UNLIMITED

①

ROYAL AIRCRAFT ESTABLISHMENT

*

Technical Report 81071

June 1981

AUTOMATIC RELOCATION OF GROUND CONTROL POINTS IN LANDSAT IMAGERY

by

A.H. Benny

*

Procurement Executive, Ministry of Defence
Farnborough, Hants

8

310450

REPORT DOCUMENTATION PAGE

Overall security classification of this page

UNCLASSIFIED

As far as possible this page should contain only unclassified information. If it is necessary to enter classified information, the box above must be marked to indicate the classification, e.g. Restricted, Confidential or Secret.

1. DRIC Reference (to be added by DRIC)	2. Originator's Reference RAE TR 81071	3. Agency Reference N/A	4. Report Security Classification/Marking UNCLASSIFIED

5. DRIC Code for Originator 7673000W	6. Originator (Corporate Author) Name and Location Royal Aircraft Establishment, Farnborough, Hants, UK
5a. Sponsoring Agency's Code N/A	6a. Sponsoring Agency (Contract Authority) Name and Location N/A

7. Title

Automatic relocation of ground control points in Landsat imagery

7a. (For Translations) Title in Foreign Language

7b. (For Conference Papers) Title, Place and Date of Conference

8. Author 1. Surname, Initials Benny, A.H.	9a. Author 2	9b. Authors 3, 4	10. Date June 1981	Pages 35	Refs. 12
11. Contract Number N/A	12. Period N/A	13. Project	14. Other Reference Nos. Space 598		

15. Distribution statement:

(a) Controlled by – Head of Space Department, RAE

(b) Special limitations (if any) –

16. Descriptors (Keywords) (Descriptors marked * are selected from TEST)

Landsat. MSS. Ground-control-points. Automatic. Relocation. Multitemporal.

17. Abstract

Images obtained from the Landsat multispectral scanner are not provided in any normally recognised map projection system, so it is often necessary for a transformation to be made. This is usually done with the aid of ground control points, whose locations on the ground (or on a map) are accurately known and which can be accurately located within Landsat images. The initial selection and location of such ground control points is usually performed manually.

For repeat – *ie* multitemporal – imagery, the same ground control points can be used. The positions of these points can be relocated within the repeat image either manually or, as described in this paper, by an almost entirely automatic computer process. This process has been found to work speedily and effectively on many repeat images of Britain, usually giving accurate relocation of well over 90% of ground control points.

The process has been developed for use with images of Britain, but it could readily be adapted for worldwide use.

5910/1

RAE Form A143

UDC 629.195 : 339.5 : 531.7.084.2 : 629.19 : 535.61 : 681.3.056 : 526.8

ROYAL AIRCRAFT ESTABLISHMENT

Technical Report 81071

Received for printing 5 June 1981

AUTOMATIC RELOCATION OF GROUND CONTROL POINTS IN LANDSAT IMAGERY

by

A. H. Benny

SUMMARY

Images obtained from the Landsat multispectral scanner are not provided in any normally recognised map projection system, so it is often necessary for a transformation to be made. This is usually done with the aid of ground control points, whose locations on the ground (or on a map) are accurately known and which can be accurately located within Landsat images. The initial selection and location of such ground control points is usually performed manually.

For repeat - or multitemporal - imagery, the same ground control points can be used. The positions of these points can be relocated within the repeat image either manually or, as described in this paper, by an almost entirely automatic computer process. This process has been found to work speedily and effectively on many repeat images of Britain, usually giving accurate relocation of well over 90% of ground control points.

The process has been developed for use with images of Britain, but it could readily be adapted for worldwide use.

Departmental Reference: Space 598

LIST OF CONTENTS

1 INTRODUCTION

Information about the surface of the earth has for long been obtained by a wide variety of methods. With the ability to ascend above the surface has occurred what is now referred to as 'remote sensing' (see Ref 1), the acquisition of information from a distance, first by aerial photography and latterly by earth satellite observation.

Observation of the earth from satellites has been made by photography and by other imaging methods. In some cases the photographs can be physically returned to earth (eg Skylab), but this is not always convenient, and is certainly not so for satellites designed to make many observations over a long period of time. As an alternative to conventional photography, therefore, the earth's surface may be optically scanned by some means and the observations converted to a form which can be transmitted to earth. In this case the data can be reconstituted into a form which resembles a photograph, and which will be referred to in this Report as a 'picture'. This Report is concerned with digital data and pictures provided by the multi-spectral scanner (MSS) on the Landsat series of earth resources satellites.

Images provided by Landsat MSS cover an area approximately 185 km square, viewed vertically downwards, and are often immediately recognisable by a viewer who is familiar with the portion of country depicted. Recognition is usually by means of salient features, such as the shape of coastlines or the position of roads. However, a small amount of spatial distortion is present, so if the precise location of any particular feature, or any exact distance, is required, it is usually necessary to resort to interpretive techniques.

Perhaps the most convenient treatment for Landsat MSS data is to convert or 'transform' it to a known cartographic coordinate system. The information is then presented in a form which is familiar, whose dimensions (scale and other properties) are understood, which may include calibration grid lines and which may be directly compared with a conventional map of the same scale, to establish locations. Satellite imagery may then be studied by a wide variety of users, without their having to be concerned with the nature and details of the recognition and transformation process.

Two distinct forms of information are in general available for precise dimensional evaluation of photographs or photograph-like images. One form of information relates to the details of the imaging system, for example in microphotography the magnification of the system, and in telephotography knowledge of any distortions introduced by the optical system as instanced by the difference between the behaviour of a sliding-back and a conventional camera. The other form of information is knowledge about the object being photographed, for example a measuring rod may be included in a photograph of an archaeological dig to provide the local scale.

The actual process of Landsat image identification and transformation makes use of both types of knowledge, ie those concerning the imaging system and concerning the earth's surface. Certain features of the image can be 'recognised' and these, together with a knowledge of the geometry of the optical imaging process, allow the remainder of the image to be fitted into its correct place.

The recognisable terrestrial features are referred to as 'ground control points' (gcps), and to be usable they must be identifiable both on the satellite image and also on some suitable calibrated map.

1.1 The Landsat MSS image

The Landsat MSS is described in detail in Ref 2. Briefly, the satellite moves in a near polar orbit (inclined at about 80 degrees to the equator) at a nominally constant height (about 900 km) above the earth's surface. As the satellite travels, the surface vertically below it is scanned in a cross-track direction, so that the observations form a raster scan of the earth's surface. The image is resolved into picture elements or pixels of about 80 m square, with some overlap of pixels along the scan line, so that each pixel is about 50 m in that direction. Observations are made at four spectral wavelengths, but this need not be considered here except to note that in practice the most convenient waveband for determining gcps is the so-called band 7, (0.8 to 1.1 microns wavelength) in the near infra-red.

From this description, some properties of the resultant image may be deduced. Firstly, the scale of the image in the direction of satellite travel should be constant, since the near circular orbit provides a near constant speed. Secondly, the scale in the cross-track direction depends upon the nature of the scanning mechanism and the curvature of the earth's surface. (Variations in height of terrestrial features may be ignored since their effect, particularly for the low altitudes of the majority of Britain, is marginal if not undetectable.) Since the natures of these two scan effects are fairly accurately known, they can be compensated for. Other known effects such as the rotation of the earth beneath the satellite can also be allowed for. The resulting image, referred to as 'system corrected', is in the form of an oblique Mercator projection: indeed Colvocoresses[3] has proposed the use of 'Space Oblique Mercator' projection for Landsat imagery.

For many purposes it would be desirable to transform the image to a more conventional map projection, *eg* some form of Transverse Mercator, and this can in principle be done by computer provided the latitude and longitude of some locations of the image are accurately known.

In practice, certain difficulties arise, and these stem from the lack of perfection of the satellite's movement. The orbit is such that the height above the earth's surface varies by approximately ±1% and the vehicle itself is subject to movements in all three of its rotational axes, *ie* it pitches, rolls and yaws. All of these movements are to some extent measured and known, but not with sufficient accuracy to allow the image to be accurately calibrated. A reasonable level of accuracy might be defined such that no pixel is out of place, with reference to the map system, by more than its own dimensions.

1.2 Geometric transformation of Landsat images

Since it is not possible to relate an image to the earth's surface with sufficient precision from available knowledge of the operation and movements of the Landsat imaging system, resort has therefore to be made to knowledge of the subject being imaged. This may be done with the aid of gcps whose position can be determined, both on the image and also on the earth's surface. Since terrestrial features have been surveyed in great detail in Britain, it is convenient to use maps for the accurate determination of the position of gcps on the earth's surface.

When sufficient gcps have been located, both on the image and on a map, some form of mathematical transformation can be used to enable all points in the image to be transformed to the map coordinate system, this process being referred to as geometric transformation.

The identification of gcps, both on an image and on a map, is in the first instance an operation which is done by human beings, as no suitable pattern recognition system has yet been set up to compare an image with a map. (It would seem feasible to perform this operation automatically by computer, for certain types of gcps, for example, selected land-water boundaries.) However the Landsat system is a continuing operation, so repeated images are produced, and it has proved feasible to use a computer to recognise and relocate known gcps in these repeat images.

A distinction should therefore be clearly drawn, between (a) the first identification and location of gcps both on an image and on a map, and (b) the relocation of known gcps in other images. (Relocation on a map is, of course, not needed.) This Report is mainly concerned with (b), and describes an almost entirely automatic computer process, the program GCP.FIND, for relocation of specified gcps within a repeat image.

The subject of geometric transformation has been discussed in the literature, one of the more detailed expositions being that of Shlien.

1.3 Some notes on the program GCP.FIND

Many countries have their own satellite data processing facilities, so this work has been primarily concerned with images of Britain, ie the mainland and surrounding islands, but excluding Northern Ireland and Ireland. However the possibility of use elsewhere has been considered, and those features of the work relevant only to Britain have been noted. It should therefore be possible to adapt the program GCP.FIND for operation in other countries, with fairly limited changes.

