Remote Sensing Image Processing.
Cut-up Algorithms for Image Analyses

Florin Rotaru, Mariana Roman, Silviu Bejinariu and Simona Pescaru

Research Institute for Information Science of the Romanian Academy
22A Copou street
6600 Iasi
ROMANIA
e-mails:florinr@iit.tuiasi.ro; marianar@iit.tuiasi.ro; silviub@iit.tuiasi.ro; simona@iit.tuiasi.ro

Abstract: In order to obtain, through remote sensing, meaningful information for agriculture, on the raster image of the zone to be analysed it has to be overlapped its digital model. The digital model represents the agricultural plot borders as a network of polygons. Each plot image has to be isolated with a view at being processed according to the direct terrain measures and extracting reliable information. The paper describes a remote sensing image processing application, that creates a database of extracted plot images and related information. Specially developed Cut-up algorithms are highlighted.

Florin Rotaru is executive manager of the Research Institute for Information Science. Born in Bucharest, in 1953, he obtained MSc. degree in Automatic Control from the Polytechnical Institute of Bucharest in 1977. He is a Senior Research Scientist in Image Processing and Pattern Recognition. His research interests include structural and decisional pattern recognition methods, applications of document image analysis, metallographic image processing and remote sensing image analysis.

Mariana Roman was born in Iasi, Romania, in 1962. She obtained MSc. degree in Computer Science from Technical University of Iasi in 1985. She is a researcher at the Research Institute for Information Science, Iasi. Her main research interests are image processing, pattern recognition and parallel computing.

Silviu Bejinariu is a senior researcher at the Research Institute for Information Science Iasi. Born at Botosani, in 1961, he took the MSc. degree in Information Science from the "A.I. Cuza" University, Iasi in 1984. His research interests include image processing, syntactic pattern recognition methods, applications of remote sensing image analysis.

Simona Pescaru was born in Iasi, Romania in 1962. She holds a MSc. degree in Computer Science from Technical University of Iasi, 1985. She is a researcher at the Research Institute of Information Science Iasi. Her research interests cover the areas of image processing and fuzzy classification.

1. Introduction

An analysis of satellite images, in order to evaluate the crop type and to assess conditions , involves complex operations that can be structured into six steps:

- Construct a numeric model of the test region by digitising maps of interest zones corresponding to those from radar images;
- Establish the control plots;
- Geometrically correct the satellite image and correlate the remote sensing data with the digital model of test regions;
- Cut-up, from the resulted data, the images of plots of which analysis we are interested in;
- Process the cut-up images – determination and extraction of the classification features;
- Interpret the processing results through referencing the soil information (crop type, soil moisture content, slope, etc).

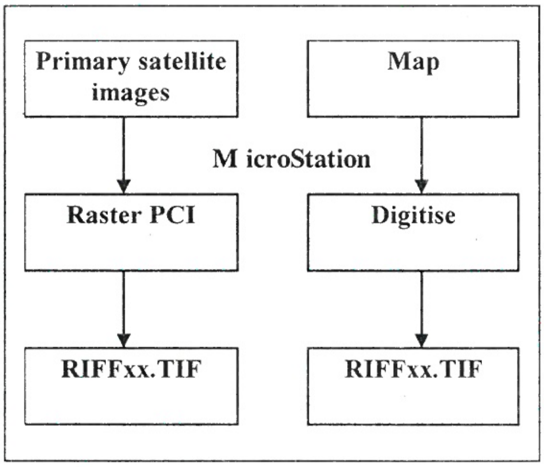
This paper presents an application that creates a database of control plot images, as a result of cut-up, from the satellite image of the area to be inventoried, of the interior of polygons from the digitised plots map. To obtain the necessary information, the plot images have to be processed and the results have to be analysed .

The data obtained for the cut-up, processing and interpretation stages are stored in different file types:

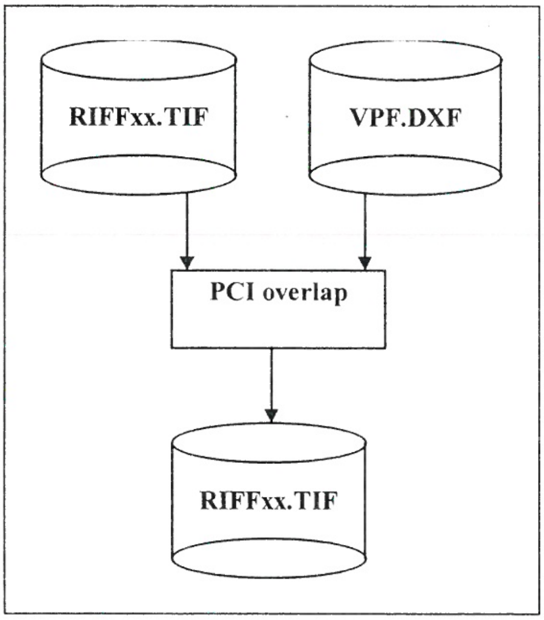
- VPF.DXF (Vector Polygon File) - in DXF format- contains, among other elements, the control plots contours. The file results from the digitisation of inventoried region maps;
- RIF_{xx}.TIF (Raster Image File) - in TIF format, satellite image file;
- $RIRF_{xx}$.TIF – includes the satellite image correlated with the digital model of the terrain (VPF.DXF).
- CIF_{xx}.BIN (Card Image File) – includes the image of one control plot;
- PPPF.DBF (Processed Pixel Polygon File) – includes the results from processing the control plot images ;

- TAPF$_{xx}$.DBF (Thematic Attributes Polygon File) - includes specific data for each beneficiary (forestry, agricultural…). It contains information resulted from the analysis, by an expert system, of the data from PPPF.DBF correlated with terrain information.

A standard GIS software such as PCI, MicroStation, etc. will supply the processed data in RIFxx.TIF and VPF.DXF files. These files store the results of the primary satellite images processing or of the digitisation of maps of the area to be inventoried.



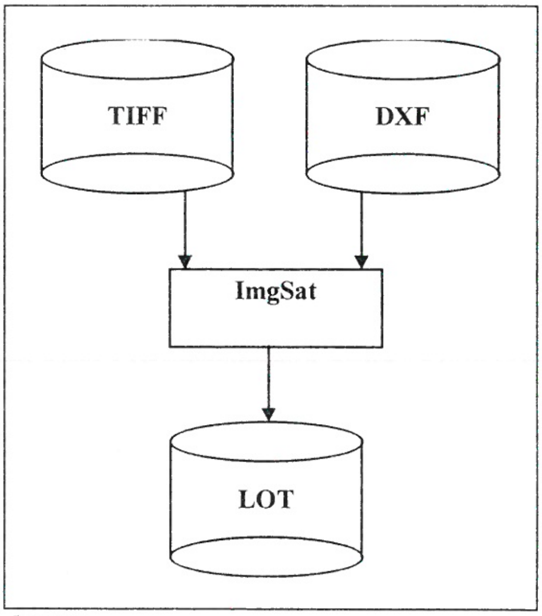
RIRFxx Are also PCI Output Files



2. Application Description

The purpose of our research is to classify plots in the inventoried area according to the crop type. In order to do so, we had to create a system that would accept satellite images – in TIFF file form-as inputs, and plot boundaries as polygons in DXF file. The output of our IMGSAT system is a database of extracted plot images along with related terrain data. The system offers two working options : automatic and interactive. In the latter case just certain plot images may be inserted in the database. In parallel with the plot image extraction, the histogram for the image has been constructed. The histogram is useful in the interpretation and classification process. The plots cut-up from the satellite image is operated using extraction algorithms for the polygonal outlined images or for the Freeman coded bounded images. The proposed extraction methods are in-depth presented in the next Section.

In the picture below the entrances, outputs and extraction method for a control plot, are shown. The primary source is the digital model of the map (from the DXF file) superposed on the digital image of the area locating the plot (TIFF file).



Inputs:

TIFF file d-digitised image file - 8 bits / pixell (256 grey levels), compressed or not compressed LZW ;

DXF file Polygons description – created during the editing stage of the digitised model of the terrain;

Output:
LOT file Plot images.

Cut-up Algorithms

We shall propose two methods. The first algorithm uses directly the polygonal bounds. From the contours described in the VPF.DXF files, the second one will determine the Freeman coded bounds from the contours described in the VPF.DXF files. Both procedures are being developed using methods of detection of the contour interior based on parity check. One major advantage of the second algorithm lies in the fact that there is no need for approximating the circle arcs with polygonal contours.

The two extraction methods are further described.

2.1 A Cut-up Algorithm for Polygonal Outlines

The method has been developed based on the polygonal contour filling algorithm presented in [1].

For each side of the polygon we memorise the co-ordinates x_i, y_i of the end-point with maximum y, and the displacements dx, dy used to calculate the second end-point co-ordinates. All these measures are sorted out in a waiting queue using a descendent order for y and an ascending sorting order for dx. In Examples 1-3 the sides are counted in the sorting order. From the waiting queue there will be moved in an active queue those sides which are crossing the horizontal on the currently processed y. The interior limited by the polygonal contour is identified by a parity check. The sections of the current horizontal to be accepted as belonging to the interior will be those determined by an odd and even intersection with sides from the active queue.

