IMAGE SEGMENTATION USING CONDITIONAL RANKORDER FILTERING

N.J. Mulder, head image processing lab.
K. Sijmons, senior scientist cartography
International Institute for
Aerial Survey and Earth Sciences (ITC)
The Netherlands
Commission III

ABSTRACT

Segmentation by the repeated application of 3x3 conditional rankorder operators is presented here as a better alternative to region growing by conditional smoothing or the use of edge detection methods.

Various interpretations of the method are discussed such as local clustering, label relaxation, local structure linking, and facet modeling.

INTRODUCTION

It is our aim to develop methods for computer aided updating of cartographic data using various remote sensing data sources.

In order to compare old map data, previous remote sensing data (such as aerial photography) and new remote sensing data a common abstract description of cartographic elements must be used.

Pattern recognition and specially structural pattern recognition provide means for the extraction of structural information from both remote sensing data and cartographic data.

A first step in pattern recognition is spatial feature extraction by means of segmentation. Segments are merged into objects. Objects are related to each other in a high order structure which contains topologic elements.

In this paper we evaluate conditional rankorder filters (3x3). The approach can be characterised as heuristic label relaxation. The remote sensing data used here is of the type integer (byte). Rankorder filters only shift data and we use an itterative process to assign values to segments, values are used as local labels. After convergence of the local labeling global labels are assigned and structural features are derived for segments. Based on these features segments are merged into objects. Objects form together a high order structure which is the bases for a scene (map) description.

The choise of conditional rankorder filters for label relaxation followed from our experience that edge detection filters perform poor in area segmentation and just acceptable in boundary detection and poor in line detection.

Conditional smoothing as a method of area segmentation by local clustering was used first [Mulder, 1983]. It requires statistical information which is suitably provided by local rankorder arrays. Smoothing results in real values which have to be rounded off resulting in slow convergence in repeated application of conditional smoothing.

As rankorder information is needed in conditional smoothing and as conditional repeated rankordering has a higher convergence rate than conditional smoothing we have abolished conditional smoothing in favour of conditional rankorder filtering for both line segmentation and area segmentation.

A disadvantage of conditional rankorder filters compared with conditional mean filters is that the number of neighbours belonging to the same class is not directly available in rankorder filters. In conditional mean filters this number can be used to select candidate line elements or edge elements.

Operator size is derived from primitive feature size. With satellite remote sensing data detectable line features are usually a fraction of the sample size (point response function in x,y).
The number of possible features in a subimage of operator size is two to the power of the number of pixels in the subimage. The dimension of the subimage feature space should be kept small. The minimum symmetric operator >1 is a 3x3 operator with corresponding 3x3 subimages.

Relaxation is a concept from physics where the state of atoms or molecules in a raster may change in a discontinues way forming groups of e.g. molecules with the same magnetic polarity.

Translated to pattern recognition a relaxation process will lead to an improved local consistency of values or labels. We use local structure elements for the local decision making which is involved in relaxation.

Structure elements are defined on 3x3 subimages. Comparing the central subimage sample with its eight neighbours and deciding for each neighbour whether or not the sample belongs to the same class as the central sample gives 2**8=256 possible binary structure elements. We define a subset of structure elements and from the effect of various rankorder operations on the subset of primitive structures the rules for label reassignement "relaxation" are derived in a heuristic way.

The process of reasigning labels is repeated until the difference between the results of two operations is less than a visually determined threshold.

Segment features are used to merge smaller segments into larger segments and to connect segments into objects. In this study we are particularly interested in line shaped objects but the method is generally applicable to other types of object shape classes.

Because of limited preparation time we concentrate in this paper on one method of segmentation into primitive regions only.

RANKORDER OPERATORS

For line feature detection we use 3x3 operators. For each position in an output file: output at i,j a corresponding input subimage array is defined with linear index k:

```
                                  1 2 3
        subimage[k]     k =       4 5 6
                                  7 8 9
```

All subimages, subimage[k] values (0,255) are sorted into array rank[r] with r=1 the lowest value, r=5 the median value and r=9 the highest value in rank[r].