The problem of relocating gcps in images is essentially a practical one, and has been treated as such in this work. The processes which are described in this Report are ones which have been found by experience to work satisfactorily for an adequately large proportion of gcps, on a wide variety of images of Britain, the whole operation requiring a reasonably small amount of computer facilities and operating time compared with other operations performed on the image. The methods described are not necessarily the 'best' or the most elegant and, indeed, considering the variety of images to be processed, it is doubtful whether a 'best' method exists.

2 FIRST LOCATION OF GROUND CONTROL POINTS

Since the first identification and location of gcps must be done before their relocation, the former process will be described first. The author of this Report has not performed any of this location work directly himself, but has made several suggestions which have proved of assistance in doing it, and these will be mentioned in this section.

2.1 Ground control point library system

The first gcp work done in Space Department RAE is that described by Williams[5] who selected about 74 locations in Landsat image path 219 row 24 of 2 July 1977. Subsequently, others at RAE have selected sufficient gcps to cover the whole of Britain. Gcps have been identified on Landsat images and on Ordnance Survey 1:50000 maps, locations being referred to the National Grid. Since a correspondence between image and map is required, the procedure results in the provision of two pairs of values for each gcp: the column and row number of a pixel in the relevant Landsat image, and the corresponding National Grid eastings and northings, recorded to an accuracy of about 0.2 mm on the map, for the centre of the selected pixel. Between 50 and 100 gcps are selected for each image, and for the whole of Britain this has involved the selection of about 1200 gcps.

The main object of this work is to create a number of data sets. One of these is a 'master map-location file', ie a list of all gcps together with their National Grid locations. Other data sets are created for each image, containing a list of the 50-100 gcps for that image with their column and row numbers in that image. These data sets are subsequently used to create a set of gcp 'chips' as described in section 3.1.

It may be noted that for Britain it is only necessary to process alternate satellite tracks, as there is a full 50% overlap of adjacent images. Consequently all of Britain can be examined in about 18 images.

Now that the whole of Britain has had gcps selected, the process described as the main subject of this Report allows all other images to have their gcps located almost entirely automatically. Thus the many hundred images with usable portions of Britain can be processed rapidly and almost automatically as required.

2.2 Ground control point location on maps

Originally, gcp location was done by Williams as follows. Gcps were visually identified on image and map, for example road intersections, recognisable curves of rivers, shapes of lakes. For each such feature the nearest image column and row were recorded, together with National Grid easting and northing. Subsequent re-examination of these gcps has shown that the locations were not always as accurate as might be wished, for example since a relative rotation of about fifteen degrees exists between image and map, the exact location on the curve of a river could be appreciably in error. Again, a road intersection could be mistaken for an adjacent one.

Following experience with correlation of patches of images, the present author suggested that small patches of images be prepared, to the same scale as the maps, and on a transparent base, so that each patch could be laid over the map and moved about until the best fit was obtained. Further, since the angle between image and map is known to at least an accuracy of one degree, (section 7.2) the transparency could include a line to indicate the relevant east-west direction of the map. This line could then be held parallel to the map east-west direction and the fit of image to map would then involve only translation in two directions, without rotation. Finally, to aid in the determination of the centre of the centre pixel of the patch, a small circle could be incorporated during the preparation of the transparency. These aids, together with others determined in the light of experience, have enabled gcps to be located extremely accurately, so much so that deficiencies within the image itself have been exposed. These deficiencies relate to the several forms of system correction applied during the preparation of the computer compatible tapes, and are described in Ref 6. Additionally, the occasional absence of scan lines from an image can be detected. For greatest accuracy it may be necessary to start with 'raw' imagery[6].

2.3 The transformation matrix

When a sufficient number of gcps have been located on both image and map, the information may be used to calculate a 'transformation matrix', which enables all image locations to be converted to map references, or vice versa. This has been described by Williams[5] and by Shlien[4]. Various types of matrix can be made, according to the number and accuracy of gcps available, and the accuracy of the required transformation. A first-order matrix allows the image to be rotated, translated and sheared only, whereas a higher-order matrix enables more complicated transformations to be performed, such as correction for non-linearity of the satellite scan mechanism. Present experience indicates that a first-order matrix gives useful results when applied to system corrected data, but that an appreciable improvement can be obtained with a higher-order matrix. Since the latter would also be capable of handling the scan line non-linearity present in non-system-corrected (or 'raw') data, it seems likely that this would form the optimum system for highest accuracy. Other aspects of the use of raw data will be touched on later in this Report.

2.4 Numbering system for ground control points

The whole of Britain (as defined in section 1.3) is mapped by, amongst others, 204 sheets of the Ordnance Survey 1:50000 Second Series. Each gcp has been given a code number consisting of two integers, the first being the number of the OS map on which it occurs and the second an arbitrary serial number on that map. Sometimes, due to overlap of map sheets, a gcp occurs on more than one sheet, and in such cases it is customary to use the lower or lowest of the sheet numbers. Failure to observe this rule carries no adverse consequences in the operation of the computer program but may cause inconvenience when referring to maps. Provision is made for serial numbers from 1 to 99 inclusive, although the average number is less than six. This numbering system is described here mainly so that the detailed operation of the computer program may more easily be understood.

3 RELOCATION OF A GROUND CONTROL POINT

The operations described in section 2, the establishment of a first set of gcps, have been performed manually, and the results have been entered into data files ready for use by a computer. The determination of gcp locations in subsequent images can be done almost completely automatically with the aid of the computer program GCP.FIND, which is the main subject of this Report. This program makes use of a method of relocating individual gcps, this being done by the subroutine LOCGCP (locate gcp), which is described in detail in section 7. The following sections describe how all gcps in an image are located.

3.1 Ground control point 'chips'

The task of establishing the location of a gcp in an image is essentially one of pattern recognition. Any gcp is by definition the description of some terrestrial feature of non-zero size. For example, the definition "the intersection of the M4 and M5 motorways" whilst it refers, as a location, to the point where the centre lines of those roads intersect, in fact only has meaning with reference to the existence of the roads themselves for some short distance at, and to either side of, the intersection. The actual location of a gcp therefore, is determined by the description of one particular point in its surrounding context. The gcp can only be located by locating a surrounding area.

For the first location the pattern recognition problem is that of relating a portion of an image with a portion of a map. In the case of Landsat MSS, the image is in the form of a raster of rectangular pixels, whilst the maps used (OS 1:50000) consist of lines and coloured areas printed on paper, ie in non-digital form. It is possible to devise various methods of relating image and map, and it seemed that the method described in section 2 gave probably the best combination of accuracy with speed for the one thousand or so gcps required. As noted, this method used a computer for the preparation of the photographic patches of image which were compared with the maps. It seems likely that a more automatic method of relating image to map would need considerably more effort to develop than the method described.

For the problem of relocating gcps the situation is greatly eased, as it reduces to the matching of portions of one image with another image, that is, both elements are in the same format, which is already a digital one.

In readiness for the matching process, small 'chips' of the original image are prepared. These chips are subimages, copied from the original image, and consist of 19 rows of 19 columns of pixel values, centred on the gcp nominal column and row location. A 'library' of gcp chips has been prepared and these are available for use by the program GCP.FIND. Each chip forms its own small computer data file and the file name is based on the gcp number, described in section 2.4. Thus gcp number 172,3 has a corresponding chip with file name G172.3 in a specified 'user area' of the disc data storage. Chips were prepared in batches, one batch per image, using the purpose-written program IM.CHIPIM.

The problem of relocating gcps is made much easier than the general pattern recognition problem when it is noted that (a) the satellite height varies very little, hence the size of a gcp is almost identical in every image, and (b) the direction of the satellite path (ie its angle to the local meridian) varies very little, hence the gcp is always orientated at about the same angle. Thus the pattern recognition problem becomes the manageable one of comparing and relating images of almost identical size and orientation. It is true that the object being imaged may change, from one satellite pass to another, eg a field may be ploughed on one occasion and have growing crops on another, and such changes, discussed later, can make the relocation problem more difficult, if not impossible in, for example, the case of a snow-covered scene, or of course a cloud-covered gcp.

In practice, it has been found that gcp relocation can usually be satisfactorily performed by pattern matching a 'chip' from one image, with another image, as described below.

3.2 Pattern matching

The method of establishing the correspondence of chip and new image location is by means of the correlation coefficient, CC , calculated between the 19 by 19 (= 361) pixels of the chip, and a 19 by 19 pixel section of the new image. The correlation coefficient is calculated for any image location by the conventional method:

$$CC = \left(\sum(x_i - x_a)(y_i - y_a)\right) \Big/ \left(\sum(x_i - x_a)^2 \sum(y_i - y_a)^2\right)^{\frac{1}{2}} \qquad (1)$$

where x_i = ith pixel value of the chip

y_i = ith corresponding value for the image

x_a = average of all 361 chip pixel values

y_a = average of all 361 image values

and the summations are conducted for $i = 1$ to 361 .

Expression (1) may be rearranged in the form

$$CC = \left(N\sum x_i y_i - \sum x_i \sum y_i\right) \Big/ \left(\left(N\sum x_i^2 - \left(\sum x_i\right)^2\right)\left(N\sum y_i^2 - \left(\sum y_i\right)^2\right)\right)^{\frac{1}{2}} \qquad (2)$$

where $N = 361$, the total number of x_i or y_i values, and the summation is as before.

Expression (2) is more convenient than (1) for computer evaluation because it does not involve the pre-calculation of the average values x_a and y_a . In subroutine LOCCCF the calculation is seen in the form

$$CC = R1/(R2 \,.\, R3)^{\frac{1}{2}} \qquad (3)$$

Since R2 only contains x_i values, it may be calculated once only, for the particular chip, and will then be available for all further correlation calculations for that chip, whereas the terms R1 and R3 must be recalculated for each position of the chip in relation to the new image.

