NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Technical Memorandum 33-774

On the Problem of Embedding Picture Elements in Regions

(NASA-CR-148279) ON THE PROBLEM OF EMBEDDING PICTURE ELEMENTS IN REGIONS (Jet Propulsion Lab.) 35 p HC $4.00 CSCL 14E

N76-26445

Unclas

G3/35 42349

JET PROPULSION LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
PASADENA, CALIFORNIA

June 1, 1976

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Technical Memorandum 33-774

On the Problem of Embedding Picture Elements in Regions

Y. Yakimovsky
R. Cunningham

JET PROPULSION LABORATORY
CALIFORNIA INSTITUTE OF TECHNOLOGY
PASADENA, CALIFORNIA

June 1, 1976

PREFACE

The work described in this report was performed by the Space Sciences Division of the Jet Propulsion Laboratory.

CONTENTS

FIGURES

PRECEDING PAGE BLANK NOT FILMED

ABSTRACT

A few new algorithms for region growing in pictures were developed. These algorithms are a step toward finding a satisfactory solution to the image segmentation problem, and in-depth understanding of the problems of nonsemantic image segmentation. The algorithms utilize a sequential decision approach for region boundaries detection. The sequential decisions are supported by a stochastic algorithm that maintains local statistics of the region near the boundaries as the region grows. A few illustrations of the algorithm's performance are included.

I. INTRODUCTION

The information associated with each picture element (pixel) is the integral of the light energy reaching the corresponding sensor element. Typically, this light is reflected from or originated on a small area on the surface of a three-dimensional (3-D) body that is geometrically projected through the camera lens system on the corresponding sensor element. The total surface of a 3-Day body may be projected on a few pixels. Our objective is to group (cluster) all the pixels upon which the surface of one 3-D body is projected into a region. This region will be 2-D domain on the picture. Clearly, the extent of one 3-D body is context and task dependent (these define the desired resolution). Many algorithms that use world model (semantics) to find regions have been developed (Refs. 1, 2, 3, and 4). However, the emphasis here is on the development of general purpose problem independent segmentation algorithms that will be in a bag of tricks (feature extractors) of a higher level pattern recognition or semantic system like those in Refs. 5 and 6, which will use them as feature extractors for the specific problem domain.

This work is one in a long sequence of works on clustering and region growing. The region growing problem is basic for pictorial pattern recognition and data reduction; a few of the typical applications are given in Refs. 2, 3, 6, and 7. Few nonsemantic region growing algorithms are given in Refs. 3, 6, 8, 9, and 10. The algorithms presented below are new in that they integrate a sequential decision approach with stochastic maintenance of the local statistics on the boundaries of the growing region. Statistical likelihood ratio type tests were used before (Ref. 8), but never with sequential decision. Reference 8 gives a broader survey of related works on region growing.

The implemented system described below is another phase in upgrading automatic segmentation in the way of speed, computer resources requirements, and reliability. Unfortunately, well defined and usable quality measures for the performance of segmentation or clustering algorithms is lacking in the literature. For that reason, the algorithm's performances are demonstrated on few sample pictures. We hope that this illustration coupled with the theory will substantiate and justify our claims for improvement.

The mathematical-statistical justification of the algorithm is commingled with the description of the decision algorithms. That way we hope the theory presentation will be tied organically to the algorithmical implementation.

II. THE HARDWARE SYSTEM

A diagram of the minicomputer-based image analysis system, on which the algorithms are implemented, is given in Fig. 1. RAPID, which is a Random Access Picture Digitizer display and memory system, is described in detail in Ref. 11. Basically, RAPID stores a digitized video image into a 256 X 256 (8-bit) byte memory array internal to RAPID.

The real time computer can randomly access any byte of the digitized image in about 5 μs by specifying an 8-bit line address and an 8-bit column address. The content of a byte is referred to in this publication as G(i, j) where $0 \le i \le 188$ is a column number, $0 \le j \le 244$ a line number, and $0 \le G(i, j) \le 255$ is the actual byte content (the digitized video level).