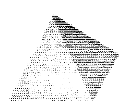
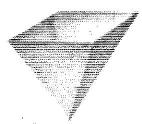
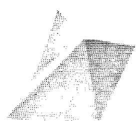
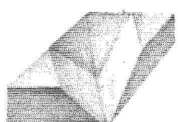
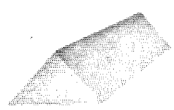
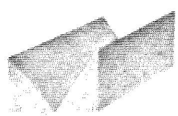
Algorithm:

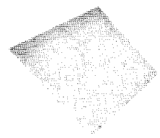
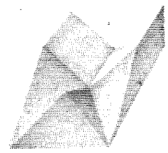
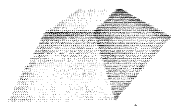
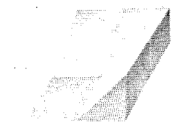
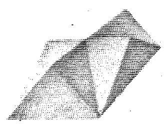
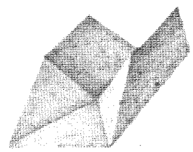
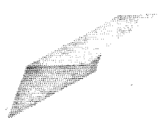
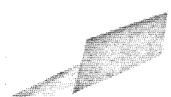
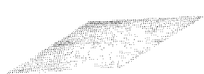
```
   do for all samples i,j
      get subimage at i,j
      sort subimage[1..9] into rank[1..9]      output:=function(rank[r])
      put output at i,j

   repeat the algorithm (new input is old output) until result converges
   to stable output values.
```

PRIMITIVE STRUCTURE ELEMENTS

Segment label relaxation is guided by the effect each relaxation step has on local structure parameters.
Conceptually the structure is defined by the local bit pattern of samples which belong to the same class as the central sample subimage[5] -- "1" and the samples which belong to the complementory class "0".
From the possible 512 bitpatterns we display the relevant subset of bit-patterns with central bit=1 and meaningfull structural interpretation.
ROT is the number of rotational invariants of the bit pattern of the non-central elements.

bits	#0	#1	ROT	facet values		facet figures	
point 000 010 000	8	1	1	--- -+- ---	+++ +-+ +++		
end of line 100 010 000	7	2	8	+-- -+- ---	+++ +-+ ++-		
lines, ridge edge, valley edge 010 010 010	6	3	4	-+- -+- -+-	+-+ +-+ +-+		
100 010 010	6	3	4	+-- -+- -+-	-++ +-+ +-+		
100 010 100	6	3	4	+-- -+- +--	-++ +-+ -++		

slopes						
010				-0+	+0-	
010	6	3	(4)	-0+	+0-	
010				-0+	+0-	
Y-crosses						
101				+-+	-+-	
010	5	4	8	-+-	+-+	
010				-+-	+-+	
corners						
110				++-	--+	 
110	5	4	8	++-	--+	
000				---	+++	
real lines						
100				+--	-++	
110	5	4	8	++-	--+	
010				-+-	+-+	
X-crosses						
010				-+-	+-+	 
111	4	5	2	+++	---	
010				-+-	+-+	
T-crosses						
111				+++	+-+	
010	4	5	8	-+-	+-+	
010				-+-	---	
111				+++	---	
010	4	5	8	-+-	+-+	
100				+--	-++	
111				+++	---	
010	4	5	8	-+-	+-+	
001				--+	++-	
step edge						
011				-++	--+	 
011	3	6	8	-++	--+	
011				-++	--+	
solid						
111				+++	---	 
111	0	8	1	+++	---	
111				+++	---	

The size of the subset relative to the full set is derived from the combinatoric formula that with n0 and n1 being the number of "0" and "1" in an eight bit word the possible number of combinations is given by the formula:

$$C(n0,n1) = 8! / (n0! \times n1!)$$

In the following table C(n0,n1) is tabulated together with the number P of primitives defined in the previous section.

n0	n1	C	P
8	0	1	1
7	1	8	8
6	2	28	12
5	3	56	32
4	4	70	26
3	5	56	8
2	6	28	0
1	7	8	0
0	8	1	1
sum		256	88

About 1/3 of the possible binary outer bit patterns have been enumerated. Slopes and lines (roof edges) have a dual interpretation in terms of bit patterns and therefore the facet interpretation is also given for all primitives.

Although a full enumeration of all bit patterns could be used for the unique labeling of all primitives followed by a syntactic analysis we use a subset here for guiding the relaxation process only. The enumerated primitives are the ideal primitives here and the other patterns must be transformed into the "ideal" elements through repeated conditional rank-order filtering.