If either the image or the chip, or both, should be uniform over their 19 by 19 extent, then the calculation of R1 produces the value zero. In the case of a uniform chip, then R2 is also zero, and for a uniform image patch, R3 is zero. In either case the CC calculation is indeterminate, being the result of zero divided by zero. An attempt to divide by zero using the computer invokes a system error message, with cessation of program operation, so this condition must be recognised and avoided. This is done by checking that R is zero and it is necessary to use double precision to ensure that an exact zero is obtained. Such cases are detected by LOCGCP, and CC is set to zero. In practice a uniform chip is never created and none exists in the chip library, but uniform portions of images have been found - these have occurred in cases of scan-line mechanism failure when part of the data has been lost and the system-corrected data has had a constant value inserted instead.

The calculation of R1 and R3 requires a significant amount of computer effort, as each summation involves 361 values, each of which is the result of a multiplication of two numbers. Further, the expression for CC has to be performed many times for each gcp (see below). In fact, the largest single activity performed by the program GCP.FIND is evaluating CC. Any means of reducing this work would therefore be welcome. Barnea and Silverman[7] discuss a class of algorithms for fast digital image registration. It seems, however, that their method is suitable for cases where the images being compared have approximately similar brightness, ie corresponding pixels have similar value. In the case of Landsat imagery the pixel values may vary considerably from image to image although the general pattern remains the same, and the fast registration method does not appear to be applicable. It is of interest to note that Orti et al[8] chose to use the cross-correlation coefficient method defined by expression (1) for matching purposes.

The numerical value of a correlation coefficient indicates the amount of correlation between the two items being compared. CC values range from +1, indicating perfect correlation, or identical patterns, to -1, occurring if one pattern is the 'negative' of the other. CC values near to zero imply little correlation. Experience of CC values found with Landsat MSS imagery is mentioned in section 4.5.

3.3 The correlation coefficient surface

For the purposes of this and succeeding sections, it is useful to consider the concept of a 'surface' of CC values. For any given gcp chip, it is possible to calculate CC centred on all column and row locations on the image. The values can be displayed in three-dimensional form, with the two horizontal axes being the column and row directions of the image and vertical direction being the value of CC. This will be referred to here as a CC surface.

In practice it is found, as might intuitively be expected, that this surface has the shape of an irregular hill standing out from a surrounding plain having mean value zero but with local noise leading to individual random values in the range of about −0.25 to +0.25. The peak of the hill corresponds with the position of maximum CC, and this is the best relocation place for that chip on that image.

In practice, the CC surface may show a number of other peaks, these being due to other ground features having a similar general appearance. For example, a gcp centred on a small body of water such as a reservoir, will correlate with other such bodies of water of approximately similar shape and size. It is possible therefore, to obtain apparently good relocations which are in fact incorrect.

3.4 Relocating a ground control point

In operation, the exact location of a chip in the new image is not of course known (otherwise there would be no need to find it), but, as will be shown in section 5, the approximate location will always be known, together with some measure of the probable maximum distance away from the actual location, say R pixels. These values - approximate column and row number, and largest expected distance from the actual location - are passed to subroutine LOCGCP as parameters and used by it in its search. LOCGCP is described in detail in section 7. It operates by starting at the supplied approximate location and conducting an outward rectilinear spiral search from there. The movements - upward, rightward, downward, leftward and so on - are due to the rectangular grid nature of the data. LOCGCP stops the search either when it has positively located the gcp, (as explained below), or when it has completed the specified number of 'turns' of the spiral.

Due to the method of operation of the main program, it is acceptable to have an occasional mislocation. Experience with many hundreds of gcps in various images of Britain shows that the following algorithm gives an acceptably high proportion (more than 90%) of correctly located gcps at the first attempt.

As the spiral search proceeds from its start, a correlation coefficient is calculated for each position. The value of the largest CC is stored (in fact the 10 largest values are stored in descending order, for later printout) and each time a higher CC value is calculated, the previous highest value is replaced. If at any time the maximum CC is greater than 0.5 and two complete rings of the spiral have been performed without a higher CC value being found, then the subroutine is brought to a halt, as consideration of the CC surface shows that it is likely that the correct location has been found, and is now being passed. In the absence of such a positive relocation, LOCGCP continues until all R rings have been searched. It then returns the highest CC value found, provided that it is greater than 0.3, together with its position. If no CC higher than 0.3 is met, an indication is provided that no relocation point can be found.

4 SUPERVISED RELOCATION OF GROUND CONTROL POINTS

Before the operation of the program GCP.FIND is described in detail it is of interest to consider the steps leading to its present design.

Work on the problem of gcp relocation started with an informal feasibility study. It soon became apparent that a chip size of about 20 pixels square could be correlated with an image in an acceptably short computer time, and that a spiral search was able to relocate gcps in a high proportion of cases. The operation was formed into a computer program, called GCP.LOCATE, (not described in this Report) whose operation was the basis for the present subroutine LOCGCP, though lacking some of the features now possessed by that subroutine.

This relocating program was then applied to images, in a user-supervised manner. The user started by making an estimate of the location of the first gcp within the new image, and then used the program to perform the exact relocation. Another nearby gcp was then chosen. Referring to the original image, the second gcp was at a known distance, in columns and rows, from the first gcp. These column and row differences could then be applied to the location of the first gcp in the new image, to give an approximate location for the second gcp in that image. GCP.LOCATE was then used to perform the exact relocation. This operation assumes that the new image is to the same scale, and is orientated at the same angle, as the first image, an assumption which, whilst seldom exactly true, is never far from the case. The differences of scale and orientation clearly introduce less absolute error for closely spaced gcps, hence the choice of near neighbours.

When several gcps have been located, one by one, with the aid of GCP.LOCATE, a transformation matrix was then calculated, using least-squares fitting. Inspection of the residuals - the distance of each gcp relocation point from the 'best' fit given by the least-squares calculation - showed whether any gcps had been badly mislocated, (assuming that most were correct). Such bad locations, if any, were then rejected and a new transformation matrix calculated. The resulting matrix allowed estimates to be made of the expected positions of other gcps, and these were then relocated, one at a time, using GCP.LOCATE. After several more gcps had been located, another transformation matrix was calculated, this being rather more exact than the first one, and the process repeated until all gcps within the new image had been examined.

At the end of these operations, most of the gcps had been correctly relocated, but a few had not been. Using the last-calculated transformation matrix, the expected positions of these few were recalculated one by one and new searches made for each. This sometimes resulted in satisfactory relocations being achieved, due to the better matrix, thus adding to the list of gcp locations.

A first-order matrix is adequate for determining the approximate location of a gcp and an existing program MATRIX[5] was used.

When several images had been processed by the method just described, it became apparent that the user was employing a 'human algorithm' and that with experience the 'decisions' involved were seen to be of a nature amenable to computation. It was therefore decided to combine the sequence of operations, as far as possible, into one all-embracing program. Experience showed that not only was this possible but that it worked well for most images examined. With further experience, refinements were added until the present program GCP.FIND had been evolved.

5 AUTOMATIC RELOCATION OF GROUND CONTROL POINTS BY PROGRAM GCP.FIND

Following the work described in section 4, the program GCP.FIND was evolved. When this program is run, it asks the user a few questions. Once these have been answered, the program then conducts its search, automatically relocating the majority of gcps and providing a list of their locations. This program will now be described in some detail, the subroutines being described in section 7.

5.1 User-supplied information

The information requested from the user includes:

- the filename of the new image,
- in some circumstances, some details about this image,
- the identity of the chosen first gcp,
- some indication of the location of that gcp in the new image.

The first three of these four items are of an undemanding nature and indeed the second item is no longer requested in the majority of cases. Only the fourth item requires some effort from the user, and the possibility of performing that automatically is discussed in section 5.7. Apart from item 4, the operation of the program can be considered to be entirely automatic.

The second item listed above concerns details about the image named in the first item. This image (which may in fact be a subimage of an entire Landsat MSS image) has associated with it certain information, such the the Landsat track and frame number, the date of the Landsat pass, and so forth. When this work started, these details were not provided with the image, and had to be supplied by the user at the terminal keyboard. It was therefore decided to attach a 'tail' to the end of each image file, containing such information about the image. This tail is now automatically provided whenever an image is constructed on disc storage from the computer-compatible magnetic tape, by the program IM.FUC.MSS[6]. If a subimage is formed from an entire image, using the program IM.SUB[9] the current version of that program ensures that a suitable tail is attached to the subimage, providing in addition to the details mentioned above, the correct top left-hand corner column and row numbers referred to the entire image. Since images are not usually retained for long periods of time on disc, in practice most images now have tails, so the second item is seldom requested.

The choice of the first gcp to be located is at the user's discretion and should preferably be one which stands out clearly from its surroundings. It is also preferable to choose a gcp well away from the edges of the image and as far as possible surrounded by other gcps, so that there are many close ones, since this reduces the searching time. In practice it is helpful to choose a second 'initial gcp' at the same time as the first one is chosen, so that if the prime one cannot be correctly located a second one is immediately available.

If only cloud-free images were to be examined, one gcp could be selected as the 'standard' first gcp for each track/frame in the Landsat system. However this program may successfully be used on images which have considerable cloud cover, provided that a proportion of the gcps are visible, and in such cases user discretion in the choice of gcp is needed.

The user is asked for some indication of the position of the first gcp, as the search is only conducted for 40 'rings', *ie* up to 40 pixels in all directions. The position may be provided in one of two modes, either as approximate column and row numbers in the new image or from measurements taken from a picture of the Landsat scene, the four dimensions required being:

x = horizontal distance of gcp from left hand side of picture,
X = horizontal width of picture,
y = vertical distance of gcp from top of picture,
Y = vertical height of picture.