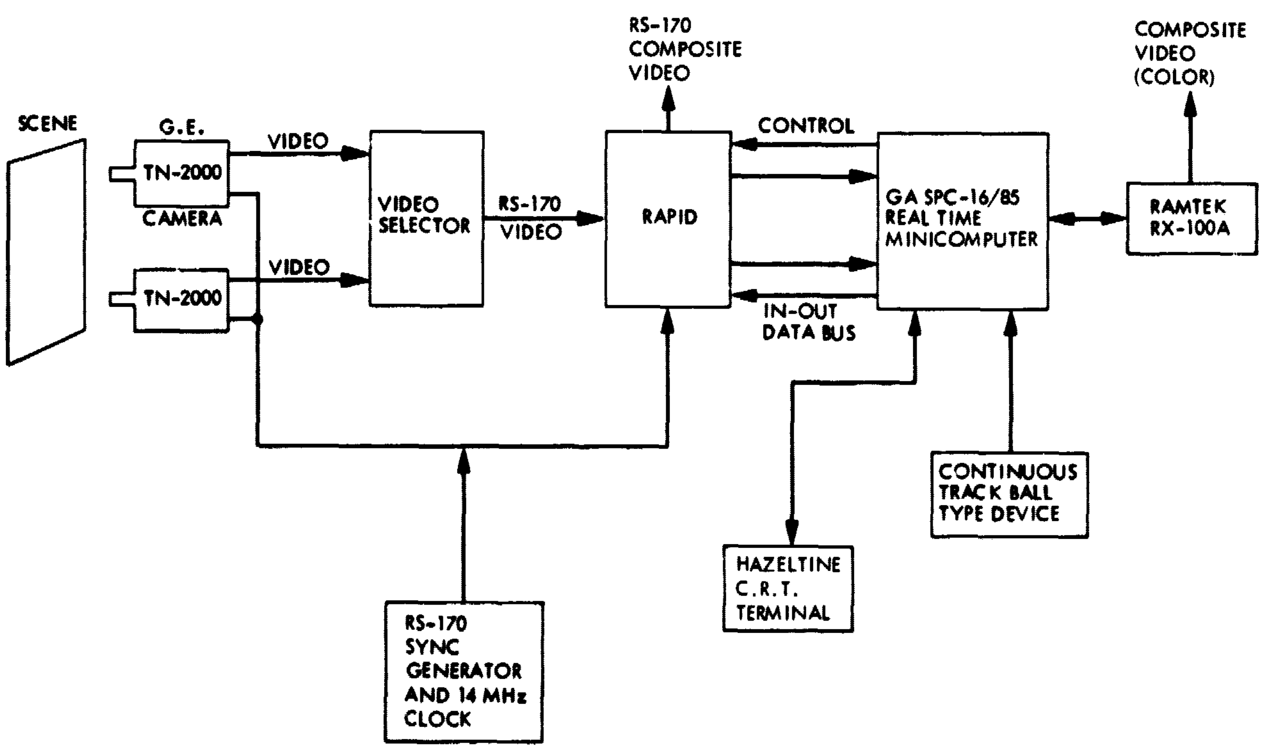


Fig. 1. Minicomputer-based image analysis system, block diagram

The cameras used in the system (Fig. 2) are pure solid state (Charge Injection Device) cameras, G.E. TN-2000, which substantially meet their specified noise level of about 0.5%. These cameras are equipped with automatic iris lenses that maintain the average video signal on the whole image nearly constant. The sensing plate is composed of 188 × 244 sensing elements. The video signal is digitized so that RAPID's memory element value G(i, j) will be proportional to the light energy falling on sensing element (i, j) of the plate.

The video output of RAPID contains current image of the content of its memory. The content of the memory can be refreshed continuously from the camera's video signal or remain in a static condition. In either case, the computer can randomly read or write a picture element in 5 μs, as was mentioned above.

In addition to the image display capabilities provided by RAPID, a RAMTEK RX-100A is interfaced to the minicomputer, and it provides gray level and color display, and graphic capabilities. Eventually, we plan to replace the RAMTEK display unit with another RAPID-like unit.

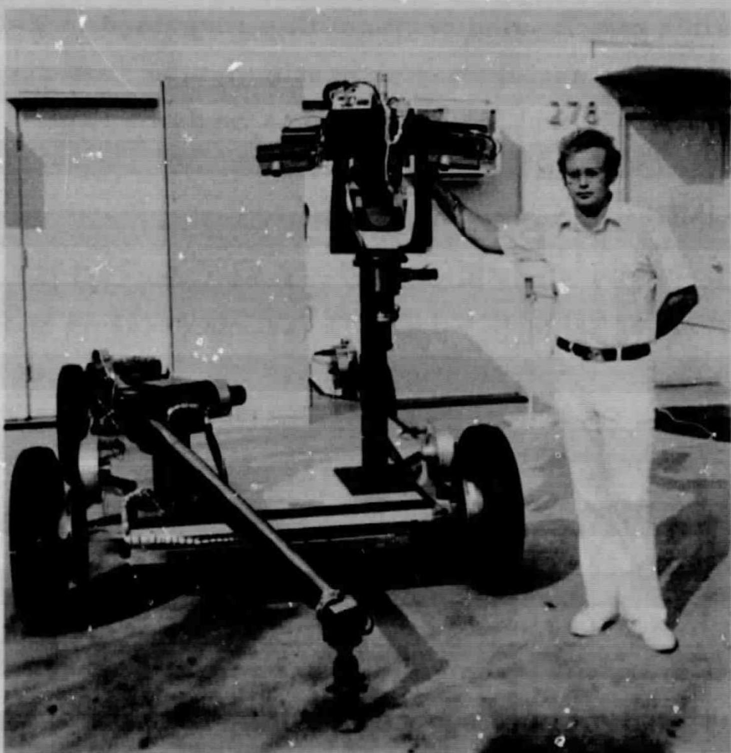


Fig. 2. Charge Injection Device camera on the robot vehicle

III. INITIAL REGION GROWING TYPES

Two approaches to initial image segmentation were implemented in our system. The first is called the "scanning" region grower; its control structure is described in Ref. 8. It scans the whole image from the top left corner line by line down to the bottom right corner and outlines all regions. The second type of region growing is called the "blooming" region grower. It starts from an arbitrarily specified point somewhere on the image and grows a region around the initial point (Ref. 12). An application of the sequential sampling mechanism (to be defined shortly) to the scanning region growing approach is given in Section IV, which is identical to Ref. 10. However, the bulk of this paper describes region growers of the blooming type. We found the reliability of the blooming region growers to be superior to that of the scanning region growers. That is due mainly to the fact that concentrating on one region allows us to keep more relevant information on the region being grown, and hence to achieve more reliable decisions. Also, the multiple applications of the blooming region grower result in a full segmentation, which also allows for overlapping regions (see Fig. 3). The need for nondeterministic segmentations (and possible overlapping regions) was suggested in Refs. 13 and 14, and we believe it will be usable by a semantic picture analysis system of the type described in Ref. 5, and for second-pass analysis (Ref. 8).

A. THE BLOOMING REGION GROWERS

The input for these kinds of region growers are: (1) the image matrix $G(i, j)$, (2) the starting point (i_0, j_0), (3) various decision confidence thresholds which define the resolution sensitivity. The output of the region grower will include at least a Boolean matrix $IN(i, j)$. Here, $IN(i, j)$ = TRUE means that the algorithm found point (i, j) to belong to the region grown around (i_0, j_0), and $IN(i, j)$ = FALSE means that the algorithm found (i, j) to be outside that region.

