7.7-10.115

# A NEW IMAGE ENHANCEMENT ALGORITHM WITH APPLICATIONS

## TO FORESTRY STAND MAPPING

Job Order 75-315

Prepared By

Lockheed Electronics Company, Inc.

Aerospace Systems Division

Houston, Texas

Contract NAS 9-12200

For

EARTH OBSERVATIONS DIVISION

*National Aeronautics and Space Administration*

# LYNDON B. JOHNSON SPACE CENTER

## *Houston, Texas*
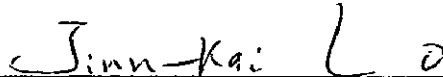
June 1975

LEC-6178

i

# A NEW IMAGE ENHANCEMENT ALGORITHM WITH APPLICATIONS TO FORESTRY STAND MAPPING

Job Order 75-315

PREPARED BY

_____

Edwin P. F. Kan, Principal Scientist

_____

Jinn-Kai Lo, Scientist

APPROVED BY

_____

I. E. Duggan, Supervisor
Forestry Applications Section

_____

O. N. Brandt, Acting Manager
Earth Observations Exploratory Studies
Department

Prepared By

Lockheed Electronics Company, Inc.

For

Earth Observations Division

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

June 1975

## TECHNICAL REPORT INDEX/ABSTRACT

| 1. TITLE AND SUBTITLE OF DOCUMENT | 2. JSC NO. |
|---|---|
| A NEW IMAGE ENHANCEMENT ALGORITHM WITH APPLICATIONS TO FORESTRY STAND MAPPING | JSC-09695 |

| 3. CONTRACTOR/ORGANIZATION NAME | 4. CONTRACT OR GRANT NO. |
|---|---|
| Aerospace Systems Division, Lockheed Electronics Company, Inc. | NAS 9-12200 |

| 5. CONTRACTOR/ORIGINATOR DOCUMENT NO. | 6. PUBLICATION DATE (THIS ISSUE) |
|---|---|
| LEC-6178 | June 1975 |

| 7. SECURITY CLASSIFICATION | 8. OPR (OFFICE OF PRIMARY RESPONSIBILITY) |
|---|---|
| Unclassified | R. Dean Bratton |

| 9. LIMITATIONS | 10. AUTHOR(S) |
|---|---|
| GOVERNMENT HAS UNLIMITED RIGHTS [X] YES [ ] NO<br>IF NO, STATE LIMITATIONS AND AUTHORITY | Edwin P. F. Kan<br>Jinn-Kai Lo |

| 11. DOCUMENT CONTRACT REFERENCES | 12 HARDWARE CONFIGURATION |
|---|---|
| WORK BREAKDOWN STRUCTURE NO.<br>Job Order 75-315 | SYSTEM |
| CONTRACT EXHIBIT NO. | SUBSYSTEM |
| DRL NO AND REVISION | MAJOR EQUIPMENT GROUP |
| DRL LINE ITEM NO. | |

13. ABSTRACT

The theory and applications are presented of a new image enhancement algorithm which refines computer classification maps of multispectral data. The refinement eliminates connected sets smaller than a prespecified size and merges them to the surrounding area.

Forestry timber stand mapping requires small geographic areas to be absorbed by surrounding large areas to form homogeneous stands. This homogeneity is often incompatible with the statistical formulation of homogeneity. Elements within a timber stand which should be labeled as one feature often correspond to more than one class mapped by existing computer classification techniques. The new algorithm is designed to postprocess classification maps to result in more usable timber stand maps.

The new image enhancement technique is compared with an accepted neighbor-checking postprocessing technique, demonstrating the superiority of the new technique for forestry stand mapping.

14. SUBJECT TERMS

| | | |
|---|---|---|
| Algorithms | Image enhancement | Pattern recognition |
| Postprocessing classification | Mapping | Remote sensors |
| Earth resources | Multispectral imagery | Timber identification |

CONTENTS

vi

FIGURES

# A NEW IMAGE ENHANCEMENT ALGORITHM WITH APPLICATIONS
# TO FORESTRY STAND MAPPING

By Edwin P. F. Kan and Jinn-Kai Lo
Aerospace Systems Division, Lockheed Electronics Company, Inc.

## 1.0   INTRODUCTION

Satellite and aircraft multispectral scanner (MSS) data have been classified and analyzed for land use planning, forestry mapping, resource inventories, and agricultural applications (refs. 1 through 3). Computer classification maps from these studies always appear to need refinement, in the sense that "holes" exist on the maps. For example, a geographic area supposedly containing one feature, such as a forest, often is not classified entirely as a forest on the map. These spotty, discontinuous maps normally are not satisfactory to the user, who desires smooth, clean maps. This report describes a new image enhancement algorithm to "clean up" computer classification maps with applications to forestry stand mapping.

Forestry classification maps from the processing of aircraft MSS data particularly need cleaning. For example, with data at a spatial resolution of less than $(10 \text{ meters})^2$, i.e., with one picture element (pixel) covering $(10 \text{ meters})^2$ on the ground, more than one type of pixel would be observed from aircraft over a "homogeneous" timber stand. Some pixels in a stand would represent crowns of the predominant tree type. Other pixels would represent understories, clearances, and shadows, among the trees. A computer classification of such MSS data would produce heterogeneous maps in which pixels within a stand would be classified into more than

one statistical class. A widely accepted classification
rule is the training-field, maximum likelihood classifier
(ref. 4).

The problem of holes in classification maps is further
compounded by the U.S. Forest Service definition of a stand
type. A stand type refers to an area larger than 4 hectares
in which more than 51 percent of the trees in the canopy belongs
to one tree species or group of species. A stand smaller
than 4 hectares should be absorbed into and classified the
same as the surrounding area under the present system of stand
type mapping. Because existing statistical classification
techniques have not been designed to clean up these small
areas, computer generated classification maps would contain
undesirable holes which should not exist on final stand type
maps.

To obtain usable timber stand classification maps, the
computer classification maps need to be postprocessed to
remove the holes. Some investigators have used spatial infor-
mation to fill the holes by checking neighboring pixels as in
the RECLASS computer program (ref. 5). In that program, a
pixel surrounded by four neighboring pixels belonging to one
different class is reclassified into the class of its neighbors.
Variations of this neighbor rule have been explored.