Any unit of distance may be used, provided that the same unit is used for all four measurements. In the case of a standard RAE Linoscan[10] print, X and Y are each a little more than 180 mm. Provided that all of the four measurements are accurate to the nearest mm (*ie* their error does not exceed 0.5 mm) then the error in the implied location on the image should not exceed 18 columns or 13 rows, which is well within the search spiral of 40 rings.

5.2 Estimation of initial transformation matrix

When program GCP.LOCATE was used manually, the scale and orientation of the new image were obtained, by implication, by assuming them to be the same as for the original image. In the present program, the scale and orientation are calculated from knowledge of the Landsat system and the track and frame number. Thus, an initial transformation matrix can be calculated, to determine the expected positions of the second and subsequent gcps, once the first gcp has been located. It is recognised that this matrix is only an approximation, but it is satisfactory if it is good enough to enable several nearby gcps to be located.

To calculate the initial matrix, the following operations are done. The pixel size is assumed to be 57 by 79 metres, these being typical values. The orientation of the image with respect to the map coordinate system (THETA) is calculated for that image track and frame number, by subroutine IANGLE, discribed in section 7. The rotation of both x and y direction are taken to be of value THETA, which implies that the image is not sheared. The angle of shear depends on the yaw angle of the satellite, *ie* any deviation of the scan direction from perpendicularity with the direction of satellite travel, combined with earth rotation effects, and it has proved satisfactory to assume zero shear. Using these assumptions, four of the six matrix values can be calculated and placed into arrays B1 and B2. These four values determine the scale and orientation of the image in relation to the map coordinate system. The final two matrix values determine the relative displacement and are provided by the operation described in section 5.3.

5.3 Relocation of first ground control point

As the user-supplied information is provided, several starting operations are performed by the program, as follows.

Once the filename for the new image has been supplied, that file is opened. Failure to open correctly, *eg* a non-existent filename has been supplied, invokes a repeat request for a name. Once opened, the file is examined to see whether it has a tail. If it has, relevant data is extracted from it, if not the information is requested from the user - Landsat track and frame number and pass date.

After the required first chip identity has been supplied, in the form of map and serial numbers, subroutine CHPXAM is called, to construct the filename for the gcp chip. An invalid chip identity is recognised by CHPXAM, and a repeat request is made until a satisfactory identity is provided. The master map location file is then searched for this chip identity and, if found, the relevant map easting and northing are stored and the chip file opened. Failure to locate the chip in the master map file causes a request for another chip identity.

An attempt is then made to relocate the first gcp chip specified. If the approximate location has been provided in the form of four picture dimensions, (section 5.1) then the program calculates the approximate column and row numbers. This calculation assumes that the pictorial representation of the image is in the form normally available in Space Department, RAE, ie a 'quick look' or better a Linoscan print, in which cases the whole CCT image is used, resulting in a slightly skewed parallelogram. The amount of skew is approximately one pixel in twenty for most images of Britain, with system corrected data. If the image is not system corrected, or if the image is at a different earth latitude, then the amount of skew will probably differ. In such cases the calculation of the column number would need modification.

As an alternative to providing the four picture dimensions, the user may supply an approximate location as column and row values, and no conversion is needed.

Subroutine LOCGCP is then called, to request a search from the location provided, for up to 40 rings, an action indicator being set to 2.0 for a nearest-integer search. If this first gcp is not found by LOCGCP, indicated by the parameter RESULT returned less than zero, a message is displayed at the user terminal, and the program returns to an earlier stage, to request another gcp and location. The user may, if he so wishes, supply the same gcp but with a different starting location, if this seems desirable.

Once the first gcp has been relocated, its column and row numbers can be used, together with its map easting and northing, to calculate the final two values of the initial first-order transformation matrix, stored in arrays B1 and B2.

5.4 Relocation of other ground control points

After the first gcp has been successfully located and the initial transformation matrix prepared, the relocation of other gcps can commence. Since the transformation matrix is known to be only an approximation, the accuracy of prediction of the position of other gcps will become progressively worse as the distance of the gcp from the first one increases. The next operation, therefore, is a scan through the master map location file to list all gcps which seem likely, on the basis of the initial matrix, to be inside or near to the image (or subimage) in use. This 'batch' of gcps is tabulated in a set of arrays, which hold the map and serial numbers, eastings and northings and distance from the first gcp, other table values being set to zero for later use. When this operation is complete, the user is informed of the number of gcps in the batch, typically 80 to 100.

Each gcp in the batch is then examined, in ascending order of distance, in a similar manner to the first one. Its filename is constructed by subroutine CHPNAM, the approximate location is calculated, now using the easting and northing together with the initial transformation matrix, the chip file is opened and LOCGCP is called to relocate it. Since the estimated position of each gcp should now be known much more accurately than the first gcp, a much smaller spiral search is sufficient, the number of rings specified being dependent upon the distance of the gcp from the initial one. In practice this is found to be suitable, provided that the first gcp has been correctly located: if it has not, then there is little likelihood of any others being correctly found. When, as usually occurs, the gcp has been successfully relocated, LOCGCP provides details of the position and correlation coefficient, CC, and the main program informs the user that the chip has been found. It may be noted in passing that a considerable amount of information on the progress of the program is provided at the user's terminal: this may be ignored, or a general watch kept on the progress, or the details can be saved on a logging file for subsequent detailed inspection. A later version of GCP.FIND omits much of this detail, and condenses the remainder.

After 10 gcps have been relocated, with CC of 0.6 or more, the subroutine MATSUB is called. This subroutine calculates a new first-order transformation matrix, based on the actual positions of the 10 gcps and this should represent an improvement upon the initial transofrmation matrix, which was not based on actual gcps. Following this calculation, subsequent LOCGCP calls only ask for 12 rings to be searched, as the better transformation matrix now available increases the accuracy of the estimated positions.

All the remaining gcps in the batch are examined, with a 12-ring search, calculating a fresh transformation matrix after every ten more have been relocated. When all in the batch have been examined, another transformation matrix is calculated. The subroutine MATSUB is described in section 7.3, but it may be mentioned here that a limit to the accepted residual size may be sp[illegible] as a parameter to this subroutine, and thus the occasional mislocated gcp is not allowed to falsify the transformation matrix.

5.5 Rechecking of doubtful locations

To complete the initial relocation process, poorly located gcps are rechecked. The subroutine MATSUB, apart from calculating a transformation matrix, assigns to each gcp an 'error' which is the residual resulting from that calculation. Usually some of the gcps have residuals which are rather larger than might be expected, this limit being arbitrarily specified as 3 pixels. Such gcps are therefore rechecked, by a repeat of the relocation process. It is sometimes found that a gcp can be relocated at a position more consistent with the others, usually because the previous search for that gcp started too far away to find it, or perhaps because a lower peak in the CC surface had been met previously. The number of such better relocations is usually small, perhaps two per image, dependent on the quality of that image. An image with haze is amongst the most troublesome, leading to many mislocations which have apparently acceptably high CC values.

5.6 Completion of the program

The final stage of the gcp relocation process is to perform a search to the nearest 0.1 pixel (section 7.1.3). To do this, each relocated gcp is considered. If its residual is greater than 7 pixels, it is considered unsuitable and ignored. Otherwise the existing relocation position is used, and LOCGCP is called, this time with a limit of ten rings, since the position is now known quite accurately. The action indicator for LOCGCP is set to 1.0, to indicate that the search is to 0.1 pixel. In fact LOCGCP performs two spiral searches in this case: first a (maximum of) six ring search for the integer location, followed by a fractional search of up to ten rings. Occasionally the location is modified during the integer search, and this is usually for the better as judged by the resulting residual.

Finally the program outputs its results in a 'permanent' form, *ie* other than the transient output to the terminal. The final relocation positions, to the nearest 0.1 pixel, are output in a 'SID list', one position for each map identity. Additionally, a final transformation matrix is calculated and output as a file, but this is not often used as such, and has been omitted from subsequent versions of the program. In practice, users tend to prefer to accept the SID list and calculate a transformation matrix of either first or higher order under their own control using a program GCPFIT (not described here).

The program ends by closing all files. The user is then asked whether he wishes to use the program again.

5.7 Possible future developments

This section mentions possible future developments of the program GCP.FIND. Since some of these developments do not appear to be promising from an overall cost-effectiveness viewpoint, and use in other regions is not at present required, no work is at present taking place. Some of these ideas might be useful however, if the scope of the work were to be extended, to cover substantial land areas outside Britain, particularly places having little cloud cover.

5.7.1 Automatic relocation of the initial ground control point

Each Landsat image has its centre at a defined nominal location on the earth's surface, as described in section 7.2.1(ii). For reasons described in Ref 2 in practice the centre of an image can be some distance from the nominal position, by up to about 30 km. It would be possible to relocate the initial gcp by allowing it to perform a spiral search sufficiently large to cover 30 km in all directions from the expected position. This would require about 360 rings in the along-track direction and 480 in the cross-track direction. A search as wide as 480 rings is open to two objections, the more obvious one being the time required: this might be as long as 8 hours for the entire search, although often the gcp would be found before all of the rings had been examined. The other objection is based on relocation experience, which indicates that a number of possible relocations are likely to be found in such an extensive search, and it would often be difficult to determine by machine which is the 'correct' one – peak CC value alone not necessarily being sufficient.

A possible method would be to conduct preliminary searches at a different 'scale'. For example, an area 190 by 190 pixels in size could be reduced to a 19 by 19 chip by, for example, averaging subsets of 10 by 10 pixels, or by selecting every tenth pixel of every tenth row. The resulting 'superchip' could then be relocated by a spiral search, comparing it with similarly treated portions of the image, each turn of the spiral being ten pixels in width. It would be necessary to select the superchip so that it contained a distinctive pattern appropriate to its size. Once this preliminary relocation had been made, a normal relocation process could then be conducted to obtain the required accuracy. If necessary, a three or four stage operation could be conducted, using several types of superchip, with reduction factors of say 20 and 5, though it seems likely that one size only would suffice.