The output may also include the Boolean array $OUT(i, j)$. $OUT(i, j)$ = TRUE means that there were some tests that suggested that (i, j) should not be included in the region grown around (i_0, j_0). Because of the statistical nature of the decision-making process, there can be points for which $OUT(i, j)$ is TRUE, but later they were added to the region; e.g., $IN(i, j)$ is

i \ j	1	2	3	4	5
1					
2		reg 1	reg 1	reg 1	
3		reg 1 reg 2	reg 2	reg 1	
4		reg 2	reg 2		

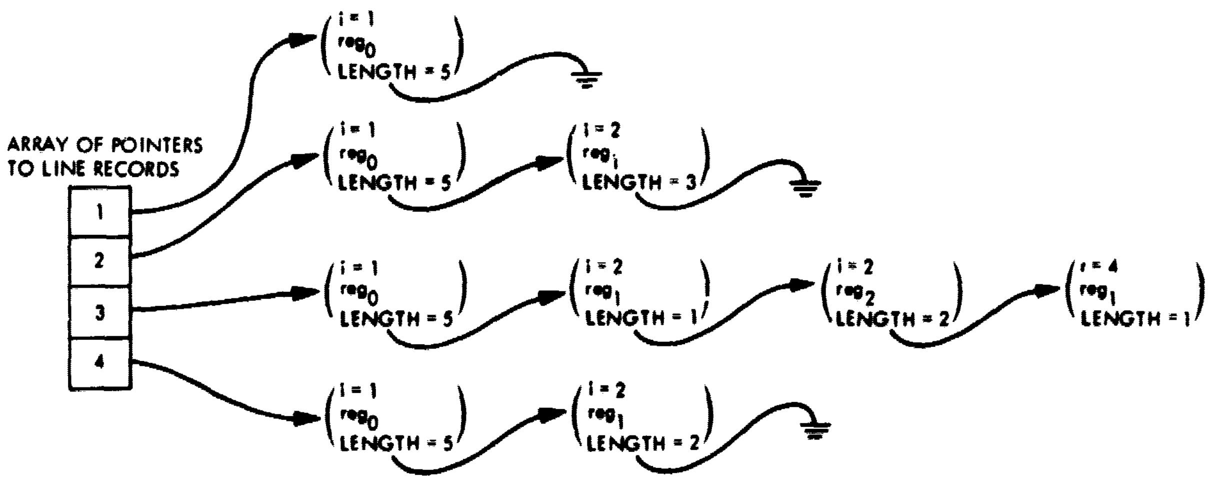
REGION ASSIGNMENT FOR POINTS

i \ j	1	2	3	4	5
1	98	99	80	83	64
2	100	28	24	20	85
3	102	32	36	20	86
4	98	34	36	92	88

G(i, j) VALUES

	SIZE	SUM G(i, j)	SUM i	MIN i	SUM $G(i, j)^2$	
reg_0	20					1
reg_1	5	138	15	2		2
reg_2	4	124			3476	3

reg_0 ALWAYS STANDS FOR THE WHOLE PICTURE FRAME



A PARTIAL DESCRIPTION OF REGION RECORDS AND MATRIX DATA STRUCTURE

Fig. 3. A partial description of region records and sparse matrix data structure

also TRUE. We anticipate that the OUT array will be used for error recovery in future systems.

The raw output in the Boolean array IN(i, j) is incorporated by a simple procedure into a sparse matrix form; this procedure also generates an appropriate region record. The region record includes: (1) information on the region growing parameters — e.g., starting point and thresholds, (2) statistics of the region sufficient to compute easily the mean and variance, and the first few moments of the 2-D shape and the gray level structure, and (3) the enclosing rectangle (the maximum and the minimum of the i and j values over the region).

The sparse matrix has for each horizontal line that intersects the region a record describing the segment(s) of the region on that line (see Fig. 3). The sparse matrix data structure allows integration of the output from few calls on the blooming region grower; hence it allows analysis of the whole image. Note that the segment may overlap; hence, the sparse matrix organization provides for overlapping regions. The sparse matrix representation is more compact than the Boolean array structure, but is somewhat less efficient in core and access than the chain encoding representation we used in Ref. 8. But it provides for more flexibility in the regions' organization. The balance was in favor of sparse matrix representation for our purposes.

B. THE BLOOMING REGION GROWERS' DRIVER

The description of the algorithms is broken into two parts. First we describe the driver, and later we describe statistical data maintenance and decision. This is done because there are six combinations of decision procedures, all of them fitting the same driver.

C. THE DRIVING ALGORITHM

As was mentioned earlier, we assume that G(I, J) is available, and that (I_0, J_0), the starting point, is specified.

1. <u>Initialization</u>

<u>I1</u>: Set IN(i, j) and OUT(i, j) to FALSE everywhere.

<u>I2</u>: IN(I_0, J_0) = TRUE

I3: Generate a new region record for a region containing only (I_0, J_0)

I4: Collect the statistic of the initial point (I_0, J_0). The statistic collected depends on the decision procedure and will be described later.

I5: Generate a list of candidate point records. A record in that list specifies an image point still outside the region (its IN value is FALSE) that is adjacent to the region, and is a candidate to be added to the growing region. In the current phase, the candidate point records list contains at most 4 records for the points $(I_0 - 1, J_0)$, $(I_0 + 1, J_0)$ $(I_0, J_0 + 1)$ and $(I_0, J_0 - 1)$. If any of these points is outside the digitized video addressing range, no candidate record is generated for that point. The format of the candidate point record is dependent on the decision algorithm and will be described later (it may contain some statistical information on the gray value of points in the region near the candidate point).

2. The main loop

L1: If the candidate point records list is empty, the algorithm terminates.

L2: Remove the candidate record on top of the candidate list.

L3: Call the decision algorithm to test whether to add to the region the point specified in the candidate record just removed. If the decision is to add, continue in L5.

L4: Let (i, j) be the point in the candidate record that was removed, then set OUT(i, j) = TRUE, and continue to L1.

L5: Let (i, j) be the point in the candidate record that was removed, then set IN(i, j) = TRUE, and update the region record to include the new point.