A natural generalization of the neighbor rule is that any
connected set of pixels is eliminated if the set contains less
than a prespecified number of pixels. The whole set is then
reclassified into the class of the surrounding pixels. By a
connected set, it is meant that every pixel in the set has at
least one neighbor in the same set; set with one pixel is also
connected. This postprocessing concept is indeed appropriate
for timber stand mapping. It will be explored in this paper,
formalized into a new image enhancement algorithm, and applied
to a forestry application using aircraft MSS data.

## 2.0 CONNECTIVITY

A set of pixels is connected if every pixel is a left-right or top-bottom neighbor of at least one other pixel. The set with one pixel is also connected. The left-right connection is along a scan line, or along the same row of the two-dimensional array. The top-bottom connection is across a scan line, or along the same column of the two-dimensional array.

The five pixels labeled "1" in figure 1(a) are connected according to the above definition, but the five pixels labeled "1" in figure 1(b) are not. The new image enhancement algorithm is developed, programed, and tested according to the left-right, top-bottom definition of connectivity. However, the algorithm can be generalized easily for diagonal connectivity in which pixels lie diagonally adjacent to one another; figure 1(b) is an extreme example of diagonal connectivity.

Formal mathematical definitions of connectivity and related concepts use terms such as "four-path" and "four-connected set." These concepts were studied by researchers in the pattern recognition of digital pictures, particularly handwritten character recognition. References 6 through 11 cover the locating of edges in digitized pictures, thinning, and seeking skeletons in pictures. The new image enhancement algorithm is a variation of the previous research concepts and applies to forestry timber stand mapping.

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

(a)   Left-right and top-bottom

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

(b)   Diagonal

Figure 1.— Connectivity of pixels labeled "1."

## 3.0 THE NEW IMAGE ENHANCEMENT ALGORITHM

### 3.1 CLASSIFICATION IMAGES

The new algorithm is designed to refine classification images obtained by processing MSS data through techniques such as the training-field, maximum likelihood classification (ref. 4). Classification images are two-dimensional arrays in which each pixel has a label representing a class, such as a timber type.

To simplify the notations without loss of generality, a classification image is assumed to be binary, in which each pixel has the labeled value 0 or 1. This two-class classification image is a general representation, because any map can be examined one feature at a time. The pixels labeled 1 represent the features of interest; and the pixels labeled 0 are of no interest. This report presents the new algorithm in terms of images of 0's and 1's.

The algorithm modifies the input binary image of 0's and 1's. The first steps of the modification are to find all the connected sets in the image and to count the number of pixels in the sets. The labels of the sets smaller than a prespecified size $n_0$ are then changed so that 1 becomes 0 and 0 becomes 1. In this change, sets smaller than $n_0$ are absorbed by the surrounding sets.

The ways of changing labels are easy to modify. For example, the programed algorithm can be modified to permit small connected sets of 1's to be absorbed and changed to 0's and to leave small sets of 0's unchanged. In section 4.2,

the tertiary images to be refined have three labels, -1, 0,
and 1. The program was developed to modify only the two
classes 0 and 1 and to leave the class -1 unchanged.

## 3.2  SKELETON FLOW OF THE ALGORITHM

Because of the finiteness of computer core memories, the
image must be analyzed block by block. To comply with the
normal line-by-line format of recording an image, a fixed
number of lines in the image is analyzed at one time. To
accommodate the worst case of a connected set one pixel
wide down a column of the image, $n_0$ lines must be studied
at one time; $n_0$ is the minimum size of a set not to be
modified. After these $n_0$ lines are analyzed, the start
line of the $n_0$ lines is purged. The line after the stop
line of the $n_0$ lines is then input to the computer memory
to make a new set of $n_0$ lines.

For a data image array of the size  M  by  N  in which
the elements are 0 and 1, three integer arrays are programed
in the computer.

1.  Data array $A(I,J)$:  $I = 1, \cdots, n_0$ ;  $J = 1, \cdots, N$

2.  Tag array $S(I,K,L)$:  $I = 1, \cdots, n_0$ ;  $K = 1, \cdots, N$ ;
    $L = 1,2,3$

3.  Count array $C(P,Q)$:  $P = 1, \cdots, N$ ;  $Q = 1, \cdots, 2n_0+3$

The use of the arrays and the skeleton of the algorithm
are described as follows.

1.  Initialization:  Input $n_0$ lines of data and store
    them in the data array $A(I,J)$.

2. Tagging: Follow the lines of A(I,J) sequentially to search for along-the-line connected sets. Tag the sets using integers starting from 1 and store them in the tag array S(I,K,L).

3. Clustering: Check the across-line connectivity of tagged sets. Modify the tags so that each connected set has a unique tag.

4. Counting: Count the pixels in the clustered, tagged sets. Store the totals in the count array C(P,Q).

5. Label Modification: Change the 0 labels to 1 and 1 to 0 in all clustered, tagged sets smaller than the minimal size $n_0$.

6. Reiteration: Input a new line into A(I,J) and return to step 1. Stop if all lines in the data image are exhausted.

The appendix has a detailed description of steps 1 through 6.

## 3.3 TAG AND COUNT ARRAYS

In the tagging step 2 of section 3.2, the program goes through the data array A(I,J). For example, in line $\hat{I}$* the program finds and establishes the $\hat{K}th$ connected set, stores the start pixel in $S(\hat{I},\hat{K},1)$, stores the stop pixel in $S(\hat{I},\hat{K},2)$, and tags it in $S(\hat{I},\hat{K},3)$. Evidently, the $\hat{K} + 1$ connected set on the same $\hat{I}th$ line has

$$S(\hat{I}, \hat{K} + 1, 1) = S(\hat{I},\hat{K},2) + 1$$

---

*I denotes an integer variable; $\hat{I}$ denotes a particular integer value of I . The same applies for K, L, P, and Q .

and

$$S(\hat{I}, \hat{K} + 1, 3) = S(\hat{I}, \hat{K}, 3) + 1$$

The count array provides $N$ counters, each with $2n_0 + 3$ entries. The $N$ counters are sufficient for analyzing any image, although the image might have more than $N$ connected sets. Actually the $N$ counters are used cyclically. The number of times that a particular $Pth$ counter is used is stored in $C(P,3)$.