CONDITIONAL MEDIAN FILTER

The median filter has the property of noise removal and it will converge segments to their median value. In the algorithm for conditional rankorder filtering we had:

output: = function (rank[r])

For the median filter the function is defined by:

output: = rank[5]

The effect on primitive structural elements is:

points, lines, corners, some crosses will be filtered away because they contain a majority of elements which are different from the central element. From five "1" 's on the central element belongs to the "in" (= "1") class and these classes of structural elements will be strengthened. An exception to the above rule on the number of "1" 's in the bit pattern is the class of slopes where the pattern of "1" 's only indicates the direction of the label relaxation. In this special case the more relevant representation of the structure is the facet model which shows clearly that the central element belongs to the "in" class and will thus be kept and strengthened.

From the tabulation of primitive structure elements it can be seen that most crosses, step edges and solids will be strengthened.

To solve the problem of the loss of line structure we first put a condition on the difference between the original central value subimage[5] and the median value rank[5]. The function which maps the input subimage intò an output value is now:

Algorithm 1:

```
IF ABS ( subimage[5] - rank[5] ) < threshold
THEN     output : = rank[5]
ELSE     output : = subimage[5]
```

The condition on the difference between the central value and the median can be interpreted as an estimate of a Laplacian in a noisy environment. Estimating a Laplacian in this way is equivalent to the method of polynomal surface fitting for finding second order differences in a noisy subimage [Haralick, 1984].

The selection of a threshold is critical for the balance between noise suppression on the one hand and strengthening of line elements on the other hand.

Figure 1a shows original Landsat thematic mapper band-3, TM3.
Figure 1b shows the result of median filtering applied to TM3.
Figure 1c shows the absolute of the difference between original and median which after thresholding with threshold = 3 provides the answer to the historical question:

"to smooth or not to smooth ..
that is the question!"

Figure 1d shows the effect of four itterations of conditional median filtering.

Individual noise points can be separated from points on a line by pre-filtering the condition file with a logic filter which keeps the central "1" value only if there are two or more "1' neighbours and else will put the central value at "0".

Figure 1e shows the effect of the logic "minority removing filter on the condition file" Boolean (abs (orig-median) $<$ 3).

Figure 1f is the result of applying condition file 1e over three itterations of conditional median filtering.

PIECE WISE CONSTANT RANKORDER FILTERING

The fact that slopes are concerved in conditional smoothing means that high contrast segments are surrounded by one or more rings of transition elements or slope elements.

As the aim of relaxation labeling can also be stated is being a segmentation into Piecewise constant photoncount "intensity" surfaces, we try a new function:

Algorithm 2:

(ORG-MED) : = subimage [5] - rank [5]

IF (ORG-MED) > threshold THEN output: = rank [8]

IF (ORG-MED) < -threshold THEN output: = rank [2]

IF (ORG-MED) < threshold AND (ORG-MED) > -threshold THEN output: = rank [5]

The function of the threshold is again to discriminate between structure and noise.

Figure 2a shows the new condition file with:
((orig-med) $\geq$ 2)→+1, ((orig-med) $\leq$ - 2)→-1 else → "0".
Figure 2b shows the result of three itterations of piece wise linear smoothing with condition file 2a.

Refering to the enumeration of primitive structure elements, the effect of this new function is:
Suppression of single samples which are statistical outliers, all structures with #1 from 2 to 8 will be kept and the integer value will converge to a local submaximum value (rank[8]) for bright spatial features or to subminimum values (rank[2]) for dark spatial features.

The above function is also equivalent with a conditional integer value expand algorithm.

In terms of surface facet interpretation (rank[8] - rank[2]) is a slope (=abs(grad(band))) estimator and the condition for going up or down the slope is (subimage[5] - rank[5]) which is a Laplacian estimator.

Figure 2c shows the slope estimator (rank[8] - rank[2]) in pseudo greyscale.

Postscript:
We have come accross situation where a first itteration with rank[2] or rank[8] replacement was justified for noise suppression but further itterations can use rank[1] or rank[9] replacement for convergencence to larger segment sizes.

SPATIAL FEATURES IN RANKORDER OPERATORS

As we have been using first derivative and second derivative values in the conditions in our decision making our methods are equivalent to sequential decision making with spatial features. It is therefore interesting to study the featurespace plot of our structural primitive classes on the features "central-value - median" and "local contrast = rank[9] - rank[1]"

Figure 3a shows the featurespace plot of structure primitive classes on rankorder estimates of the Laplacian and maximum-slope (local contrast). The relevant decision boundaries are shown in the featurespace plot.