The system would therefore involve the use of one superchip, or possibly a series of two or three, for each Landsat scene. This system would seem well suited to regions of the world which have little cloud or haze. In Britain there is considerable cloud cover, and apart from completely unusable scenes, there are many where a useful amount of land is visible, but often not (say) the centre. It would therefore be necessary to have a number of superchips for each scene, and the user could select the most promising one. Even so, the larger superchips may not be relocatable, when the conventional sized ones can still be relocated in the spaces between clouds.

No experience has yet been gained with the use of superchips. It does seem possible however, that there may be difficulties in selecting suitably distinctive areas in some inland parts of Britain and in other countries, though the extensively convoluted coastline of Britain should provide many suitable shapes.

In summary therefore, this method should be suitable for some, but not necessarily all, parts of Britain and other countries.

5.7.2 Use of raw data for first and relocation

All imagery used for this work has been in the system-corrected form[6], which is known to include several types of alteration. For images of Britain the alterations include both a displacement of lines by one pixel at intervals of about 20 scan lines (the exact number varies), and the insertion of extra pixels at intervals which may be as close as one per 29 pixels in the centre of each scan line. Thus most chips, being 19 by 19 pixels in size, will include at least one of these types of alteration. These alterations are valuable for providing an overall rectification of the image, but are made at the expense of local discontinuities. It seems probable that both the initial location of gcps on maps and also the subsequent automatic relocation could be done with a higher level of correlation if such discontinuities did not exist.

There is now a considerable investment of effort in the existing set of chips extracted from system-corrected images, together with their column and row details, so it is not economic to change now to the use of raw data. However, if a new system were being set up, either for other regions of the world or for a different type of sensor, other than Landsat MSS, serious consideration should be given to the use of data containing as little rectification as possible. Since the nature of a gcp system is to allow the application of some form of transformation, using raw data would merely result in a slightly more complicated transformation over which one has complete control, rather than transforming material which has already suffered one process of rectification.

Recent information indicates that some system correction is now being done by a resampling technique, rather than by adding pixels. The above mentioned objection would be less valid in this case.

5.7.3 Modelling of satellite behaviour

A complete rectification system would take account of the several geometrical considerations normally used for rectification, and would also take account of the movements of the satellite, in particular its pitch, roll and yaw. Shlien[4] has shown that it is possible to deduce evidence about the satellite pitch, roll and yaw from the details of an image if sufficient gcps are selected. Conversely, a more accurate fit of gcps to image should be obtainable if information were available about the satellite movements. In fact this information is known, if not to a high level of accuracy, but the information is not transferred to CCT (at least those available from Earthnet at the time of writing).

To obtain the highest accuracy possible, it would therefore seem desirable to use raw (*ie* not system corrected) data, and to have available and use all known information on the satellite attitude and its changes.

5.7.4 Automatic decision of relocation quality

Whilst observing operation of the program GCP.FIND, the user finds himself making judgements about the quality of each relocation position. These judgements are based on such factors as the value of the correlation coefficient CC and how close relocation is to the expected place. For example, a CC value of 0.9 within one or two pixels of its expected location appears 'good', whereas a similarly high CC value of say 10 pixels away from its expected place excites some doubt. Normally it is better to suspend judgement until the whole image has been processed, at which stage the poorer locations are rechecked.

The author considers that it is better to restrict the operation of the program GCP.FIND to actually finding the locations, and leave the judgement of their quality to a separate program, (such as GCPFIT) applied to them. The only occasion during program operation when a judgement is needed is during the early stages when a rather higher value of CC (0.6) is insisted upon, to reduce the possibility of incorrect locations upsetting the first matrix recalculation.

5.7.5 Regions of the world other than Britain

GCP.FIND was designed originally for use with Landsat images of Britain. For example, the maps used are in the British National Grid, which is not applicable to other parts of the world.

The program could be made much more general in its application by using some more widespread map system, such as Universal Transverse Mercator (UTM). This would chiefly involve alterations to the subroutine IANGLE and, depending upon the map system used, might require additional data from the user, such as, for regions near the edge of a UTM zone, which zone is to be used. The map reference master file of gcps would of course have to be provided in the appropriate coordinate system.

A few other changes would be necessitated and these have been mentioned during the course of the program description.

6 OPERATING THE PROGRAM GCP.FIND

This section provides detailed instructions on how to operate the program GCP.FIND. It assumes some familiarity with the host computer system (Prime 750) and some experience of using computer programs in general.

6.1 Initial conditions

The program makes use of (a) a Landsat MSS image in the form of a Space Department image-format file, (b) a master map location file and (c) a set of gcp chips, so all of these must be available before operation can commence. Items (b) and (c) are normally present, in the user areas required by GCP.FIND, but it is the user's responsibility to provide item (a). Normally a Landsat band 7 image is used, because all of the chips have been prepared from band 7 images. Images in other bands have been tested, but, unsurprisingly, band 6 has shown less successful results than band 7, and bands 5 and 4 have shown poor results. Normally, an image is obtained in the form of a computer-compatible magnetic tape (CCT) containing all four bands, so selection of band 7 presents no problem. Image-format files may be obtained from a CCT by use of the program IM.FUC.MSS[6], in which case a tail is provided for each file. For purposes of comparison, it may be mentioned that the typical operating time to transfer an image by means of IM.FUC.MSS is over one hour, since it involves two passes of a 2400 ft tape. This figure assumes sole use of the system: operation may be slowed in a multi-user environment.

6.2 Operation

Program GCP.FIND has been observed to relocate all gcps in a new image in a typical sole-user elapsed time of about one quarter minute per gcp, *ie* 25 minutes for about 100 gcps. The program has been designed so that all user interaction takes place at the beginning, after which the process may be left unattended. It is recommended that information presented at the user terminal, much of which passes rapidly, should be logged, so that it can be inspected at leisure, if need arises. It may be convenient to run the program as a 'phantom', to free the terminal for other use, but if this is done it is desirable to ensure first that the initial locations, up to the first matrix recalculation, are satisfactorily achieved.

Fig 1 lists the first 45 lines of operation seen on the terminal during a typical run. This figure will now be considered line by line.

COMO LOGFIL This user request opens a log file named LOGFIL which logs all of the subsequent terminal output. When the run has been completed the user should key in COMO -END to close the current log file.

GCP.FIND This requests the program to be run. The program announces its name and version date. The current date and time are then output, together with two times which refer to the processor and input/output time usage.

IMLIB REV1 The initializing of the IMLIB system by use of subroutine INITI (section 7.5) causes this message to be output.

PROVIDE DETAILS OF THE SNAP:

TYPE INPUT FILE NAME: A 'snap' implies a unique Landsat track/frame/date. The reply should be the name of the image-format file containing the band 7 image which is to have relocations performed. If an invalid or non-existent filename is provided, a further request will be made. When the file has been opened, its 'text' is output, together with its size in columns and rows. In the present case the image file has a tail, otherwise the user would be asked to provide the track, frame and date.

INITIAL GROUND CONTROL POINT: XXX X The user responds with his selected initial gcp identity in the form of two integers: the map number and serial number. If the identity is illegal or does not exist, the user will be informed, otherwise the chip file will be opened, its 'text' written out and its size provided - this should always be 19 by 19, as here, with the present system.

PROVIDE POSITION OF GCP AND SIZE OF PICTURE

DATA (MM, REAL) IN FORM x X y Y. OR IF COL AND ROW The user may provide picture dimensions as described in section 5.1 or, as here, provide an approximate column and row number. In the latter case:

APPROXIMATE G.C.P. LOCCATION REFERRED TO ENTIRE IMAGE

COL: User provides approximate column number,

ROW: User provides approximate row number.

Another version of the program requests the name of the file for the output 'SID' list. For this version, a standard temporary filename is automatically provided by the machine.

The program may then be left to continue, proceeding in the manner described elsewhere, until completion. Fig 1 shows the spiral search for the first gcp and its successful completion. Finally:

ANOTHER IMAGE? to which the user replies YES or NO as he requires.

6.3 Error conditions

As already mentioned, operation may not be trouble-free. During the above description it was noted that invalid replies to questions often gave rise to error or warning messages. Other possible causes of error messages are now described.

6.3.1 Messages from the main program

The following messages can be provided by the main program. The causes and effects are explained.

FAILED TO OPEN MAPGCP.MASTER The master map location file was not successfully opened. In practice this should never happen. Causes could be (a) a system fault of some kind or (b) the master file is not present - perhaps it has been deleted. Since the program cannot proceed without the information contained in this file, it stops. The user is advised to consult the system manager.

THIS SNAP DOES NOT HAVE A "MAIN" TAIL This message will occur if an image is used which has been in existence since before tails were invented. In this case the user must answer some questions - track, frame and date - the answers to which are normally provided by information in the tail.

THIS GCP NOT IN MAPGCP FILE Occurs if the user provides a gcp identity (map and serial numbers) which does not exist. The user must provide another pair of values corresponding to an existing gcp.

GCP PROBLEM The program makes some checks on the size of each chip, when it is opened. All chips are 19 by 19 pixels, and should never fail this test. The message has never occurred in practice. If it did occur, the chip in question would be ignored and a new 'initial gcp' would be requested.

THIS GCP CANNOT BE LOCATED. TRY ANOTHER If the initial gcp is not relocated by LOCGCP within its 40-ring search, this message is output. The user should select another initial gcp or provide another estimated location for the present one, repeating the process if necessary until relocation is successfully achieved.