L6: Test the four points (i - 1, j), (i + 1, j), (i, j - 1), (i, j + 1) (the four neighbors of (i, j), which was just added to the region), and decide whether to generate a candidate point record for any of those four points, and generate an appropriate record if one is suggested.

There are two decision options as to whether to generate a candidate record: (1) generate a candidate record if IN(.,.) of the suggested point is false and the candidate point is in the digitized video area, and (2) generate a candidate record for the suggested point only if both IN and OUT are false and the point is inside the digitized video frame. Only option (1) is used in the present system. If a new candidate record is generated, it is put at the end of the candidate point records list.

L7: Continue to L1.

D. THE DECISION PROCEDURES

There are two points where decisions are made and options for decision procedure are available: (1) the point where decisions are made to add candidate points to the region, (2) the point where decisions are made whether to generate a candidate record for a point adjacent to a point just added to the region (see L6 above). The first type of options affects the data structure of a candidate record, and it affects steps I4, I5, L3, and L6 of the driver algorithm of the blooming region grower.

1. Option 1

Three different decision algorithms were implemented to decide whether to add a point to the region. The first and the simplest one requires that the region record will include the running value of the maximum (MAXG) and the minimum (MING) of G(i, j) over points already inside the region. If (i, j) is the candidate point, it will be added to the region if both predicate

$$G(i, j) < MING + TR \qquad \text{and} \qquad G(i, j) > MAXG - TR$$

are satisfied, where TR is a decision threshold specified to the algorithm before it is called. This decision guarantees that always TR > MAXG - MING for the region. The candidate record for that option contains only the i and j coordinate of the candidate point. When a point is added to the region, MAXG and MING are updated, if necessary, in step L5.

2. Option 2

The second decision option is more complicated. It requires maintenance of a current estimate of the statistics of $G(i, j)$ values in the region near the point in the candidate record. The candidate record in this case contains the following information: (1) i, j coordinate of the candidate point, (2) running average $\overline{G}$ of $G(i, j)$, and (3) running average $\overline{G^2}$ of $G(i, j)^2$ (used to compute local variance).

In step I5, the initial statistics, e.g., the average of $G(i, j)$ and $G(i, j)^2$ around (I_0, J_0), are estimated. We are doing that by setting N equal to the set of all points (i, j) such that

$$N = \left\{ (i, j) \,\middle|\, \left| i - I_0 \right| < S, \left| j - J_0 \right| < S, \left| G(i, j) - G(I_0, J_0) \right| < R \right\}$$

where S (the window size) and R (the allowed gray value range) are two external parameters that control the algorithm. $\overline{G_0}$ and $\overline{G_0^2}$ are then set to:

$$\overline{G_0} = \frac{\sum_{(i, j) \in N} G(i, j)}{\sum_{(i, j) \in N} 1}$$

$$\overline{G_0^2} = \frac{\sum_{(i, j) \in N} G(i, j)^2}{\sum_{(i, j) \in N} 1}$$

We probably could have done about as well by initializing.

$$\overline{G_0} = G(I_0, J_0)$$

$$\overline{G_0^2} = G(I_0, J_0)^2 + N$$

where N is the typical variance of the noise of the cameras, which is about 1 in our case. In any case, the candidate point records list after step I5 will be:

$$\left(I_0 + 1,\ J_0,\ \overline{G_0},\ \overline{G_0^2}\right)$$

$$\left(I_0 - 1,\ J_0,\ \overline{G_0},\ \overline{G_0^2}\right)$$

$$\left(I_0,\ J_0 + 1,\ \overline{G_0},\ \overline{G_0^2}\right)$$

$$\left(I_0,\ J_0 - 1,\ \overline{G_0},\ \overline{G_0^2}\right)$$

assuming all four points are within the image frame. If in step L6 a new candidate record is generated, it will be done as follows. Assume $\left(i,\ j,\ \overline{G},\ \overline{G^2}\right)$ was the candidate record of the point just added to the region, and (k, l) is the point adjacent to (i, j), from which a new candidate record is generated. The new candidate record will be (k, l, G', G'') where:

$$G' = \frac{G(i,\ j) + W \cdot \overline{G}}{1 + W}$$

$$G'' = \frac{G(i,\ j)^2 + W \cdot \overline{G^2}}{1 + W}$$

W is the decay factor that is another parameter that controls the algorithm's performance. G' and G'' are a stochastic estimate of the average G(i, j) and $G(i,\ j)^2$ of points belonging to the region near the new candidate point. This computation requires minimal overhead, especially where $1 + W = 2^n\ n \geq 1$, in which case a simple shift operation will replace the division operation.

The information in a candidate record $\left(i_0,\ j_0,\ \overline{G},\ \overline{G^2}\right)$ is used in step L3 to decide whether to add the candidate point to the region. This is a sequential decision process. It is controlled by two arrays of decision thresholds $\left\{L_i\right\}$ and $\left\{H_i\right\}_{i = 1,\ \cdots,\ n}$ such that $H_i \geq L_i$ always and Hn = Ln where n is the maximum number of sample points for the sampler (to be defined later).

3. **Algorithm D (The Decision Algorithm)**

We assume the candidate point record is $\left(i_0, j_0, \overline{G}, \overline{G^2}\right)$.

D1: $VAR = \overline{G^2} - \overline{G}^2$.

D2: $D_0 = \left(G(i_0, j_0) - \overline{G}\right)^2$.

$D = D_0$.

D3: $D \geq H_0 \times VAR$; then immediately the decision is made not to include (i_0, j_0) in the region. Here $G(i_0, j_0)$ is too far from the average gray level of points of the region near (i_0, j_0), and the decision is made immediately. Note that the threshold H_0 is scaled by the local variance; hence, in some sense, the noise and texture of the region are compensated for in the decision. This is true for all decisions.

D4: If $D \leq L_0 \times VAR$ then the candidate point is immediately added to the regions.

D5: Initialize N and S; set $k = 0$.

Set $S = \{(i_0, j_0)\}$, $N = \{(i_0 - 1, j_0), (i_0 + 1, j_0), (i_0, j_0 - 1), (i_0, j_0 + 1)\}$, assuming all four points are in the image range.

