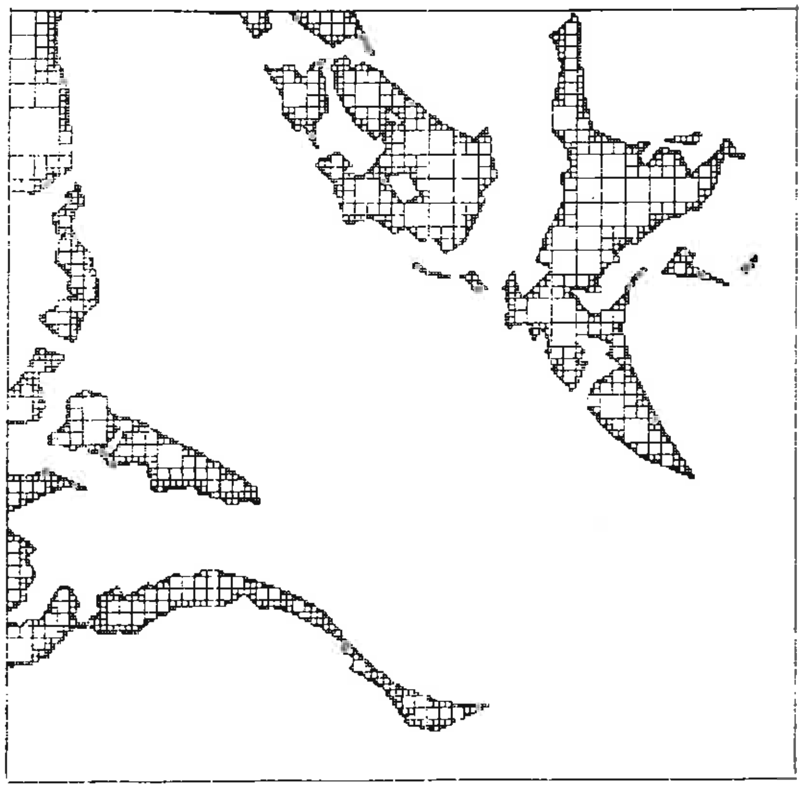

Figure 17 Brown earth soils below 800 feet over grit and not covered by woodland.