In the counting step 4 of section 3.2, the program goes through the tag array $S(I,K,L)$. For example, in line $\hat{I}$ at the $\hat{K}th$ clustered, tagged set, the program has two alternatives.

1.  The $\hat{K}th$ set is connected to the above line $\hat{I} - 1$ .

2.  The $\hat{K}th$ set is not connected to the above line $\hat{I} - 1$ , and is a new clustered, tagged set.

In alternative 1, the program looks up the $\hat{P}th$ counter, where $\hat{P} = \text{mod}[S(\hat{I},\hat{K},3),N]$ . The mod$(M,N)$ denotes the remainder of $M$ divided by $N$ . The $C(\hat{P},1)$ contains a $\hat{Q}$ value, where $4 \leq \hat{Q} \leq 2n_0 + 3$ , such that the program stores the start pixel of the $\hat{K}th$ set in $C(\hat{P},\hat{Q})$ and the stop pixel of the set in $C(\hat{P}, \hat{Q} + 1)$. The program increases the current $\hat{Q}$ by 2. In alternative 2, a new counter is used for storing the start and stop pixels of the $\hat{K}th$ set. The entries $C(P,4)$ to $C(P, 2n_0 + 3)$ of any $Pth$ counter store the start and stop pixels of the clustered, tagged sets from the first line until the last line in which these sets appear. The $C(P,2)$ stores the current total number of pixels in the clustered, tagged sets.

## 4.0  TIMBER STAND MAPPING

A goal of the Forestry Applications Exploratory Studies Project (ref. 12) is to map timber stands by computer-aided analyses of MSS data collected by aircraft and satellites. The MSS data collected over the Sam Houston National Forest in east Texas were analyzed with a computer algorithm of the LARSYS-type (ref. 4). Computer classification maps were generated and postprocessed by the new image enhancement algorithm.

### 4.1  DATA SETS

The three MSS data sets used are 1X, 2X, and 3X. The 1X data were collected during Mission 230 of the C-130 aircraft of the National Aeronautics and Space Administration on March 21, 1973, at a 3-kilometer altitude. The data have a spatial resolution of (8 meters)$^2$; i.e., a pixel covers an 8- by 8-meter area on the ground. In the Forestry Applications Exploratory Studies Project, the 1X data set is called edit 9 (ref. 13) and covers about 250 scan lines with 700 pixels per line.

The 2X and 3X data are electronically reduced from 1X by pixel-averaging (ref. 14). The 2X and 3X data cover the same geographic area of 1X but have spatial resolutions of (16 meters)$^2$ and (24 meters)$^2$, respectively. The numbers of scan lines and pixels per line are reduced correspondingly by two and three times. The 2X and 3X data simulate the data that would be acquired at two and three times the altitude of 1X.

Figure 2 is a three-channel rendition of the 1X, 2X, and 3X MSS data. Because of the pixel-averaging process producing 2X and 3X data from 1X, the 2X data have a smoother texture than 1X, and the 3X data have a smoother texture than the 2X. The physical ground features on the data sets consist of a regeneration stand, pine stands, mixed hardwood stands, cutover land, and open areas of roads, pipelines, and ponds.

Using the same parameters in a clustering procedure, the 1X, 2X, and 3X data were analyzed, resulting in the cluster classification maps in figures 3(a), 4(a), and 5(a). A clustering procedure groups the pixels with similar data values into clusters, or homogeneous spectral signature classes. A cluster classification map is the image of the clustered data in which each pixel is labeled with the class to which it belongs.

Figures 3(a), 4(a), and 5(a) show three-class decompositions of the data. The lightest shade is pine stands; the medium shade, hardwood stands, cutover land, and regeneration areas; and the darkest shade, open areas. These figures are by no means ultimate timber stand classification maps, but they suffice here to illustrate the image enhancement power of the new algorithm.

## 4.2 APPLYING THE ALGORITHM

Figures 3(a), 4(a), and 5(a) are tertiary images with the labels -1, 0, and 1. The algorithm changes 0 to 1 and 1 to 0 when necessary, leaving -1 unchanged. The -1 label corresponds to the open area represented by the darkest shade in the maps.

Figure 2.— 3-channel color renditions of MSS/24 data.

NASA S 75 11007 · TRI

**FOREST CLASSIFICATION MAPS**
**BEFORE AND AFTER POSTPROCESSING**
SAM HOUSTON NATIONAL FOREST 1X DATA

(a) BEFORE POSTPROCESSING

(b) AFTER POSTPROCESSING, WITH $\eta_0 = 5$

Figure 3.— Forest classification maps before and after postprocessing Sam Houston National Forest 1X data.

Figure 3.— Concluded.

NASA S75 11009

TRI

# FOREST CLASSIFICATION MAPS
## BEFORE AND AFTER POSTPROCESSING
### SAM HOUSTON NATIONAL FOREST 2X DATA

(a) BEFORE POSTPROCESSING

(b) AFTER POSTPROCESSING, WITH $\eta_0 = 5$

Figure 4.— Forest classification maps before and after postprocessing Sam Houston National Forest 2X data.

(c) AFTER POSTPROCESSING, WITH $\eta_0 = 10$

(d) AFTER POSTPROCESSING, WITH $\eta_0 = 40$

Figure 4.— Concluded.