Step D5 is reached where $L_0 < D_0/VAR < H_0$, and a sequential decision process has to be initiated. The set S of image points, which was already seen by the sampler while working on (i_0, j_0), is initialized to include one point: (i_0, j_0). From that stage on, samples will be taken from the set N which will contain all points adjacent to points in S and not in S (candidate points for the sampler). These samples are used to decide whether to add (i_0, j_0) to the region. N is initialized to include four points: $(i_0 - 1, j_C)$, $(i_0 + 1, j_0)$, $(i_0, j_0 + 1)$, and $(i_0, j_0 - 1)$.

D6: $k = k + 1$.

D7: Select a sample from N to be the k-th sample. This sample will be selected to be the point in N for which G is maximized over points of N if $G(i_0, j_0) \geq \overline{G}$, and to be the point on which G is minimized otherwise.

In the following steps, we will assume that (i_k, j_k) was selected to be the k-th sample point.

D8: $D_k = \left(G(i_k, j_k) - \overline{G}\right)^2.$

$D = D + D_k.$

D9: If $D \geq H_k \cdot \mathrm{VAR}$, then return immediately with a decision that the point (i_0, j_0) is out of the region; otherwise, if $D \leq L_k \cdot \mathrm{VAR}$, return immediately with a decision that (i_0, j_0) is in the region; otherwise, continue to D10 (note the scaling of the decision by the local variance).

D10: Add (i_k, j_k) to S. Test $(i_k - 1, j_k)$, $(i_k + 1, j)$, $(i_k, j_k - 1)$, and $(i_k, j_k + 1)$; if any of those points is not in S and inside the picture frame, add it to N. Go to D7.

This algorithm is guaranteed to terminate after n iterations since $L_n = H_n$.

The sampling mechanism tries to find the best evidence that (i_0, j_0) is the start of new region of gray level distributions that is uniformly brighter or darker than the current one. Most of the decisions are reached in the first very few sample points so there is not that much overhead for the decision. The sampling is independent of the shape of the new region so that the sampling can follow a line of new gray values in an arbitrary direction.

4. Option 3

The third option is a modification of the Algorithm D described for option 2. In addition to the statistics $\overline{G}$ and $\overline{G^2}$, option 3 requires maintenance of statistics involving the i, j coordinates of points in the region near the candidate point. The candidate record becomes

$$\left(i, j, \overline{G}, \overline{G^2}, \overline{I}, \overline{J}, \overline{I^2}, \overline{J^2}, \overline{IJ}, \overline{GI}, \overline{GJ}\right)$$

where $\overline{I}$ and $\overline{J}$ are the running averages of i and j, respectively. $\overline{I^2}$ and $\overline{J^2}$ are the running averages of i^2 and j^2, respectively; $\overline{IJ}$ is the running average of $i \cdot j$; and $\overline{GI}$ and $\overline{GJ}$ are the running averages of $G(i, j)i$ and $G(i, j)j$, respectively.

Initial statistics $\overline{G_0}$ and $\overline{G_0^2}$ are obtained in step $\underline{I5}$ as described under option 2.

Step $\underline{I5}$ is expanded to initialize the additional statistics as follows:

$$\overline{I_0} = \frac{\sum_{(i,\ j)\ \epsilon N} i}{\sum_{(i,\ j)\ \epsilon N} 1}$$

$$\overline{J_0} = \frac{\sum_{(i,\ j)\ \epsilon N} j}{\sum_{(i,\ j)\ \epsilon N} 1}$$

$$\overline{I_0^2} = \frac{\sum_{(i,\ j)\ \epsilon N} i^2}{\sum_{(i,\ j)\ \epsilon N} 1}$$

$$\overline{J_0^2} = \frac{\sum_{(i,\ j)\ \epsilon N} j^2}{\sum_{(i,\ j)\ \epsilon N} 1}$$

$$\overline{IJ} = \frac{\sum_{(i,\ j)\ \epsilon N} i \cdot j}{\sum_{(i,\ j)\ \epsilon N} 1}$$

$$\overline{GI} = \frac{\sum_{(i, j) \in N} G(i, j) \cdot i}{\sum_{(i, j) \in N} 1}$$

$$\overline{GJ} = \frac{\sum_{(i, j) \in N} G(i, j) \cdot j}{\sum_{(i, j) \in N} 1}$$

where the set N is defined as before.

The candidate point record list is initialized to be:

$$\left(i_0 + 1, j_0, \overline{G_0}, \overline{G_0^2}, \overline{I_0}, \overline{J_0}, \overline{I_0^2}, \overline{J_0^2}, \overline{IJ_0}, \overline{GI_0}, \overline{GJ_0}\right)$$

$$\left(i_0 - 1, j_0, \overline{G_0}, \cdots, \overline{GJ_0}\right)$$

$$\left(i_0, j_0 + 1, \overline{G_0}, \cdots, \overline{GJ_0}\right)$$

$$\left(i_0, j_0 - 1, \overline{G_0}, \cdots, \overline{GJ_0}\right)$$

assuming these four points are within the picture frame.

These statistics are updated as new candidate records are generated in step L6. If $\left(i, j, \overline{G}, \overline{G^2}, \overline{I}, \overline{J}, \overline{I^2}, \overline{J^2}, \overline{IJ}, \overline{GI}, \overline{GJ}\right)$ is the candidate record of the point (i, j) just added to the region, and the point (k, 1) is a new candidate adjacent to (i, j), then the new candidate record is

$$(k, 1, G', G'', I', J', I'', J'', IJ', GI', GJ')$$

where

$$I' = \frac{i + W \cdot \overline{I}}{1 + W}$$

$$J' = \frac{j + W \cdot \overline{J}}{1 + W}$$

$$I'' = \frac{i^2 + W \cdot \overline{I^2}}{1 + W}$$

$$J'' = \frac{j^2 + W \cdot \overline{J^2}}{1 + W}$$

$$IJ' = \frac{ij + W \cdot \overline{IJ}}{1 + W}$$

$$GI' = \frac{G(i, j)\, j + W \cdot \overline{GI}}{1 + W}$$

$$GJ' = \frac{G(i, j)\, j + W \cdot \overline{GJ}}{1 + W}$$

with G', G'', and W as described under option 2.