6.3.2 Messages from subroutine LOCGCP

NN IS TOO MANY RINGS FOR LOCGCP This would occur if LOCGCP were called with a number NN of rings in excess of 41. Should any of the declared parameter values, NSQ etc, be changed, the value 41 might be affected. This message is really an aid for the program writer, as it would expose certain program writing errors. Users should never see this message.

DENOMINATOR IS NOT POSITIVE This message, too, is a survival from the program writing stage, and the message should now never be seen by users. During the correlation coefficient calculation the square root of $R4 = R2 \times R3$ (section 3.2) is calculated. R4 must not be negative as otherwise the system routine SQRT will fail and R4 must not be zero, as otherwise the resulting CC would be infinite (the result of a 'divide by zero'. This message informs the user if either of these conditions should occur.

6.3.3 Message from subroutine MATSUB

MATSUB: ONLY N SUITABLE GCPS FOUND MATSUB cannot perform a meaningful calculation if less than four gcps have been relocated, so the program is brought to a stop. This occurrence is related to the nature of the image or the position of the initial gcp within it. For example, if for some reason the initial gcp has been mislocated, the subsequent gcps will probably be either mislocated also, or not relocated at all. Thus there will be insufficient apparently good gcps for MATSUB. Again, if there are few clear gcps on the image, due to the presence of clouds, only a small number may be found in the batch of those within 50 km of the first. In most cases the user can deduce the nature of the problem from inspection of the log file of the program run and examination of a picture of the image.

6.3.4 Message from subroutine IANGLE

IANGLE: PATH M OR ROW N OUT OF RANGE Subroutine IANGLE checks the validity of the Landsat path (track) and row (frame) within the limits of the whole system, 1-251 and 1-119 respectively. Since IANGLE is designed for use in Britain only, it would perhaps be more apposite if these bounds were restricted to allow only British scenes to be dealt with. This message should never occur when using an image having a tail, as the track and frame numbers should always be valid. In the case of a tail-less image the user is asked to supply these details and could provide invalid values, in which case the message would be output and the program would stop.

6.3.5 Other messages

Most other messages provide information on the operation of the program rather than about errors. A complete run of GCP.FIND produces a log file which may be many hundreds of lines long and provides considerable detail concerning the progress of the search. It is hoped that most of these messages are self explanatory. As mentioned above, later versions of the program provide less output information.

7 SUBROUTINES USED BY PROGRAM GCP.FIND

7.1 Subroutine LOCGCP

The general method by which subroutine LOCGCP determines the location of a gcp chip in a new image has been described. This subsection describes the subroutine itself in some detail.

Subroutine LOCGCP has four parameters: the column and row numbers at which the search is to start, the number of rings, R, to search and an action indicator. After operation, it returns the relocation column and row numbers, if found, and the result of the search, which may be in the form of a correlation coefficient value, but if less than zero in value, is interpreted as indicating the manner in which the search failed. Thus, if the best CC value found is less than 0.3, this is not considered to be a relocation, and the indicator is returned as -1, to indicate this fact. A value of -2 is returned if the subroutine is asked to search starting from a location which is outside the image, or insufficiently within it: any position less than 9 pixels in from any edge of the image will not allow a correlation between chip and image to be calculated. It is the responsibility of the calling program to examine the result returned by LOCGCP, and to take appropriate action.

The next operation of LOCGCP is to read the chip pixel values into an array IPATCH, of size 19 by 19. As each line of the image is read in, the opportunity is taken of performing the calculation of R2 (referred to in section 3.2) for use subsequently in the CC calculation.

7.1.1 The search workspace

A Landsat MSS image is normally over 3000 by 2000 pixels in size, which is far too large to hold in the fast memory of the host computer (a Prime 750 machine), so the data is held available in slow memory. It is therefore advantageous to transfer a portion of the image into fast memory and then conduct the spiral search, rather than to perform the search by repeated access of data from slow memory. An early operation of LOCGCP, therefore, is to transfer a suitable amount of the image into the workspace, IWORK, in fast memory. The size of IWORK has been chosen as 101 pixels square, *ie* a little over 10000 words, which is a reasonable size for fast memory. This square array is centred on the starting location and thus allows 41 rings of the spiral to be searched. If an R value greater than this is requested, an error message is output. If an R value less than 41 is specified, only the required amount of image is brought into the workspace, minimising the data transfer time. Thus, if the location is known fairly accurately, the whole process is speeded up, both because of the reduced amount of data to transfer (a slow input process) and the reduced computation time involved.

If the specified search location is close to the edge of the image (though not of course nearer than 9 pixels) there may not be a complete square of image around that location. This situation can be handled: LOCGCP reads in as much image as is available into the workspace and then conducts the search. If at any time during the search the chip extends beyond the available image, then no CC calculation is made. This arrangement ensures that the maximum amount of searching which is mathematically meaningful is in fact done.

7.1.2 The spiral search

With both chip and search area of image available in fast memory, the outwardly spiralling search can commence. The search starts at the location specified by the input parameters, and then moves in steps: one step up, one step right, two separate steps down, two separate steps left, three steps up, and so on. At each step, the correlation coefficient is calculated as described in section 3.2. CC is thus calculated at each position along the spiral. After each calculation it is examined to see if it is any larger than any previous CC value so far met during this search, and if so, the new CC is stored, together with the column and row numbers. Up to nine previous values in those arrays are retained, and provided CC is greater than 0.5, the number of the current ring is recorded, at all times retaining the number of the ring containing the highest value of CC so far found.

Section 3.3 described the CC surface, which in practice consists of a hill, whose peak is the position of largest CC value, rising from a rough plain. The spiral search, starting from some place not too distant from the hill, can then be imagined. As the spiral search proceeds, each turn rises higher up the flank of the hill, until eventually the peak is passed. Thus if a reasonably high CC value is met and then no higher value is met on the next two rings, it is a reasonable assumption that the peak has been passed. If however, no CC value as high as say 0.5 has been met, it is uncertain that the peak has been found, and the search is allowed to continue until all the rings have been completed.

As described, the model is open to objections - it is possible to postulate hills with double peaks, or ones which are surrounded by a high plain level. It is the essence of a suitable gcp that it does not have such behaviour. Ideally even at different times of year or with different sun elevation etc, the CC surface should show a single, fairly steep and tall hill rising from a low plain. Gcps which do not show such desirable behaviour should, if possible be removed from the list, as the fact becomes evident.

In practice it is sometimes found that a gcp is correctly relocated with a peak CC value less than 0.5 and in such cases some computation time will have been wasted by continuing the search unnecessarily. However it is not possible to use the value of CC as an indication of true relocation, as experience shows that values from about 0.3 upwards can correspond to correct location, and peaks of up to 0.6 or more are sometimes incorrect, due to a chance correspondence.

Experience has shown that the search rules described above work very successfully, providing that a little care has been taken in the choice of gcps. However, gcps which are mislocated by LOCGCP tend to be those difficult to locate manually on first identification, so in practice there is a natural bias towards 'good' gcps. The major source of incorrect locations is when, for some reason such as a change of ground use or the gcp has been veiled by a light haze, no location is possible. In such cases LOCGCP may detect a position where CC is for chance reasons greater than 0.3, and this random location is returned and accepted. Such mislocations can usually be detected and eliminated by the main program later on.

When the search has concluded, some results are displayed on the user's terminal. These include the location itself, as column and row, together with the peak CC value, but include in addition a table of CC values for the 5 × 5 square of image locations surrounding that peak, and the (up to) ten next highest CC values found during the search. These results enable the user to visualize the CC surface and the approach of the spiral to the correct location. Much of this information has been omitted from a recent version of the program, as some users are not interested in it.

7.1.3 Location to fractions of a pixel

Inspection of CC surfaces as described above showed that often a value adjacent to the peak was only slightly less high, indicating that the 'true' location was probably between the two integer locations. This suggested the possibility that a more accurate value could be determined. The method chosen was to start from the integer location of maximum CC and to spiral outwards in rings of 0.1 pixel rather than the usual 1.0 pixel. To perform the calculations of CC, a grid of pixel values at intervals of 0.1 column and row are needed, and these are obtained by linear interpolation from the adjacent two, or more usually, four, pixel values. This method enables the 'true' location to be obtained to the nearest 0.1 pixel. Linear interpolation is chosen as probably the quickest method, and although clearly not mathematically the most accurate is probably more than sufficient for the purpose, considering the nature of the overall system. 0.1 pixel corresponds to about 6 m in the scan direction and 8 m in the travel direction.

LOCGCP is instructed whether to search to an integer or a fractional location by means of an action indicator parameter. A value of 2.0 indicates an integer search whereas a value of 1.0 (actually any value not within the range of about 1.5 to 2.5) indicates that a fractional search is required. The calculation of values to 0.1 pixel spacing is in principle similar to that done for integer pixels. The creation of the spiral is done by a series of steps, as before, and, as for the integer case, is stopped when the peak CC value has been passed.

7.2 Subroutine IANGLE

Since Landsat satellite orbits are neither polar nor equatorial, the path of the satellite, or rather its ground nadir, will make some non-zero angle with true north. The transformations described in this Report are made to British National Grid, the northings of which are only true north at one longitude (2 degrees west). For any image, the centre of which is specified by a path and row number, there will be a difference angle between satellite path and grid northing. It is the purpose of subroutine IANGLE to calculate this difference angle for any specified path and row number.

Referring to the method of operation of the overall program, once the first gcp has been located attempts are made to locate other gcps at some distance from the first. The approximate locations of these are calculated from (amongst other things) knowledge of the difference angle between the satellite path, (*ie* the y direction of the image) and the map northing. The search for gcps is limited by subroutine LOCGCP to 41 rings, *ie* up to 41 pixels in all directions. Since the location may be up to say 100 km distant, which is a maximum of 100 × 16 = 1600 pixels away, it is necessary that the difference angle be known to an accuracy of better than 41 in 1600 or about 1.4 degrees. It is therefore important that IANGLE provides a result which is less than 1.4 degrees in error, and to do this for the range of path and row numbers applicable to all images of Britain.