From figure 3a it is apparent that rank[9] - rank[1] is not a pure slope feature it is rather a scale factor for the local contrast multiplicatively combined with the "rank-Laplacian" = central value - median.
In the featurespace of figure 3a we find four clusters of ideal structure classes. Mixed classes and the effect of noise produce featurevectors outside the representation of ideal classes as shown in figure 3b which is a 2-dimensional histogram on the same features as figure 3a.

CONCLUSIONS

Conditional rankorder itterative label relaxation has a faster convergence rate than the related method of region growing by conditional smoothing.

The conditions used in the relaxation filter are equivalent to a classification based on two features: a rankorder estimation of the Laplacian and the local number of "the same" values.

The performance on remote sensing data (TM3) for line detection is reasonable when one considers the method was designed for segmentation of the image into initial primitive segments which have to be merged into larger segments and objects by the use of segment features.

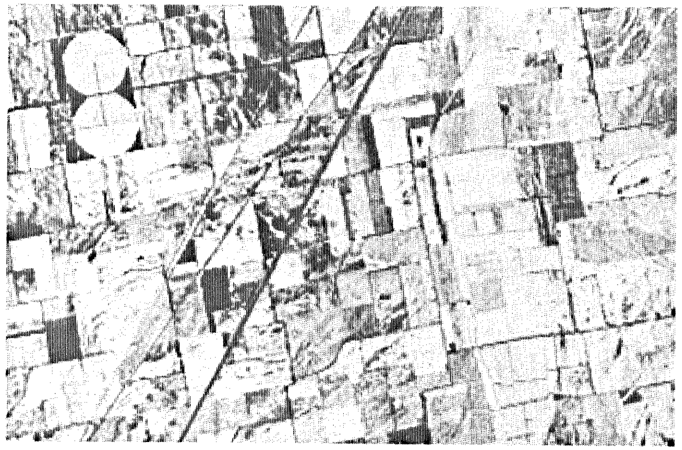
The use of only two local features in the conditional part of the relaxation operators is a limitation. Further developments will include a full enumeration of all 256 bit patterns in a 3x3 subimage and a look up table classification (condition) defining the action of the relaxation operator. The original rankorder features will be used for the transformation (classification) of the local values into the local bit pattern.

Elongated segments can be edge segments or line segments. As they are not completely separable on local operator level they must be separated on segment level. Merging primitive regions into larger objects is the next step in line object detection. It is not treated here as it involves only classical pattern recognition methods.

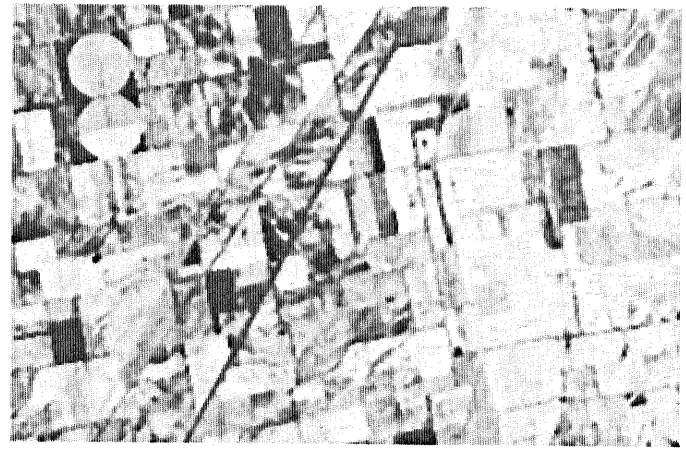
REFERENCES

Haralick R.M. "Digital Step Edges from Zero Crossing of Second Directional Derivatives"
IEEE, vol.PAMI-6, no. 1, January, 1984.

Mulder N.J. "Decision making"
paper symposium Graz, 1983
copies: ITC P.O. Box 6, Enschede, Netherlands.