When the decision procedure is called in step L3, the statistics in the candidate record $\left(i_0, j_0, \overline{G}, \overline{G^2}, \overline{I}, \overline{J}, \overline{I^2}, \overline{J^2}, \overline{IJ}, \overline{GI}, \overline{GJ}\right)$ are used to obtain A, B, and C, which define the function

$$E(i, j) = Ai + Bj + C$$

E(i, j) is a linear prediction of the grey level value of point (i, j), if the point belongs to the current region. E(i, j) is used in place of $\overline{G}$ in steps D2 and D8. The coefficients A, B, and C are the values that minimize

$$M = \sum \left(G(i, j) - Ai - Bj - C\right)^2$$

where the sum is taken over points of the region near the candidate point. A, B, and C are obtained by solving the system of linear regression equations:

$$A\overline{I^2} + B\overline{IJ} + C\overline{I} = \overline{GI}$$

$$A\overline{IJ} + B\overline{J^2} + C\overline{J} = \overline{GJ}$$

$$A\overline{I} + B\overline{J} + C = \overline{G}$$

Mathematically, we are treating the image points as ordered triples $\left(i, j, G(i, j)\right)$ and finding the best plane fit through the points already belonging to the region in the neighborhood of (i_0, j_0). The decision to add (i_0, j_0) to the region is based on the distance of $\left(i_0, j_0, G(i_0, j_0)\right)$ from this plane, scaled by the quality of the plane fit to points already in the region (VAR). This allows the algorithm to compensate both for changes in gray level intensity due to texture or shading of the object as in option 2, and for structured slopes in the grey levels.

As in option 2, the decision process is controlled by two arrays of decision thresholds, $\left\{L_i\right\}$ and $\left\{H_i\right\}$, $i = 1, \cdots, n$, such that $H_i \geq L_i$ for all i and $L_n = H_n$ where n is the maximum number of sample points.

5. Algorithm E

<u>EQ</u>: compute A, B, and C.

<u>E1</u>: $\text{VAR} = \left(G(i_0, j_0) - A\overline{I} - B\overline{J} - C\right)^2$.

<u>E2</u>: $D_0 = \left(G(i_0, j_0) - E(i_0, j_0)\right)^2$. $E(i_0, j_0)$ is the predicted value of $G(i_0, j_0)$.

$D = D_0$

<u>E3</u>: $D \geq H_0 \cdot \text{VAR}$, then immediately the decision is made not to include (i_0, j_0) in the region. Here, $G(i_0, j_0)$ is too far from the predicted gray level based on points of the region near (i_0, j_0), and the decision is made immediately. Note that the threshold H_0 is scaled by the local variance; hence, in some sense, the noise and texture of the region are compensated for in the decision. This is true for all decisions.

E4: If $D \le L_0 \times VAR$, then the candidate point is immediately added to the regions.

E5: Initialize N and S; set $k = 0$.

Step E5 is reached where $L_0 < \frac{D_0}{VAR} < H_0$ and a sequential decision process has to be initiated. The set S of image points that were already seen by the sampler for point (i_0, j_0) is initialized to include one picture point (i_0, j_0). From that point on, samples will be taken from the set N, which will contain all points adjacent to points in S outside of S (candidate points for the sampler). This sampling will determine whether to add (i_0, j_0) to the region. N is initialized to include the four points. $(i_0 - 1, j_0)$, $(i_0 + 1, j_0)$, $(i_0, j_0 + 1)$ and $(i_0, j_0 - 1)$.

E6: $k = k + 1$.

E7: Select a sample point from N to be the k-th sample. This sample will be the point (i, j) in N for which

$$\left(G(i, j) - E(i, j)\right)^2$$

is a maximum over the points of N; e.g., (i_k, j_k) is the point N that is furthest from the predicted value.

E8: $D_k = \left(G(i_k, j_k) - E(i_k, j_k)\right)^2$.

E9: If $D \ge H_k \cdot VAR$, then return immediately with a decision that the point (i_0, j_0) is out of the region; otherwise, if $D \le L_k \cdot VAR$ return immediately with a decision that (i_0, j_0) is in the region; otherwise, continue to E10 (not the scaling of decision by the local variance).

E10: Add (i_k, j_k) to S. Test $(i_k - 1, j_k)$, $(i_k + 1, j)$, $(i_k, j_k - 1)$, and $(i_k, j_k + 1)$; if any of those points is not in S and inside the picture frame, add it to N. Go to E7.

IV. THE USE OF SEQUENTIAL SAMPLING FOR EDGE DETECTION FOR THE SCANNING REGION GROWER

The scanning region grower (object detector), which is reported in Ref. 8 (see Subsection IV-D), is based on edge confidence evaluation. This edge evaluation is done by an edge detector operator that satisfies two requirements. The first is to return a value 0 at most none edge points; the second is to return a positive value reflecting the confidence in the existence of the edge when it is applied to a point that is likely to be an edge of an object.

The following technique was developed to upgrade performance of the region grower (object locator) based on edge detection, which was reported in Ref. 8. The main motive behind the development of the new edge operation was performance upgrading along the following lines:

(1) Efficient detection of points that are obviously not edge points.

(2) Achievement of independence of the edge value from shape and structure of the objects (lines, sharp corners, etc.).

(3) Keep the edge evaluation self-scaling with respect to linear changes in signal and noise.

To achieve those seemingly conflicting goals, we used the sequential decision approach.