Figure 5.— Forest classification maps before and after postprocessing Sam Houston National Forest 3X data.
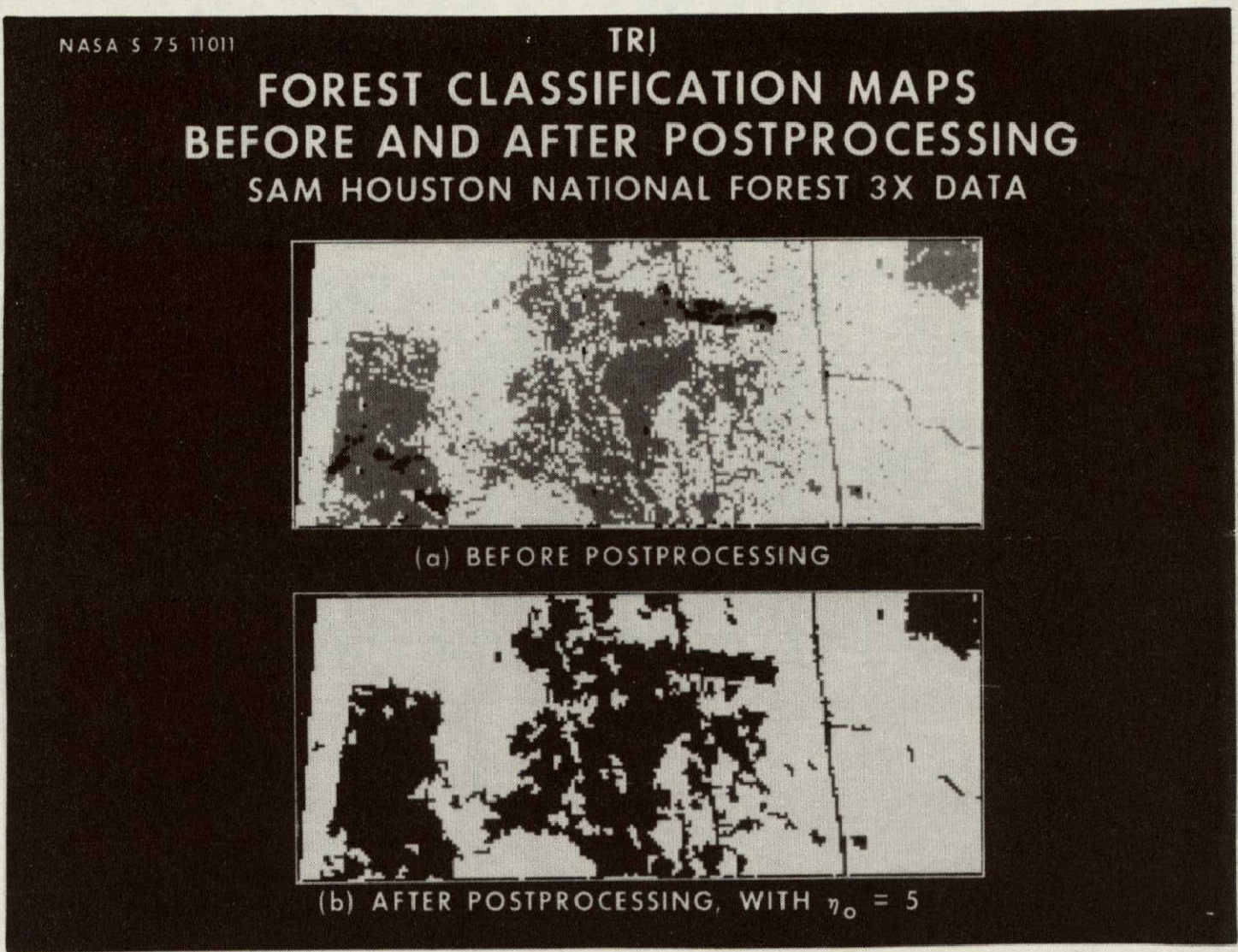
NASA S 75 11012
TRI
**FOREST CLASSIFICATION MAPS**
**BEFORE AND AFTER POSTPROCESSING**
SAM HOUSTON NATIONAL FOREST 3X DATA (CONT)
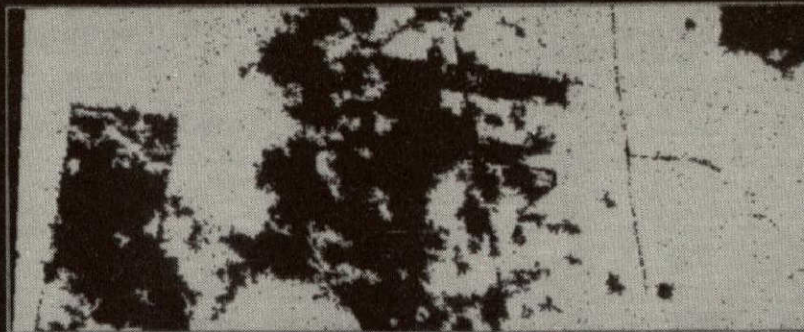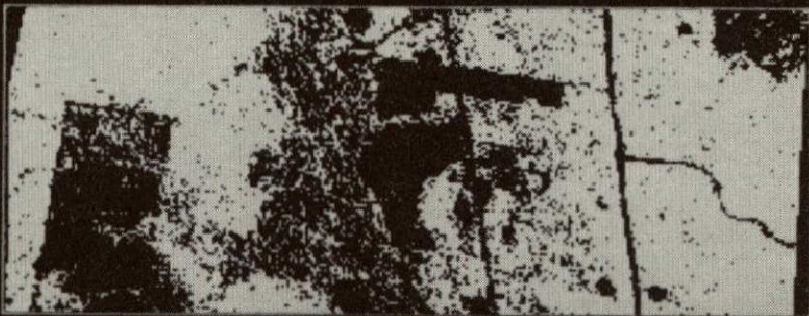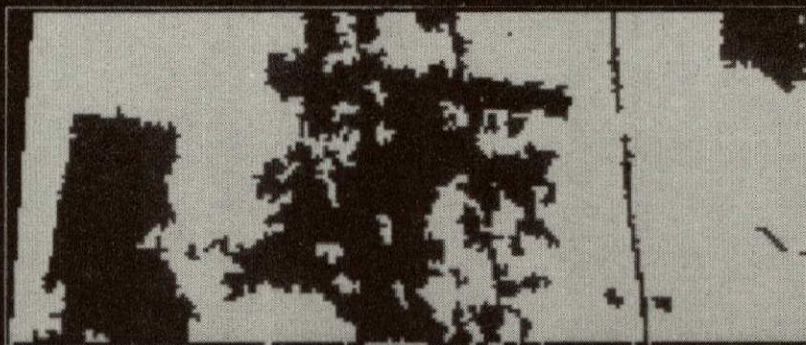
(c) AFTER POSTPROCESSING, WITH $\eta_0 = 10$
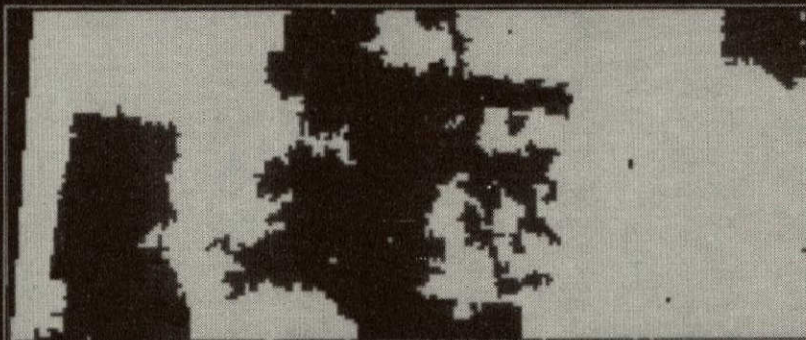
(d) AFTER POSTPROCESSING, WITH $\eta_0 = 40$