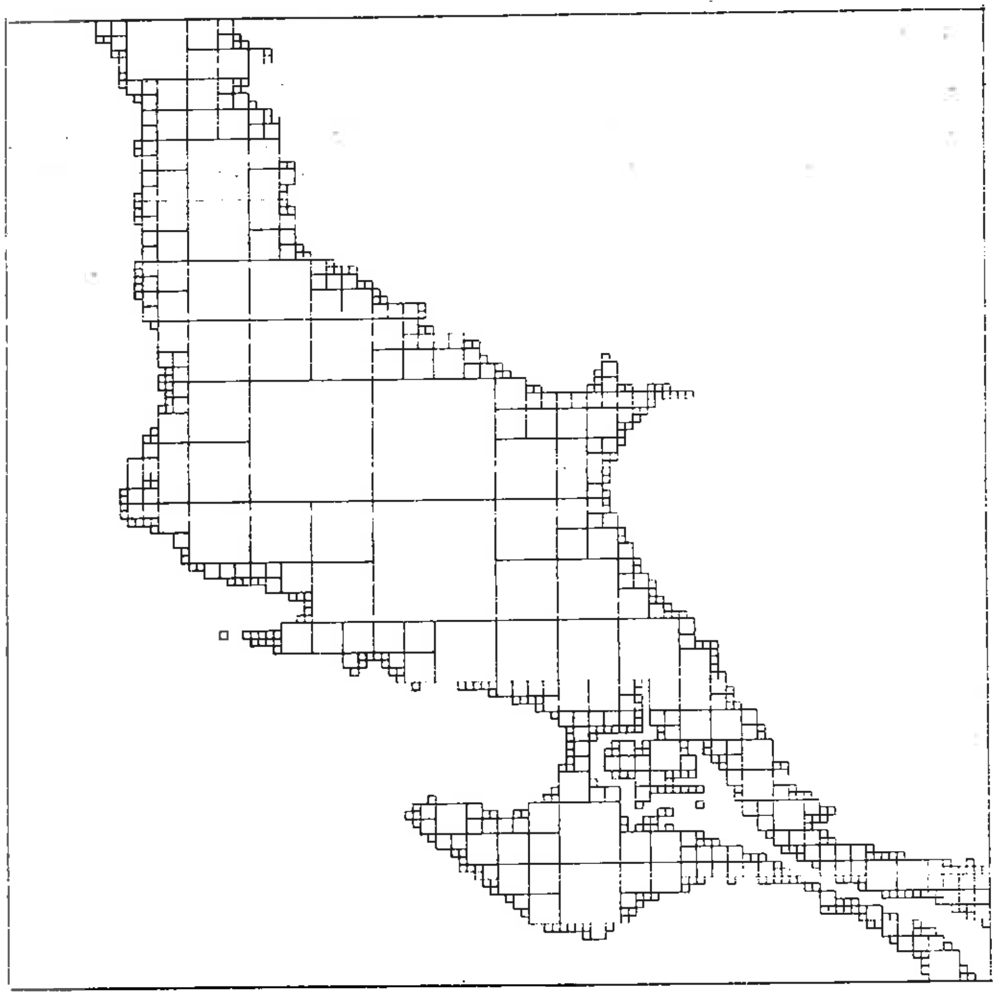

Figure 18 Land in the study area between 300 and 400 feet generalised by omitting the smallest pixels.