Updating of the active queue with sides from the waiting queue happens by taking into account three types of attributes, to be established when the waiting list is complete. These attributes will represent:

a) **Number of sides to be transferred to the active queue**: 1 - the current one if its end-point is not a local maximum or if its pair for a local maximum is the preceding side; 2 – if a local maximum has been reached; more than 2 – if more than one local maximum have the same value as the current y.

In the next Examples the 'number of sides to load' attribute is:

Side	Attribute	Side to load
1	2	1, 2
2	1	
3	2	3, 4
4	1	
5	1	
6	5	6 – 10
7 – 10	1	
11	5	11 – 15
12 – 15	1	
16	3	16 – 18
17, 18	1	

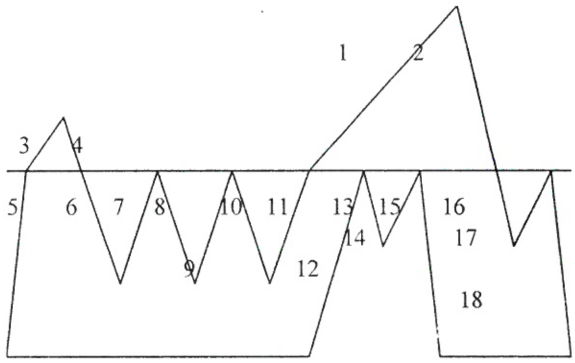


Figure 1

In the second Example the waiting queue is:

Side	Attribute	Side to load
1	2	1, 2
2	1	
3	4	3 - 6
4 – 6	1	
7	3	7 - 9
8 – 9	1	
10	1	

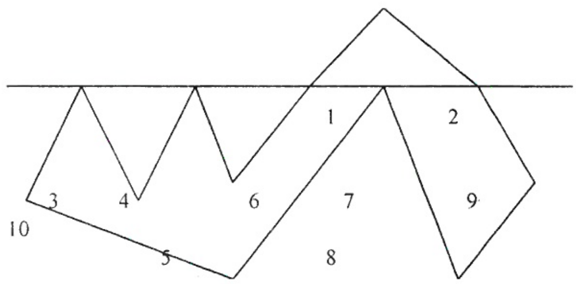


Figure 2

b) **The place where, relative to the current side position in the active queue, to insert a new pair of sides (if any) before discarding the active side.** If the horizontal throw local maximum crosses the active side, the pair of sides determining that maximum has to be loaded in the active queue before discarding the current side. For such cases, the place in the active queue where the new sides will be inserted, is to be decided. This attribute may have one of two values: y_{before} or y_{after}, in order to let decide whether the pair of sides determining the local maximum will be inserted in the active queue before or after the current side. This is the first change with respect to the algorithm presented in [1].

In Example 3 , side 1 has the attribute: $y_{before} = y_A$ (the horizontal throw local maximum formed by sides 7 and 8, respectively 9 and 10 intersects side 1 in A and those sides have to be inserted in the active queue when $y_{current} = y_A$, before side 1). Side 2 has: $y_{before} = y_C$ (a local maximum determined by sides 11 and 12 causes the insertion of those sides into the active queue in a position preceding side 2) and $y_{after} = y_B$ (a local maximum formed by sides 3 and 4 and respectively, 5 and 6 will determine the insertion of those sides into the active queue just after side 2).

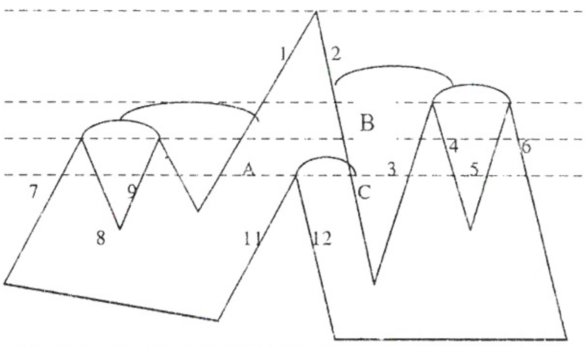


Figure 3

c) **Number of sides to load at the end-point of the current line.** If there is a local minimum at the end-point of the current line no new line has to be loaded on the active queue. In Example 1 sides 6-7, 8-9, 10-11, 13-14, 16-17 have this attribute equal to 0.