Figure 5.— Concluded.

The algorithm was applied to the cluster classification map for 1X data for $n_0$ values of 5, 10, and 40. Connected sets smaller than $n_0$ were absorbed by the surrounding sets. The enhanced maps are in figures 3(b), 3(c), and 3(d). Similarly, figures 4(b), 4(c), 4(d), 5(b), 5(c), and 5(d) show the enhanced maps of 2X and 3X data for $n_0$ values of 5, 10, and 40.

## 4.3    RESULTS

Figures 3 through 5 show how the new image enhancement algorithm can remove the holes in classification images while preserving significant boundary information. For the 1X data at a (8 meter)$^2$ spatial resolution, postprocessing with $n_0 = 40$ produced the best result. The same $n_0$ value produced the same result for 2X data. For the 3X data, postprocessing with $n_0 = 10$ provided the best result because $n_0 = 40$ produced a map in which some significant features were altered. These results show the possible compromise between data spatial resolution, classification accuracies, and optimal postprocessing with the best $n_0$ value. Indeed, another implication from the results was the possible advantage to timber stand classification by using lower resolution data (i.e., data having large on-ground pixels); such a conclusion had been drawn from previous works (ref. 14).

The new image enhancement algorithm removed from the original 1X data the variously shaped, connected holes smaller than the $n_0$ values. Linear features were retained except in figure 5(d), in which $n_0 = 40$ was too large for data at that spatial resolution. The curvilinear shape of

the boundaries was preserved, not smeared. These set-removal, shape-preservation properties are needed in timber stand mapping. These properties are weak or absent in other postprocessing techniques, such as neighbor-checking. Section 5.0 demonstrates via examples the superiority of the new image enhancement technique, in regards to such properties which are required in timber stand mapping.

## 5.0   THE NEW IMAGE ENHANCEMENT ALGORITHM AND
## NEIGHBOR-CHECKING PROCEDURES

### 5.1   THE NEIGHBOR-CHECKING ALGORITHM

A neighbor-checking algorithm, such as RECLASS, post-processes a classification map by modifying the pixel labels when a majority of the surrounding pixels have different labels.   Sections 5.1 through 5.5 refer to the eight-neighbor-checking algorithm in which $m_x$ is the average of all eight neighboring labels surrounding $x$   (fig. 6) and $T$ is the input threshold in which $0.5 \leq T \leq 1$ .   The algorithm changes $x$ to 1 if and only if $T \leq m_x \leq 1$ .   The algorithm changes $x$ to 0 if and only if $0 \leq m_x < 1 - T$ .

Neighbor checking always uses the label values of the map before relabeling, which retains the old values until the checking is complete.

For $T = 0.5$ ,   the center pixels $x_1$, $x_2$, $x_3$, $x_4$, and $x_5$ ,   respectively, in figures 6(a), 6(b), 6(c), 6(d), and 6(e) will be changed to 1's by this neighbor-checking algorithm.   Similarly, $x_6$, $x_7$, $x_8$, and $x_9$ ,   respectively, in figures 6(f), 6(g), 6(h), and 6(i) will be changed to 0's.   The modification is irrespective of the original values of the center pixels.

### 5.2   SET REMOVAL

The new image enhancement algorithm can remove unwanted isolated sets, i.e., isolated areas (fig. 7), more easily than the neighbor-checking algorithm.   Isolated areas in

| 1 | 1 | 1 |
|---|---|---|
| 1 | $x_1$ | 1 |
| 1 | 1 | 1 |

(a) $x_1$

| 0 | 1 | 1 |
|---|---|---|
| 1 | $x_2$ | 1 |
| 1 | 1 | 1 |

(b) $x_2$

| 0 | 0 | 1 |
|---|---|---|
| 1 | $x_3$ | 1 |
| 1 | 1 | 1 |

(c) $x_3$

| 0 | 0 | 0 |
|---|---|---|
| 1 | $x_4$ | 1 |
| 1 | 1 | 1 |

(d) $x_4$

| 0 | 0 | 0 |
|---|---|---|
| 0 | $x_5$ | 1 |
| 1 | 1 | 1 |

(e) $x_5$

| 0 | 0 | 0 |
|---|---|---|
| 0 | $x_6$ | 0 |
| 1 | 1 | 1 |

(f) $x_6$

| 0 | 0 | 0 |
|---|---|---|
| 0 | $x_7$ | 0 |
| 0 | 1 | 1 |

(g) $x_7$

| 0 | 0 | 0 |
|---|---|---|
| 0 | $x_8$ | 0 |
| 0 | 0 | 1 |

(h) $x_8$

| 0 | 0 | 0 |
|---|---|---|
| 0 | $x_9$ | 0 |
| 0 | 0 | 0 |

(i) $x_9$

Figure 6.— The nine possible configurations. With $T = 0.5$ , the neighbor-checking algorithm changes $x_1$, $x_2$, $x_3$, $x_4$, and $x_5$ pixels to 1. The $x_6$, $x_7$, $x_8$, and $x_9$ pixels become 0.