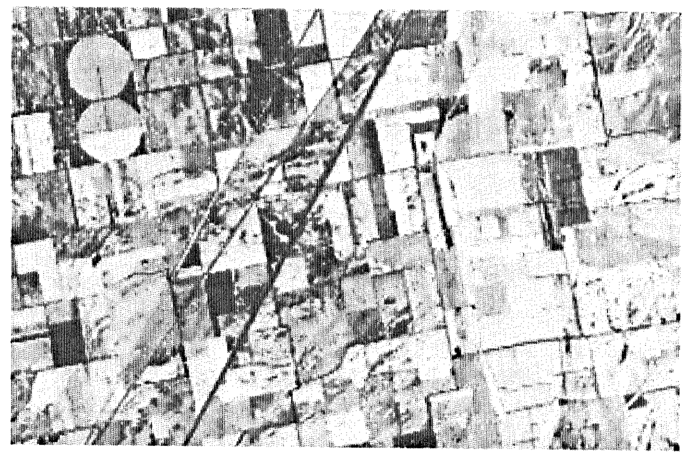


1A. TM3 Original

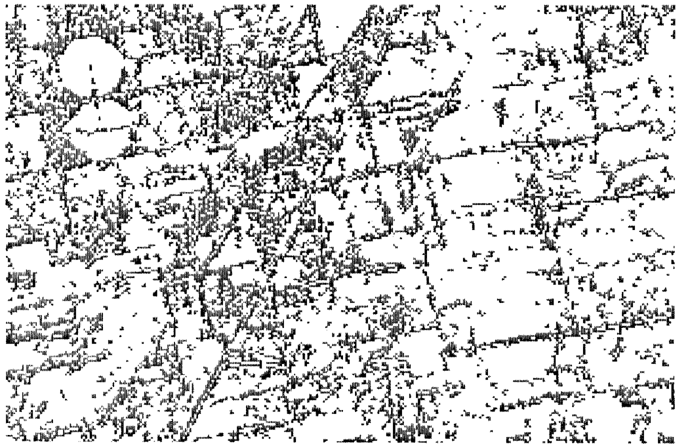


1B. TM3 Median

1C. TM3 ABS(orig. – median) > 3

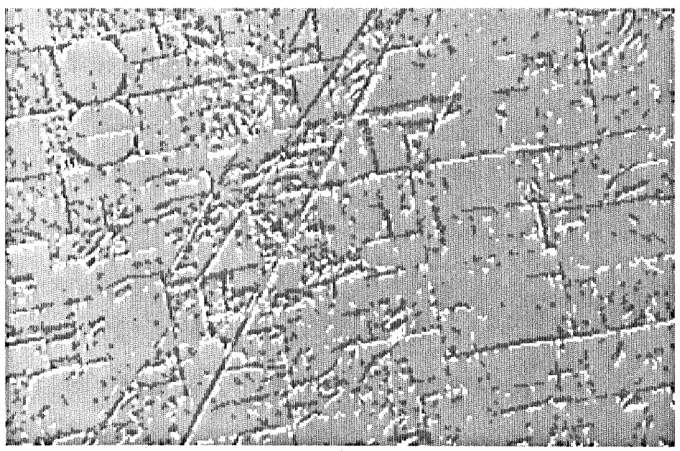


1D. 3 iterations cond. median



1E. ABS(orig. – median) > 3 and # 1 > 2

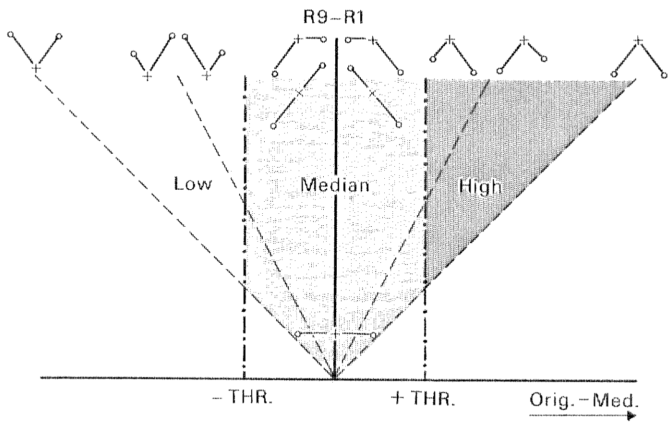
1F. 3 iterations cond. median (1E)