A. SAMPLE POINT SELECTION

The edge detector is built around a sampling mechanism (miniregion grower) that tries to collect the best evidence for the existence of two different gray level distribution structures around the test point. The sampler tries to find a "bright" neighborhood and a "dark" neighborhood around the analyzed point. Let S_D be the dark and S_B the bright neighborhood; then, on initialization; $S_B = S_D = P_0$ where P_0 is the analyzed point. Let

$N_B(N_D)$ be all points adjacent to $S_B(S_D)$ and not in $S_B(S_D)$. The sampler takes the brightest (or darkest) point in $N_B(N_D)$, puts it in $S_B(S_D)$, removes it from $N_B(N_D)$, and updates $N_B(N_D)$.

This is an iterative process and our goal is to stop it early if there is clearly no edge around P_0 and return to value 0, and, if there is an edge, return the "strength" of that edge.

B. STOPPING CRITERIA

Let $(X_1, X_2, \cdots, X_n)$ be the ordered sequence of value readings at the points selected in the bright neighborhood, and $(Y_1, Y_2, \cdots, Y_n)$ the sequence of readings at the points selected for the dark neighborhood. Our first goal is to stop the sampling process if there is no edge. For that purpose we define a series of threshold $(T_1, T_2, \cdots, T_n)$ such that if on the i-th iteration

$$(X_i - Y_i) < T_i \qquad 1 \le i \le n$$

The sampling is stopped and value of 0 (nc-edge) is returned by the operator.

To minimize dependence on the properties of the specific image, the thresholds are picked based on the statistics of the specific image. The histogram of $g(x, y) - g(x + dx, y + dy)$ for $|dx| + |dy| = 1$ is generated by sampling these values from the whole image. Then the T_i - s are defined as percentage thresholds of that histogram. In the liberal case the T_i's were set to the lowest value such that at least $\alpha - \beta/i$ of the pairs sampled has a difference in value less than T_i. The range of i used was between 1 and 5, setting $\alpha = 0.9$ and $\beta = 0.3$ (the percentage threshold starting from 0.6 to 0.84) for the liberal case, and lower (by 2) T_i values for the conservative case.

If the sampling went all the way up to N (a predetermined integer constant set to 5 in our examples) with the differences satisfying the reject threshold, then we compute the edge confidence value as follows:

$$\mu_2 = \frac{\sum_{i=1}^{N} X_i}{N} \qquad \mu_1 = \frac{\sum_{i=1}^{N} Y_i}{N}$$

$$V_2 = \frac{1}{N} \sum_{i=1}^{N} (X_i - \mu_2)^2 \qquad V_1 = \frac{1}{N} \sum_{i=1}^{N} (Y_i - \mu_1)^2$$

$$\mu_0 = \frac{\mu_1 + \mu_2}{2}$$

$$V_0 = \frac{1}{2N}\left(\sum_{i=1}^{N} (X_i - \mu_0)^2 + \sum_{i=1}^{N} (Y_i - \mu_0)^2\right)$$

and the edge confidence value was defined as

$$\text{confidence} = \frac{V_0^2}{V_1 V_2}$$

This value was truncated to 0 if it was less than a certain fixed threshold. Thresholds of at least 16 have a good property: if the samples are taken from a uniform slope area of gray values, e.g.,

$$X_i = A + i \cdot B$$

and

$$Y_i = A - i \cdot B$$

The edge confidence value is 16. We do not want to define points in a uniform gray value slope area as edge points. Results of region growing based on that value are shown below.

Since the confidence value mentioned above is very expensive to compute, we compare its performance with another edge strength evolution, which was

$$\text{strength} = \sum_{i=1}^{N} (X_i - Y_i)$$

Results were substantially better when the variance scaled technique was used.

C. REMARK ON THRESHOLD SETTING

Since many cameras are nonlinear in respect to their light sensitivity, scaling of the threshold with respect to the intensity of the point may be desired. An alternative approach will be to use photon counting cameras instead of regular TV tubes for the input device (this instrument measures the time it takes to receive a specific number of photons from a given direction).

D. THE REGION GROWER

The edge operator is applied to each point in the picture and returns a value of the confidence that the point is an edge point. This data is used to grow regions that are defined as connected valleys of edge values. That is let E(P) be the edge value of the point P; then, for a region R, P_0 is a minimum edge value point. For example,

$$E(P_0) = \min_{P \in R} \{E(P)\}$$

and Q is another point of R; then there is a path P_0, P_1 (P_n = Q) of points from R such that P_i is adjacent to P_{i+1} and $E(P_i) \leq E(P_{i+1})$. A one-pass algorithm that grows those regions is described in detail in Ref. 8.

V. EXAMPLES

This section shows examples of the region grower options described in this paper applied to three scenes (Figs. 4 through 24); these scenes were taken from the RAMTEK RX-100A unit. Examples of a scanning region grower are also included. Each scene has been segmented four times using the options described in this paper as follows:

(1) Option 1: TR = 12.

(2) Option 2: $L_0 = 3$, $H_0 = 13$, $n = 20$, $L_{20} = H_{20} = 4$, $W = 5$.

(3) Option 2: $L_0 = 4$, $H_0 = 16$, $n = 20$, $L_{20} = H_{20} = 10$, $W = 15$.

(4) Option 3: $L_0 = 4$, $H_0 = 16$, $n = 20$, $L_{20} = H_{20} = 10$, $W = 5$.

Notice that option 2 was used twice, first with "conservative" thresholds, and then with "liberal" thresholds. With the conservative thresholds, there is a tendency to stop before reaching the actual region boundaries, but points that do not belong to the region are rarely included. With the liberal thresholds, there is a tendency to include points that do not belong to the region, but points that do belong to the region are rarely excluded. For example, notice how the sky has been segmented in Figs. 6 and 7. We feel that by applying a combination of region grower options and/or thresholds to a scene, by a semantic system, segmentation errors can be detected and a good set of region boundaries can be obtained for the scene.

Each scene also has been segmented twice with a sequential decision based scanning region (Section IV) grower, using conservative and liberal thresholds, to show a comparison of the methods.