As noted, the difference angle consists of two elements, one being the angle between the satellite path and true north, and the other the difference between true north and Grid north. Both of these angles vary with latitude and in the case of the second with longitude also. Since the image location is provided as a path and row number, it is necessary first to convert these to longitude and latitude.

The operation of IANGLE is therefore the following sequence:

(i) The latitude and longitude of the centre of the image are calculated from the Landsat path and row number,

(ii) the satellite heading, expressed as the angle between its path and true north is calculated for that latitude,

(iii) the difference angle between true and National Grid north is calculated for that latitude and longitude (this angle is referred to as the convergence),

(iv) the rotation angle is calculated by addition of (ii) and (iii).

Items (i) to (iii) will now be considered in more detail.

7.2.1 Calculation of latitude and longitude from path and row

The Landsat 'Worldwide Reference System' of paths and rows is described in Ref 2. Relevant details include:

(a) There are, in total, 251 evenly spaced paths. Although these are not traversed in sequence by the satellite, nevertheless it is convenient to number them in a sequential manner. The separation of paths is therefore 360/251 = 1.4343 degrees, each path being identical to its neighbour but displaced in longitude by this amount westward from its predecessor. The location of the paths is not defined exactly in Ref 2 in terms of longitude, although it is noted that path 1 is "the first track that includes mainland North America". However, examples of the relationship between path/row and longitude/latitude are provided and from these it is possible to calculate the longitude at which any path crosses the equator. In particular, path 251 (which is the same as path 'zero') is found to cross the equator at 64.05 degrees west. (This and all other longitudes in this Report are referred to the Greenwich meridian.)

(b) Each complete orbit takes 103 minutes (more exactly 6196 seconds) and images are centred on locations 25 seconds apart. Thus there would be 248 complete images in one complete orbit (more exactly 247.84) although only 119 of these row locations are used. The 119 are recorded on the north-south portion of the orbit which is the sunlit side of the earth, and path number 60 is at the equator. Row 1 is the most northerly row used, being at 80 degrees 1.2 minutes north, and row 119 the most southerly, at the same latitude south. Row 59 is centred on a location 25 seconds before the equator is reached, and so on.

From this information it is possible to calculate geographical locations from path and row numbers.

(i) Latitude

If the orbit of the satellite is assumed to be a circle, it may be divided into 248 equal sections, corresponding to adjacent row numbers. Since row 60 is at the equator, any row ROW is located on the orbit at a fraction (60-ROW)/62 of a quadrant from the equator, *ie*

$$\text{latitude of ROW on the orbit} = \text{pi} \times (60-\text{ROW})/124 \text{ radians.}$$

This is the latitude on the orbit, which is at an inclination angle A to the equator. When the orbit is related to the earth's surface the earth latitude Lat is given by the expression

$$\sin(\text{Lat}) = \sin(A) \, . \, \sin(\text{pi} \times (60-\text{ROW})/124)$$

this calculation being for a sphere rather than a more complex shape.

For an ellipsoid

$$\tan(\text{Latc}) = e \times e \times \tan(\text{Latd})$$

where Latc is the geocentric latitude (Lat above)

Latd is the geodetic latitude and

e is the ratio of the minor to the major axis of the ellipsoid, a value of 0.997 being used.

The latitude thus depends upon the row number only. The longitude depends upon both path and row number and is considered next.

(ii) Longitude

As mentioned above, path zero has been found to cross the equator at 64.05 degrees west.

Each successive path is displaced, in longitude only, by 360/251 degrees westward. For any given path therefore, the path number may be multiplied by 360/251 to provide the westward longitude shift for that path, relative to path zero.

Considering any one path, the longitude at row number ROW is an angle Lon further east than at row number 60 (the equator) where

$$\tan(\mathrm{Lon}) = \cos(A) \,.\, \tan\left(\mathrm{pi} \times (60-\mathrm{ROW})/124\right)$$

and this expression is used to evaluate the relative longitude for any given row number.

These longitude calculations have assumed that the earth is stationary, whereas in fact it is rotating in relation to the sun, and hence the satellite's sun-synchronous orbit, at the rate of one revolution in 24 hours. For each increase in row number an interval of 25 seconds has elapsed during which time the earth has rotated by 360 × 25/(24 × 3600) degrees, *ie* 0.10416 degree per row. It is therefore necessary to adjust the calculated longitude by this amount for each row number difference from the equatorial row, number 60.

These various steps necessary for the calculation of the longitude occur as separate statements in the computer listing of the subroutine.

Calculations of longitude and latitude made by these means are shown below, in degrees:minutes, for the several path/row examples given in Ref 2.

Path	Row	Long/Lat from Ref 2				Long/Lat from IANGLE			
–	1	–		80:01	N	–		80:01	N
13	35	73:28	W	35:58	N	73:29	W	35:57	N
47	35	122:14	W	35:58	N	122:15	W	35:57	N
105	65	143:41	E	7:13	S	143:41	E	7:13	S
146	101	67:14	E	58:31	S	67:17	E	58:30	S

There is seen to be fair agreement between the values. The residual differences may be due to such causes as using a different value for e (several slightly different values are in use), a different value for the satellite orbit inclination, or to use of a simplified (spherical) earth model for some of the expressions.

The image supplied for any particular path and row is usually displaced from its nominal position (see Ref 2), often by a larger distance than the above errors in the calculated locations. Examination of fifteen images shows that the centre may be displaced from the nominal position by up to 0.47 degree in longitude and 0.21 degree in latitude, these figures including the above calculation errors of about 0.05 degree and 0.02 degree respectively. Thus the calculated longitude and latitude are adequately accurate for their purpose. The effect of the error in location of the centre of the image is considered later. (If account is taken of the satellite in use, it is found that the longitude displacements for Landsat 1 are in the range 0.337 to 0.464 degree, and for Landsat 2, −0.179 to 0.230 degree. It thus seems that Landsat 2 occupied an orbit roughly 0.4 degree eastward of Landsat 1 during the relevant time period.)

7.2.2 Determination of the satellite heading

The satellite's heading, the angle between its path and geodetic north at a given latitude B is determined with the aid of the expression

$$\cos A = \cos B \cdot \cos C$$

where A is the inclination of the satellite orbit to the equator,
B is the latitude,
C is the inclination of the satellite orbit to that latitude parallel.

The inclinations of the orbits of the Landsat series are provided in Ref 2 and measurements of the orbits are recorded in Ref 11. Since the Landsat missions are intended to provide a consistent operational system, the orbit inclinations do not vary greatly and a figure of 99.1 degrees may be used for all of the satellites. Since the direction of satellite travel is not relevant here, the angle in the first quadrant, 80.9 degrees, may be used for convenience. Due to the consistency of all Landsat orbits this angle will in general be in error by appreciably less than 0.1 degree and consequent errors in C should be less than 0.1 degree for the latitude of Britain.

After application of the above equation at the appropriate latitude, the angle C is converted to an inclination with respect to the local meridian rather than the local parallel.

7.2.3 Difference between true north and Grid north, or convergence

The British National Grid reference system used for this work is an orthogonal system superimposed on a transverse Mercator projection. The central meridian of the Grid (2 degrees west) is the only northing line which is true north, all other northings having some convergence from true north. This angle may, according to Ref 12, be approximated by the expression

$$\text{convergence} = (\text{Lon} + 2) \cdot \sin(\text{Lat})$$

where Lon and Lat are the longitude and latitude in degrees. This expression was confirmed by comparison with a table of values calculated for this purpose and is in error by less than 0.004 degree for all areas which are normally mapped on the British National Grid.

Further increase in accuracy is not justified, as the present expression contributes negligible error to the subroutine.

7.2.4 Accuracy of subroutine IANGLE

The overall accuracy of operation of subroutine IANGLE is determined by the accuracies of its several constituent parts. These have been discussed in the appropriate subsections and can now be combined.

Considering first the longitude and latitude, a study of fifteen images detailed in Table 1 shows that the centre of an image may be as much as 0.47 degree and 0.21 degree away from its calculated position. Such differences may be shown to give rise to errors of up to 0.35 degree and 0.13 degree in the resulting difference angle. (These figures are only approximate as they vary with latitude.)

In addition there are possible errors in satellite heading of up to 0.1 degree and in convergence of 0.004 degree.

Thus when all errors are taken into account it appears that the subroutine IANGLE should be able to provide angles which are in error by less than about 0.5 degree.

A comparison has been made between the rotation angles provided by IANGLE and the actual rotation angles determined from the relevant transformation matrices. Fifteen images of widely distributed portions of the UK were examined. Each image provides two rotation angles, one of which, the rotation of the y-axis of the image, is due to the causes described in this section. The other angle, the rotation of the x-axis of the image, contains additional small effects due to any yaw angle of the satellite.

Table 2

Differences between calculated and actual values for images (degrees)

Path	Row	Centre displacement Longitude	Latitude	y-angle	Calculated	Difference
216	23	0.472	0.013	11.748	11.891	0.143
217	24	0.061	0.209	13.372	13.150	-0.222
218	23	0.441	-0.013	13.965	14.180	0.215
218	24	0.403	0.061	14.236	14.273	0.037
219	24	0.161	-0.042	15.286	15.396	0.110
219	25	0.140	0.035	15.194	15.455	0.261
220	22	0.239	0.016	16.160	16.406	0.246
220	23	0.206	0.015	16.297	16.469	0.172
220	24	0.339	0.058	16.461	16.519	0.058
220	25	0.337	0.048	16.843	16.555	-0.288
221	18	-0.062	-0.080	17.015	17.273	0.258
222	19	0.164	-0.146	18.147	18.589	0.442
222	20	0.464	0.059	18.337	18.652	0.315
224	19	0.186	-0.108	20.901	21.035	0.134
224	20	-0.179	-0.018	20.806	21.060	0.254

Thus when the angles calculated by subroutine IANGLE were compared with the rotation required by the image y-axis the average difference found was 0.21 degree and the largest difference 0.44 degree. These figures are consistent with the expected accuracy of the subroutine.