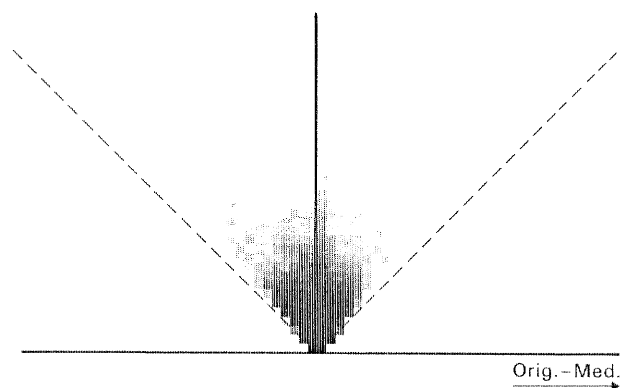
2A. Condition file > 3 = Wh, < – 3 = Bl, else grey

2B. 3 iterations Maximum, Median, Minimum (2A)

2C TM3 Rank 8 – Rank 2

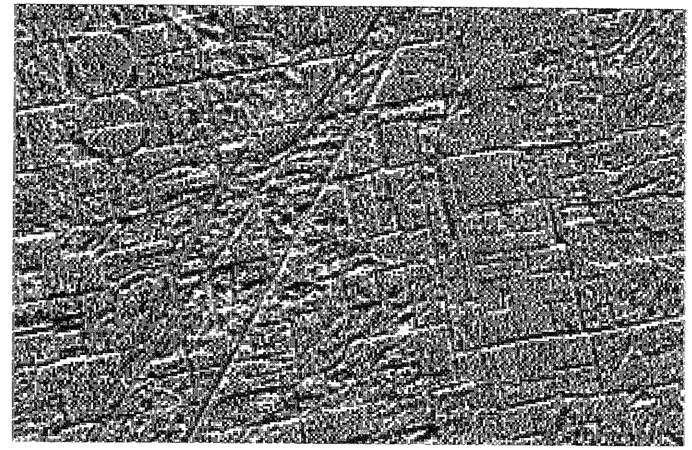


3A Featurespace (Orig – Med.) (R9 – R1)

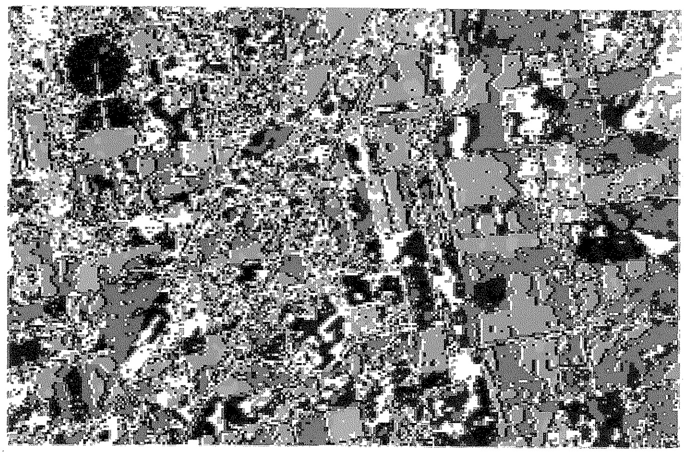


3B 2 Dim. Histogr. (Org. – Med.) (R9 – R1)

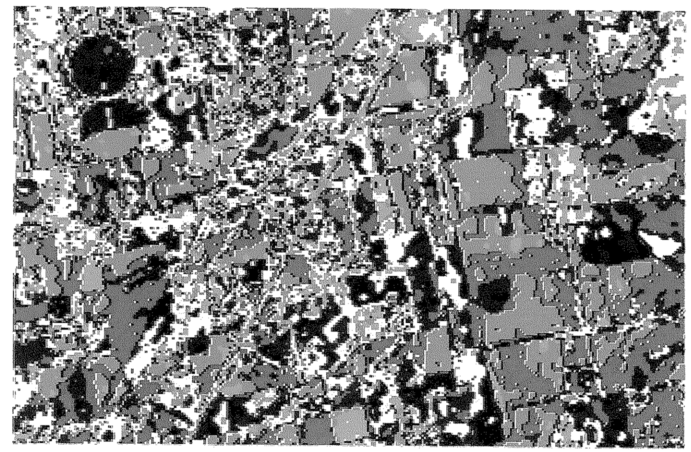
F1, TM3 Original, pseudo colour



F2. Orig. – Median, pseudo colour



F3. Conditional Median, 3 iterations TH=3



F4. Conditional HI – MED – LO, 3 iterations TH=3