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

Figure 7.— An isolated set of 1's removable by the new algorithm with $n_0 \geq 13$ but unremovable by the neighbor-checking algorithm with $0.5 \leq T \leq 1$ .

timber classification maps must be removed when the area size is smaller than $n_0$ .

The isolated set of 1's in figure 7 is removable by the new algorithm if $n_0 \geq 13$ . However, using $0.5 \leq T \leq 1$ in the neighbor-checking algorithm, the eight neighbors of each 1 have the mean value $m_x \geq 0.5$ . The neighbor-checking algorithm can remove none of the 1's in the figure.

An "unremovable set" is a connected set which cannot be eliminated by label modification by applying a given algorithm a finite number of times. The isolated set in figure 7 is removable by the new algorithm but unremovable by the neighbor-checking algorithm. This fact holds for any connected set containing the isolated set in figure 7.

In the smaller isolated set in figure 8(a) the 1's are removable by the new algorithm with $n_0 \geq 6$ . The first use of the neighbor-checking algorithm with $T = 0.5$ reduces the set size but fails to remove the set. See figure 8(b). The second use of the neighbor-checking algorithm removes the set.

Therefore, the isolated set in figure 8(a) is removable by the new algorithm with $n_0 \geq 6$ . The set is removable by the neighbor-checking algorithm on the second try.

In conclusion, isolated data sets as in figures 7 and 8 must be removed for timber mapping. The new image enhancement algorithm is more powerful in removing isolated sets than the neighbor-checking procedure.

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

(a)   Original isolated set

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

(b)   Modified image after one use of the algorithm

Figure 8.— An isolated set of 1's removable by two uses of
the neighbor-checking algorithm with  T = 0.5 ;   the
resulting modified image after two uses contains all 0's
and is not shown here.  This set of 1's is removable by
the new image enhancement algorithm with  $n_0 \geq 6$ .

## 5.3  LINEAR FEATURES

The new image enhancement algorithm preserves linear features such as roads and pipelines; but the neighbor-checking algorithm eliminates them in most cases.  Linear features often form administrative boundaries in timber stand maps.  The long, narrow appearance of these features is not just useful but sometimes mandatory, and a good post-processing image enhancement technique should preserve them. The data sets in this report all have these features.

For example, figure 9 shows two linear features represented by 1's.  The new image enhancement algorithm with $n_0 \leq 8$  preserves the two linear features.  However, the neighbor-checking technique with  $0.5 \leq T \leq 1$  eliminates the linear features completely.

## 5.4  CURVILINEAR BOUNDARIES

The new image enhancement algorithm preserves curvilinear boundaries, but the neighbor-checking algorithm does not.  Curvilinear features, as in figures 10(a) and 11(a), often appear in timber stand maps.  Preservation of these features during postprocessing is essential.

With the appropriate size of  $n_0$ ,  the new algorithm preserves the curved boundary between the 0's and 1's on figures 10(a) and 11(a).  The size of  $n_0$  depends on whether figures 10(a) and 11(a) are parts of larger images or stand-alone images.

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

Figure 9.— Linear features shown by two sets of 1's. The new algorithm with $n_0 \leq 8$ preserves the features, but the neighbor-checking algorithm with $0.5 \leq T \leq 1$ removes them.

28

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

(a)   Original image

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

(b)   First use

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

(c)   Second use

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

(d)   Third use

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

(e)   Fourth use

Figure 10.— The change in the curvilinear boundary between
the 0's and 1's after uses of the neighbor-checking
algorithm with   T > 0.5 .

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

(a)   Original image

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

(b)   First use

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

(c)   Second use

Figure 11.— The change in the curvilinear boundary between the 0's and 1's after uses of the neighbor-checking algorithm with $T \geq 0.5$ .

However, the first use of the neighbor-checking algorithm with  T > 0.5  on figure 10(a) changes the boundary to that of figure 10(b).  Three successive applications reduce the image to that in figures 10(c), 10(d), and 10(e).  The protruded portion of 1's eventually becomes smeared and removed, as in figure 10(e).  Similarly, two uses of the neighbor-checking algorithm with  T ≥ 0.5  on figure 11(a) remove the protruded portion of 1's, as shown in figures 11(b) and 11(c).

## 5.5  STRUCTURAL PATTERN RECOGNITION

In structural pattern recognition, the new image enhancement algorithm treats classification maps on a higher level than the localized neighbor-checking procedure.  Structural pattern recognition is the algebraic or syntactic approach (ref. 15).

The new image enhancement algorithm decides on modification or nonmodification until a connected set is detected.  However, the neighbor-checking procedure localizes its decision based on the labels of pixels surrounding the center pixel.  Both the high-level and low-level treatments of the structural approach have advantages and disadvantages, the discussion of which lies beyond the scope of this report.

# 6.0 CONCLUSIONS

The report shows the theory and applications of a new image enhancement algorithm which postprocesses and refines computer classification maps of multispectral data. The development of the procedure resulted from the need to remove holes in the data to obtain more useful timber stand classification maps by computer processing. The new algorithm was applied to three data sets collected over the Sam Houston National Forest at three different spatial data resolutions.

The results show that the new algorithm produced cleaner classification maps in which holes of small predesignated sizes were eliminated and significant boundary information was preserved. These cleaner postprocessed maps better resemble true-life timber stand maps and are thus more usable products than the pre-postprocessed ones. Compared to an accepted neighbor-checking postprocessing technique, the new algorithm is indeed more appropriate for timber stand mapping.

More work remains to be done, as in the relationship between spatial data resolution and classification accuracies, optimal usage of the procedure, and optimal parameters used in the algorithm. Variations of the new image enhancement should be applied to more, varied data sets. The computer time required to perform the algorithm should be compared to the time required by other algorithms. The overall effect of treating multiclass images as binary images of 0's and 1's should be examined as to the modification of connected sets during step-by-step applications of the algorithm to individual features.