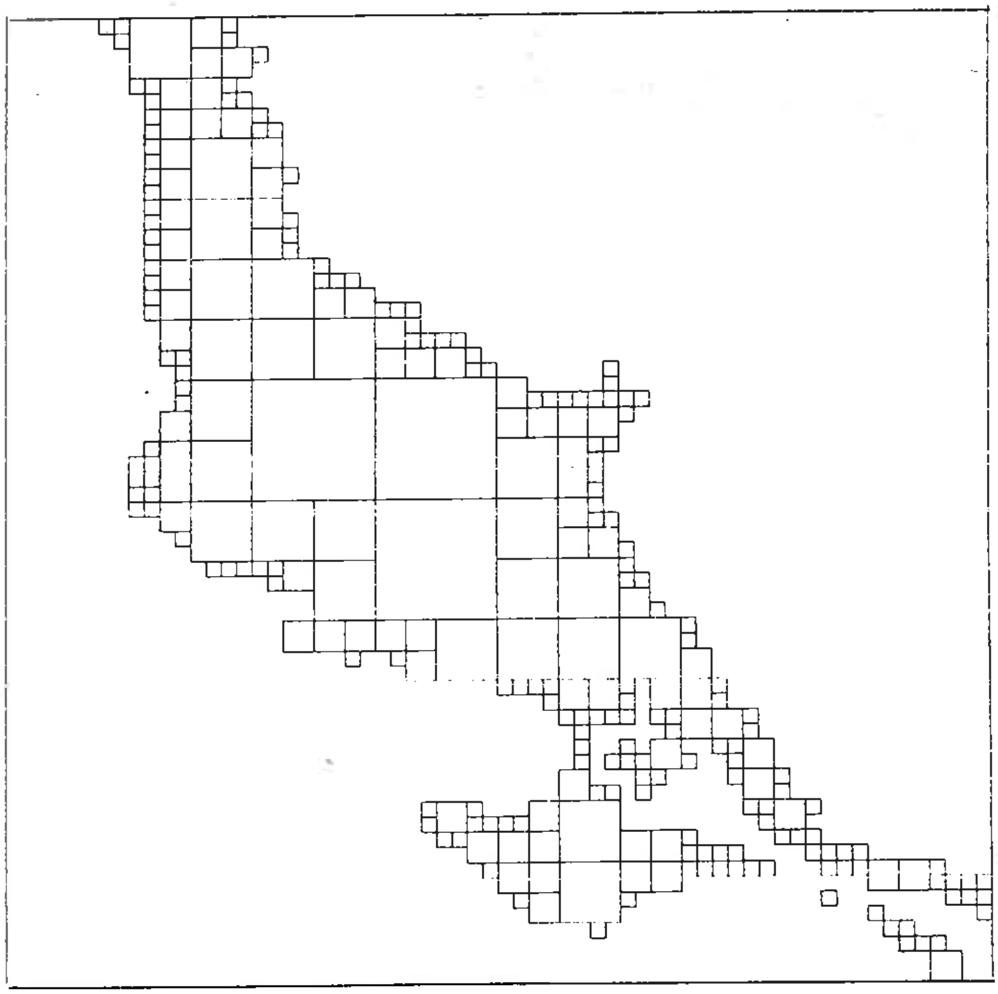

Figure 19 Land in the study area between 300 and 400 feet generalised by omitting the smallest and the second smallest pixels.