A special case, untreated in [1], is the horizontal side, as in Example 4. Horizontal segments will be ignored when the waiting queue is defined. If the two sides adjacent to the horizontal are both beneath it, those two segments will be considered as forming a local maximum.

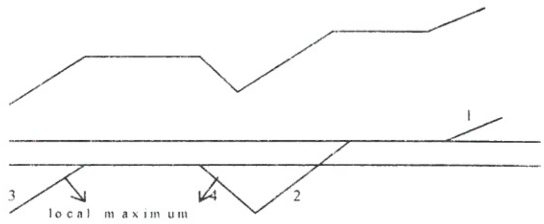


Figure 4

The algorithms for the polygon interior detection are described in the following.

Algorithm 1

1. Sort all polygon edges according to the value Yi and then dX, dY, negative values preceding positive ones.
2. Set the flags for each polygon side:
   flag a — number of local maxima at the same Y. This value shows how many edges must be transferred from the waiting list to the active list .
   flag b(2) — Yi of the lines that must be read when this value of the current Y is reached during the capturing process.
   flag c — "1" if at the end of the current line, the next line from the waiting list must be read;
   "0" if the end of the current line is a local minimum.
3. Set Y=Ymax. (Xmax,Ymax, Xmin, Ymin represent the co-ordinates of the corners of the rectangle containing the polygon area).
4. Transfer first edges from the waiting list to the active list according to the flags
5. As long as the waiting list is not empty do steps 5.1-5.4.
   Begin

   5.1 If the active list is empty or the value of Y is beneath the lower end-point of one of its members then remove those members and transfer from the waiting list to the active list as many elements as indicated by the flags.

   5.2 Extract the space area between alternate members of the active list.

   5.3 If the current value of Y is equal to the Y of "b" attribute of a current edge in the active list, transfer the appropriate edges to the active list before or after the current edge according to the attribute "b".

5.4 Decrement Y.

End

2.2 A Cut-up Algorithm for the Freeman Coded Contours

Starting from the polygonal bounder described in the VPF.DXF file, the Freeman coded contour will be generated for each plot. A pixel of the bounder has the value "1" if it is not a local extreme and the value "2" if it is a local maximum or minimum. For each horizontal crossing the contour there are captured the segments determined by an odd and even intersection with a contour point with the label "1".

The algorithm for the interior detection, the pseudo-code and the experimental results are presented below.

Algorithm 2

1. Build a Freeman code set of contour points starting from the polygonal contour.

2. Set contour points to the value "1" if there are not local minima or maxima else set to the value "2".

3.Y=Ymax. (Xmax,Ymax, Xmin, Ymin represent the co-ordinates of the corners of the rectangle containing the polygon area).

4. While Y >= Ymin capture the area between alternate intersections with contour points having value "1". Decrement Y.

3. Conclusions

The processed satellite images have the dimension of 6000x6000 pixels. The minimal control plot is of 5 ha, which means 500 pixels. Given the large amount of information to be processed, and for the purpose of experimenting the extraction algorithms, we had to use an environment that permitted the development of a 32- bit application, VC++ 4.0.

The developed system performs the following functions:

- load and view satellite image from TIF / BMP file;
- load plots bounders from DXF files;
- generate control plots polygons (UUT);
- cut-up control plot image;
- write / read plot images in / from intermediary files – in a compressed or not compressed LZW form;
- extract plots feature .

The processing time performance was tested on an image of dimension 800x600 and a DXF file of 1095 polygons. The intermediary file with the plots uncompressed images was of 1,990 Kb. Processing time, on a Pentium 133MHz 16Mb RAM computer, is less than 5 seconds.

Figure 5 presents the mainframe of the application.

BIBLIOGRAPHY

1. PAVLIDIS, T., **Algorithms for Graphics and Image Processing**, COMPUTER SCIENCE PRESS, 1982.

2. PAVLIDIS, T., **Structural Pattern Recognition**, SPRINGER - VERLAG, Berlin-Heidelberg, 1977.

3. SCHOTTEN, L., JANSSEN, L. and NIEUWENHUIS, G. J., **Comparison of Optical and Microwave Satellite Data for Land Cover Inventory Purposes Using Digital Field Boundaries**, ADVANCES IN REMOTE SENSING, Vol.2, No.3, 1993.

4. SMITH, D. and MAJOR, J., **Complementary of Radar and Visible Infrared Sensors in Assessing Rangeland Conditions**, Symposium Canadien sur Télédétéction, Québec 1993.