## 7.0 · SUMMARY

The theory and applications are presented of a new image enhancement algorithm which refines computer classification maps of multispectral data. The refinement eliminates connected sets smaller than a prespecified size and merges them to the surrounding area. Such a postprocessing is motivated by and accomplishes the requirement of a forestry application of timber stand mapping, where small areas are conventionally required to be absorbed into surrounding large areas to form "homogeneous" stands. The computer algorithm was developed, programmed, and applied successfully to three multispectral data sets collected over Sam Houston National Forest at three different spatial data resolutions. A comparison against an accepted neighbor-checking post-processing technique reinforced the conclusion on the appropriateness of the new procedure for timber stand mapping.

This work was motivated by the necessity of cleaning "holes" in computer generated classification maps of forestry multispectral data. An area that is conventionally called a stand of forest often fails to be entirely classified by computers as forest due to the nonhomogeneity within the stand, or due to openings between tree crowns, or even due to poor classification designs. The problem of holes is further compounded by the U.S. Forest Service definition of a stand type. A stand type refers to an area larger than 4 hectares in which more than 51 percent of the trees in the canopy belongs to one tree species or group of species. A stand smaller than 4 hectares should be absorbed into and classified the same as the surrounding under the present system of stand type mapping. Because existing statistical classification techniques have not been designed to clean

up these small areas, computer generated classification maps
would contain undesirable holes which should not exist on
final stand type maps.

A new computer algorithm was developed and programmed
to solve the above problem with the idea of searching for
connected sets, determining their sizes, and eliminating
those small sets. The minimum size, $n_o$ , of a connected set
not to be eliminated can be specified by the user. A set of
pixels is connected if every pixel is a left-right or top-
bottom neighbor of at least one other pixel·in the set; the
set of one pixel is also connected. Diagonal connectivity
can be similarly examined but not programmed for the present
study. This idea of connectivity was studied by past
researchers in the pattern recognition of handwritten
characters. However, the present variation of the connec-
tivity concept and its application to timber stand type
mapping are novel.

Classification images considered in the study were binary
images, in which each pixel has the label value 0 or 1. This
two-class classification image is a general representation,
because any map can be examined one feature at a time.
Modification of the computer program to multi-class images
is relatively simple. A typical classification image is one
derived from processing multispectral scanner data through
techniques such as the training-field, maximum likelihood
classifier.

The new image enhancement algorithm was applied to post-
process classification maps of the Sam Houston National Forest
of Texas. The three data sets used were 1X, 2X, and 3X,

having spatial resolution of $(8 \text{ meters})^2$, $(16 \text{ meters})^2$, and $(24 \text{ meters})^2$, respectively. 2X and 3X were electronically reduced from 1X by pixel-averaging.

Results of applying the algorithm showed that cleaner classification maps were produced, in which holes of small predesignated sizes were eliminated and significant boundary information was preserved. These cleaner postprocessed maps better resemble true-life timber stand maps and are thus more usable products than the pre-postprocessed ones.

Comparison was made via examples to demonstrate the appropriateness of the new algorithm against another widely accepted postprocessing algorithm, the neighbor-checking algorithm. Superiority of the new algorithm was demonstrated in regards to the properties of set-removal, shape-preservation of linear features and of curvilinear boundaries. These properties are required in timber stand mapping. It was shown that the new algorithm can remove unwanted isolated sets more easily. Important linear features in timber type maps such as those due to roads and pipelines, are preservable by the new algorithm while usually eliminated by the neighbor-checking algorithm. Curvilinear boundaries are preserved by the new algorithm instead of being smeared and distorted by the other procedure.

In conclusion, the new image enhancement procedure is useful in post-processing computer generated classification maps and producing more usable products that better resemble true-life timber stand maps. More work remains to be done such as the relationship between spatial data resolution and classification accuracies, and optimal use of the algorithm.

# 8.0 REFERENCES

1. Heller, R. C.; et al.: Evaluation of ERTS-1 Data for Inventory of Forest and Rangeland and Detection of Forest Stress, Final Report. (Pacific Southwest and Rocky Mountains Forest and Range Experimental Stations, Forest Service, U.S. Department of Agriculture; NASA Contract S-70251-AG.) Aug. 1974.

2. Remote Multispectral Sensing in Agriculture. Bull. 873, Laboratory for Applications of Remote Sensing, Purdue University (Lafayette, Indiana), vol. IV, Dec. 1970.

3. Erb, R. B.: A Compendium of Analysis Results of the Utility of ERTS-1 Data for Land Resources Management. NASA TM X-58156, 1974.

4. Phillips, T. L., ed.: LARSYS Version 3, User's Manual. Laboratory for Applications of Remote Sensing, Purdue University (Lafayette, Indiana), June 1973.

5. Colwell, R. N.: An Integrated Study of Earth Resources in the State of California Based on ERTS-1 and Supporting Aircraft Data, Final Report. (Forestry Remote Sensing Laboratory, University of California, Berkeley; NASA Contract S-21827.) July 1973.

6. Binford, T. O.; and Tenenbaum, J. M.: Computer Vision. Transactions on Computers, Institute of Electrical and Electronics Engineers. May 1973, pp. 19-24.

7. Hueckel, M. H.: An Operator Which Locates Edges in Digitized Pictures. J. Assoc. Computing Machinery, vol. 18, no. 1, Jan. 1971, pp. 113-125.

8. Rosenfeld, A.: Connectivity in Digital Pictures. J. Assoc. Computing Machinery, vol. 17, no. 1, Jan. 1970, pp. 146-160.

9. Rosenfeld, A.; and Pfaltz, J. L.: Sequential Operations in Digital Picture Processing. J. Assoc. Computing Machinery, vol. 13, no. 4, Oct. 1966, pp. 471-494.

10. Stephanelli, R.; and Rosenfeld, A.: Some Parallel Thinning Algorithms for Digital Pictures. J. Assoc. Computing Machinery, vol. 18, no. 2, Apr. 1971, pp. 255-264.

38

11. Unger, S. H.: Pattern Detection and Recognition. Proceedings of the Institute of Radio Engineers, vol. 46, Oct. 1958, pp. 1744-1750.