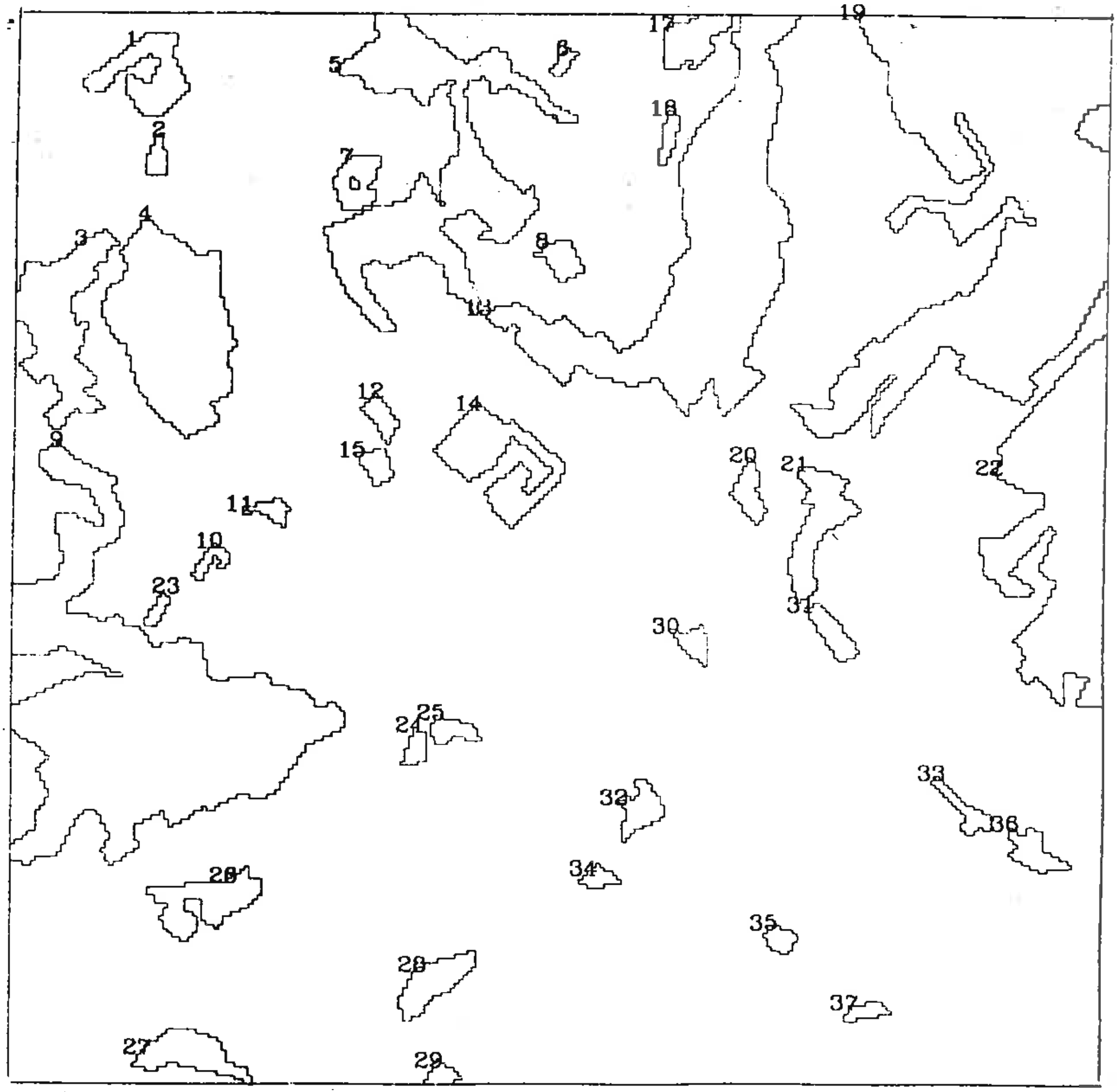


Figure 20 The result of the component labelling operation on the "woods" quadtree.

woods relation

| label | x1 | x2 | y1 | y2 | size |
|---|---|---|---|---|---|
| 1 | 15 | 40 | 5 | 25 | 241 |
| 2 | 30 | 35 | 29 | 39 | 37 |
| 3 | 0 | 24 | 52 | 100 | 529 |
| 4 | 20 | 52 | 49 | 102 | 1056 |
| 5 | 72 | 131 | 0 | 76 | 1383 |
| 6 | 124 | 131 | 8 | [illegible] | 22 |
| 7 | 74 | 85 | 34 | 47 | 99 |
| 8 | 121 | 133 | 54 | 64 | 67 |
| 9 | 0 | 78 | 103 | 204 | 3066 |
| 10 | 42 | 51 | 127 | 135 | 32 |
| 11 | 54 | 65 | 116 | 123 | 36 |
| 12 | 81 | 90 | 91 | 103 | 49 |
| 13 | 109 | 183 | 0 | 96 | 2196 |
| 14 | 99 | 129 | 93 | 112 | 237 |
| 15 | 80 | 89 | 104 | 113 | 48 |
| 16 | 110 | 128 | 107 | 123 | 123 |
| 17 | 151 | 167 | 0 | 13 | 140 |
| 18 | 150 | 155 | 23 | 36 | 38 |
| 19 | 182 | 256 | 0 | 101 | 3775 |
| 20 | 168 | 177 | 106 | 122 | 68 |
| 21 | 182 | 199 | 108 | 141 | 233 |
| 22 | 227 | 256 | 76 | 165 | 1500 |
| 23 | 31 | 37 | 139 | 147 | 24 |
| 24 | 91 | 97 | 171 | 180 | 35 |
| 25 | 98 | 110 | 168 | 174 | 48 |
| 26 | 32 | 59 | 204 | 222 | 221 |
| 27 | 29 | 57 | 243 | 256 | 178 |
| 28 | 91 | 109 | 224 | 241 | 141 |
| 29 | 97 | 106 | 250 | 256 | 31 |
| 30 | 154 | 163 | 146 | 156 | 45 |
| 31 | 187 | 199 | 141 | 155 | 71 |
| 32 | 142 | 153 | 183 | 198 | 85 |
| 33 | 216 | 230 | 183 | 196 | 52 |
| 34 | 133 | 143 | 203 | 209 | 32 |
| 35 | 177 | 185 | 218 | 225 | 37 |
| 36 | 234 | 249 | 194 | 204 | 78 |
| 37 | 196 | 207 | 236 | 241 | 34 |

Figure 21 Data corresponding to each object in the "woods" quadtree is held as a tuple in the relational database with X and Y coordinates and number of pixels within each object (i.e. area).

| label | name | type |
| --- | --- | --- |
| 1 | Holly wood | mixed |
| 3 | Hill wood | deciduous |
| 4 | Hillcar wood | mixed |
| 5 | Northwood carr | deciduous |
| 19 | Seventy acre plantation | coniferous |

Figure 22 Aspatial data may be added to the tuple as required by the user, such as type of woods, name, ownership.

## Linear Keys

1212 1213 1214 1223 1232 1233 1234 124 132 133 134 14

23

3121 3122 3142 3144 322 3232 3233 3234 324 342

4

Table 1 A list of linear keys produced by encoding the binary image of a region shown in figure 1 as a linear quadtree.