Summarising, the overall accuracy of subroutine IANGLE appears to be consistent with the calculated accuracy of about 0.5 degree, adequate for its intended use, which demands a limit of 1.4 degrees.

It should be emphasised that the use of subroutine IANGLE is only appropriate to the British National Grid system and for regions normally mapped to that Grid. This does not include, for example, any part of Ireland.

The mathematical expressions used are in general not rigorous (for example some earth models are based on a sphere rather than an ellipsoid), but are seen to be adequately accurate representations.

7.3 Subroutine MATSUB

Subroutine MATSUB has the purpose of calculating a first-order transformation matrix from a set of pairs of locations in the two reference frames: image and National Grid coordinate systems. This subroutine is in fact adapted from an existing program MATRIX[5] so it will not be described in detail here. However it now contains some modifications, as follows.

MATSUB has been altered so that it reads the corresponding sets of image locations and map coordinates from appropriate arrays via a COMMON block, rather than by obtaining them from a data file. In addition, it uses a small amount of discrimination. If any gcp has a residual (determined by a previous MATSUB calculation) of greater than an amount specified by one of the subroutine parameters, or has a correlation coefficient of less than zero, that gcp is not included in the current calculation. Initially all residuals are set to zero by the main program, so the first MATSUB operation uses all available gcps. A correlation coefficient of less than zero implies that the gcp has, for one reason or another, not been relocated: in fact any CC value less than 0.3 will be provided to MATSUB as less than zero.

If less than four suitable gcps are provided, MATSUB is not able to function, so it provides an appropriate error message, and then stops. This might occur if the new image were largely obscured by cloud, so that few gcps could be relocated.

When MATSUB has performed the least-squares fitting, it then prepares a new transformation matrix which is passed to the main program through the COMMON block MATRIX.

During the course of its operation, MATSUB provides some output to the user terminal, concerning both the details of the individual gcps, their locations and residuals, and also details of the new transformation matrix, including the implied pixel sizes and the relative orientation both along and perpendicular to the satellite direction of travel. This data output has been omitted from a recent version of the program.

MATSUB therefore is a fairly unsophisticated subroutine, providing only a first-order transformation matrix. It is, however, quite adequate for use in GCP.FIND, and more complication is not justified. Once a set of gcps has been relocated using GCP.FIND, the user may, if he wishes, use a program GCPFIT to obtain a first- or higher-order transformation matrix, selecting such gcp locations as he requires.

7.4 Library and system subroutines

7.4.1 Subroutines in library HIMLIB

Program GCP.FIND makes use of several subroutines held in a library named HIMLIB. The HIMLIB routines used are listed here, with a brief description of their action.

CHPNAM: forms a chip filename from a gcp identity.

CLSTMP: closes a temporary file.

OIRX: opens an image-format file for reading, using the filename as a parameter.

OTMPRW: opens a temporary file for reading or writing.

RDTAIL: reads the tail (if any) of an image-format file.

TIMES: outputs to the specified channel (1 = terminal) details of the date and certain times. This allows estimates to be made of the usage of computer processing and input/output time.

7.4.2 Subroutines in library IMLIB

Use is made of some subroutines held in the image-format library IMLIB. These are:

CLOSEI: closes an image-format file.

INITI: initializes data in a COMMON block used by IMLIB routines.

OIR: opens an image-format file for reading.

7.4.3 System subroutines

The program GCP.FIND has been developed for use on a Prime 750 computer and is written in Prime FORTRAN IV. It makes use of a number of Prime system subroutines, most of which are of a familiar nature, such as READ or WRITE. Less familiar routines include:

CLOS$A: closes a file.

RWND$A: 'rewinds' a file, *ie* returns to the start.

YSNO$A: presents a question at the user terminal and sets the logical value TRUE or FALSE according to the response YES or NO.

It is necessary to include the insert SYSCOM>A$KEYS in the declaration section of all programs and subroutines which use such system subroutines, to 'explain' to the compiler the value of words, used as parameters in these subroutines, which include the "$" character.

8 RESULTS AND CONCLUSION

The 'product' of GCP.FIND is a list of all the gcps which have been relocated within a given image. Ideally, all gcps which lie within the image will be relocated, and to a high accuracy. In practice, due to small mislocations of the original gcps, an accuracy of about ±0.4 pixel is to be expected, rather than the 0.1 pixel apparently possible.

The alternative to using GCP.FIND is to locate as many as possible gcps manually, which is a time-consuming labour-intensive process. The success or otherwise of GCP.FIND is therefore to be measured by where its performance lies, whether 'perfect', or 'better' or 'worse' than the manual method.

To provide a qualitative answer to this question, it would be necessary to record systematically the results of use of GCP.FIND on a wide range of images, and no such systematic recording has been done. However the program has been used on many different images (over 100) by the author and several other users, and the following comments represent the experience found.

GCP.FIND is very much more convenient to use than the manual method and no user would now contemplate manual relocation, except possibly for an image which was so cloud-covered that only a very small number of widely scattered gcps could be discerned. The program can certainly be said to be very much better than manual methods.

The proportion of gcps relocated within an image varies, depending on such things as the season of the year compared with when the gcp chips were obtained. For similar seasons, the proportion relocated approaches 100%, whilst in the worst cases attempted about 60% of the gcps were found, when the image was about six months different in season from the original chips, and some snow present in some portions of the scene.

The accuracy of relocation may be judged from the standard deviations of the gcps. In the best cases the standard deviation may be less than 50 m (about 0.6 to 0.8 pixel) with one or two out of perhaps 150 being bad 'outliers'. In the worst situation, the standard deviation may be 120 m with up to 15-20% badly relocated, mainly because they cannot be located at all.

It should be pointed out that heavy cloud cover, or sometimes in Britain poor atmospheric clarity, reduce the value of images, so these results refer to images which appear usable after a very brief visual inspection.

In summary, GCP.FIND provides a rapid method of relocating gcps, with little user effort, and enables large quantities of imagery to be processed, which would not have been contemplated with manual methods. Provided that the image quality is good, the process has a high success rate in accurately relocating gcps.

REFERENCES

No.	Author	Title, etc
1	R.N. Colwell	History and future of remote sensing technology and education. Conference of Remote Sensing Educators (CORSE-78) (1978) NASA Conference Publication 2102.
2	US Geological Survey	Landsat Data Users' Handbook (1979)
3	A.P. Colvocoresses	Space Oblique Mercator. Photogrammetric Engineering, Vol 40, 921-926 (1974)
4	S. Shlien	Geometric correction, registration, and resampling of Landsat imagery. Canadian J Remote Sensing, Vol 5, No. 1, p74 (1979)
5	J.M. Williams	Geometric correction of satellite imagery. RAE Technical Report 79121 (1979)
6	A.H. Benny	A study of Landsat MSS data tapes. RAE Technical Report 80018 (1980)
7	D.I. Barnea H.F. Silverman	A class of algorithms for fast digital image registration. IEEE Transactions on Computers, Vol C-21, p179 (1972)
8	F. Orti A. Garcia M.A. Martin	Geometric correction of Landsat MSS images using a ground control point library. Proceedings of the fifth annual conference of the Remote Sensing Society, held at Durham, pp17-26 (1979)
9	G.J. Davison	Image Processing Software, Issue 1. RAE Technical Memorandum Space 280 (1980)
10	P.J. Neilson	LINOSCAN 204D image generator in remote sensing applications. The Photogrammetric Record, IX, p849 (1979)
11	J.A. Pilkington D.G. King-Hele H. Hiller D.M.C. Walker	Table of Earth Satellites, Vol 3, 1974-1978. RAE Technical Report 80001 (1980)
12	Ordnance Survey	Some facts and figures relating to the Transverse Mercator Projection and the National Grid. Leaflet No. 37 (1972)

```
COMO LOGFIL
OK, GCP.FIND
GCP FIND           15-SEP-80


 THU, 20 SEP 1980       13:44:10          04.61           29.91


IMLIB REV1
PROVIDE DETAILS OF THE SNAP:
TYPE INPUT FILE NAME: LOTIDE7
LOWTIDE IN BRISTOL CHANNEL BAND7
  3159 COLS,  2206 ROWS.
INITIAL GROUND CONTROL POINT:  XXX X
172 3
DENNY ISLAND IN CHEW VALLEY RESR
    19 COLS,    19 ROWS
PROVIDE POSITION OF GCP AND SIZE OF PICTURE
DATA (NN, REAL) IN FORM x X y Y.   0s IF COL AND ROW
0 0 0 0
APPROXIMATE G.C.P. LOCATION REFERRED TO ENTIRE SCENE
COL: 550
ROW: 1660
RING:  0
RING:  1
RING:  2
RING:  3
RING:  4
RING:  5
RING:  6
RING:  7
RING:  8

BEST INTEGER LOCATION IS COL  552, ROW 1654
COEFFICIENT IS  0.8541

  0.7302  0.7245  0.7036  0.6506  0.5037
  0.7812  0.8110  0.8020  0.7549  0.6735
  0.7715  0.8400  0.8541  0.8191  0.7477
  0.6759  0.7642  0.7944  0.7776  0.7313
  0.5467  0.6322  0.6732  0.6602  0.6437

  0.8541  0.8400  0.7944  0.7642  0.6759  0.6732  0.6640  0.6609  0.6503  0.6412
     552     551     552     551     550     552     547     548     549     543
    1654    1654    1655    1655    1655    1656    1660    1660    1660    1661
```

Fig 1   First 46 lines of output of a typical run of program GCP.FIND