12. Forestry Applications Exploratory Study Project: A Preliminary Plan. JSC-09420, Sept. 1974.

13. Kan, E. P.: The Forestry Applications Project on Timber Resources, FY75 Interim Report. JSC-09410, Dec. 1974.

14. Kan, E. P.; et al.: Data Resolution Versus Forestry Classification and Modeling. Proceedings of the Second Symposium on Machine Processing of Remotely Sensed Data, Purdue University (Lafayette, Indiana). June 3-5, 1975.

15. Kulikowski, J.: Algebraic Methods in Pattern Recognition. Courses and Lectures No. 85. Springer-Verlag Publishers (New York City), ISBN 0-387-81128-1, International Center for Mechanical Sciences, 1972..

APPENDIX

THE SIX STEPS OF THE NEW ALGORITHM

## APPENDIX

## THE SIX STEPS OF THE NEW ALGORITHM

The six steps of the new image enhancement algorithm are initialization, tagging, clustering, counting, label modification, and reiteration. The notations are

1.  Data array $A(I,J)$: $I = 1, \cdots, n_0$ ; $J = 1, \cdots, N$ .

2.  Tag array $S(I,K,L)$: $I = 1, \cdots, n_0$ ; $K = 1, \cdots, N$ ;
    $L = 1,2,3$ .

3.  Count array $C(P,Q)$: $P = 1, \cdots, N$ ; $Q = 1, \cdots, 2n_0+3$ .

4.  Input data image $D(I,J)$: $I = 1, \cdots, M$ ; $J = 1, \cdots, N$ .

The integer arrays take values of 0 and 1. The symbol ":=," as in $X := Y$ , denotes the placement of the value of $Y$ into $X$ .

## A.1 INITIALIZATION

The initialization steps follow.

1.  Initialize $A(I,J)$, $S(I,K,L)$, and $C(P,Q)$ .

2.  $A(I,J) := D(I,J)$ . $J = 1, \cdots, N$ ; $I = 1, \cdots, n_0$ .

3.  $LL := 1$ ; $tag := 0$ .

4.  Do the steps in section A.2 for $I = 1, \cdots, n_0$ .

## A.2 TAGGING

Along-the-line connected sets are found and tagged.

1.  $K := 1$ .

2.  Along the $I$th line of $A(I,J)$, find the $K$th connected set. Then $tag := tag + 1$ .

3. $S(I,K,1)$ := the $J$ coordinate along the $I$th line of the first element of the $K$th connected set in $A(I,J)$ .

4. $S(I,K,2)$ := the $J$ coordinate along the $I$th line of the last element of the $K$th connected set in $A(I,J)$ .

5. $S(I,K,3)$ := tag ; $K$ := $K + 1$ .

6. Repeat steps 2 through 5 until all connected sets on the $I$th line are tagged.

## A.3 CLUSTERING

Across-line connected sets are identified and tagged uniquely.

1. $IL$ := $LL + 1$ .

2. $PP$ := $1$ ; $R1 = mod(IL - 2, n_0) + 1$ ; $R2 = mod(IL - 1, n_0) + 1$ .

3. Test these conditions:

    a. $S(R2,PP,1) \leq S(R1,K,2)$ .

    b. $S(R2,PP,2) \geq S(R1,K,1)$ .

    c. $A[R1,S(R1,K,1)] = A[R2,S(R2,PP,1)]$ .

4. If the conditions in step 3 hold,

$$S(I,K,3) := \min[S(R1,K,3),S(R2,PP,3)] \qquad I = R1,\cdots,R2$$

5. Check each tag of all connected sets along the R1st and R2nd lines. If the tag equals $\max[S(R1,K,3),S(R2,PP,3)]$ , change it to $\min[S(R1,K,3),S(R2,PP,3)]$ .

6. Repeat steps 3 and 4 for $K = 1,2,\cdots,N$ .

7. $PP$ := $PP + 1$ . Repeat steps 3 through 6 until $PP = N$ .

8. If $IL = n_0$ , go to step 10.

9. $IL$ := $IL + 1$ . Go to step 2.

10. IL := IL - 1 . Repeat steps 2 through 7 and 10 until IL = 2 .

## A.4 COUNTING

The pixels of clustered, tagged sets are counted and stored in $C(P,Q)$.

1. $I := \text{mod}(LL - 1, n_0) + 1$ .

2. Compute $Q_K$ and $R_K$ for all $K$ by the equation $S(I,K,3)/N = Q_K \cdot N + R_K$ .

3. If $C(R_K,1) = 0$ , $C(R_K,2) := Q_K$ . Put $S(I,K,1)$ and $S(I,K,2)$ in the two next available words in the $R_K th$ counter. The address of this next available word is stored in $C(R_K,3)$ .

4. If $0 < C(R_K,1) < n_0$ and if $Q_K = C(R_K,2)$ , put $S(I,K,1)$ and $S(I,K,2)$ in the two next available words in the $R_K th$ counter.

5. $I := \text{mod}(I + 1, n_0)$ .

6. Repeat steps 2 through 6 the number of times shown by $n_0 - 1$ .

## A.5 LABEL MODIFICATION

The tags of clustered, tagged sets smaller than $n_0$ are modified.

1. $I := \text{mod}(LL - 1, n_0) + 1$ .

2. In checking through the counters $P = 1, \cdots, N$ , if $0 < C(P,1) < n_0$ , change the values in $A(I,J)$ for the $J$ locations $C(P,4)$ and $C(P,5)$ , between $C(P,6)$ and

C(P,7), and so forth. The change is $A(I,J) := 0$ if $A(I,J) = 1$ ; $A(I,J) := 1$ if $A(I,J) = 0$ .

3. Output the I*th* line of A(I,J) into the final post-processed image.

## A.6  REITERATION

Input a fresh line of data from D(I,J) into A(I,J) until all lines are exhausted.

1. LL := LL + 1 if all lines are not exhausted. Stop if they are exhausted.

2. $R := LL + n_0$ .

3. $I := \mod(R - 1, n_0) + 1$ .

4. $A(I,J) := D(R,J)$ . $J = 1, \cdots, N$ .

5. Go to the steps in section A.2.