5. PEDERSEN, J.P., GUNERIUSSEN, T. and JOHANSEN, B., **Application of ERS-1 SAR Data in Combination With LANDSAT and SPOT Data for Forestry Mapping in a Norwegian Forest Region**, International Geoscience and Remote Sensing Symposium Proceedings, Tokyo, 1993, pp. 46-48.

6. TOUTIN, T., **Intégration d'images multi-sources: prèmiers resultats avec SPOT et**

RADAR aeroporte, CANADIAN JOURNAL OF REMOTE SENSING, Vol. 20, No.1, 1991.

7. PCI Remote Sensing Corporation, **Training Data Set, V.6.0**, 1996.

8. PCI Remote Sensing Corporation, **Getting Started. Desktop, V. 6.0**, 1996.

9. PCI Remote Sensing Corporation, **User's Guide, V. 6.0**, 1996.

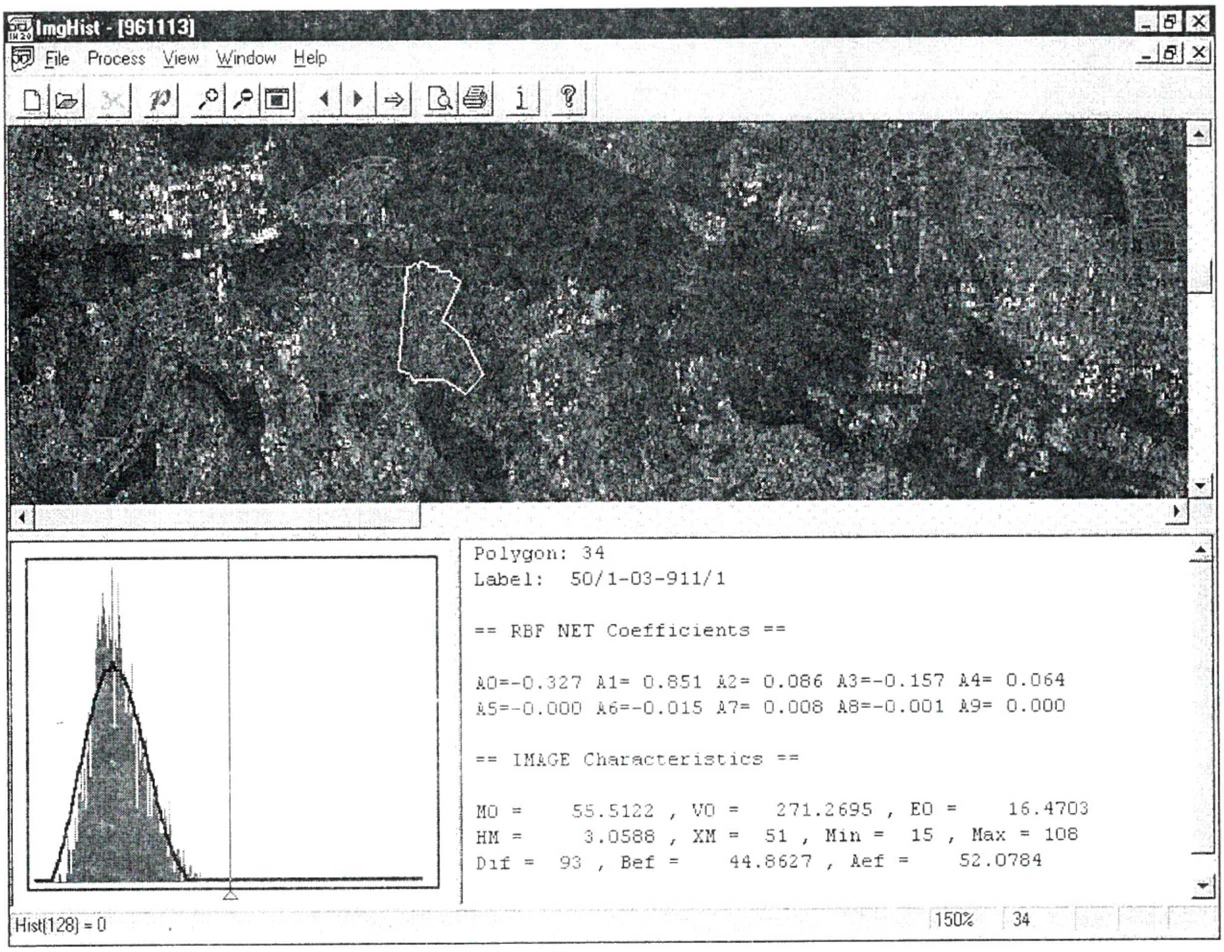


Figure 5