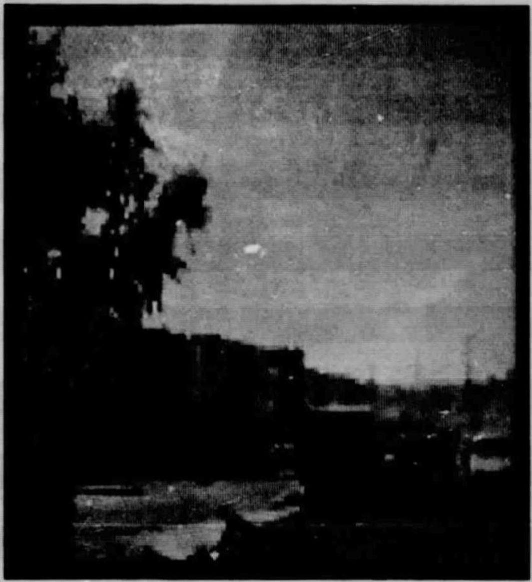


Fig. 4. Outdoor scene

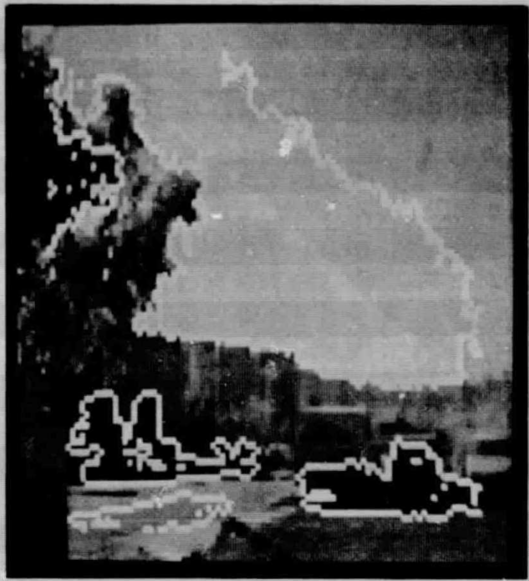


Fig. 5. Segmentation with option 1

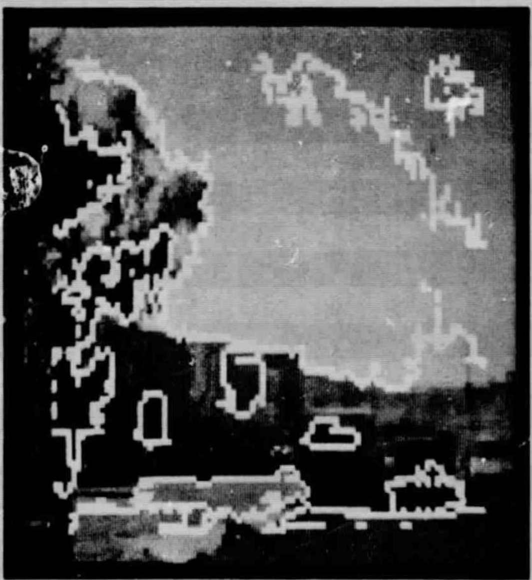


Fig. 6. Segmentation with option 2 — conservative

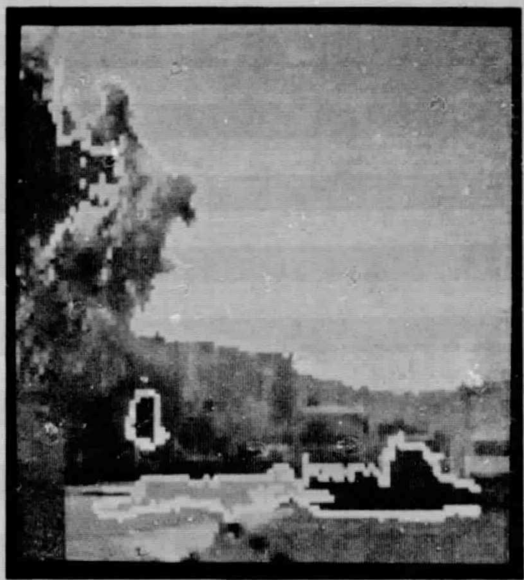


Fig. 7. Segmentation with option 2 — liberal

Fig. 8. Segmentation with option 3

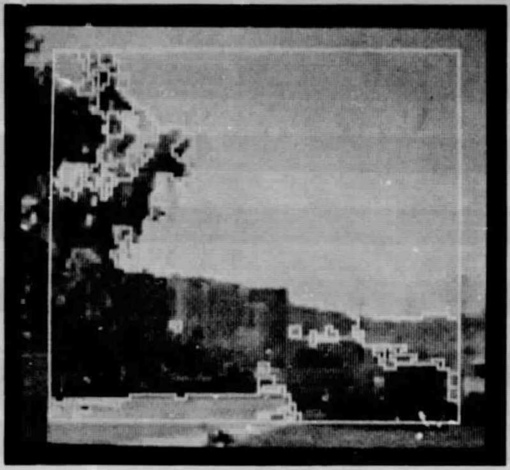


Fig. 9. Segmentation with scanning region grower — liberal

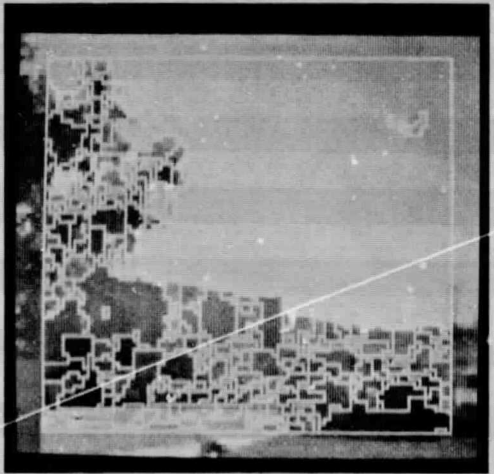


Fig. 10. Segmentation with scanning region grower — conservative

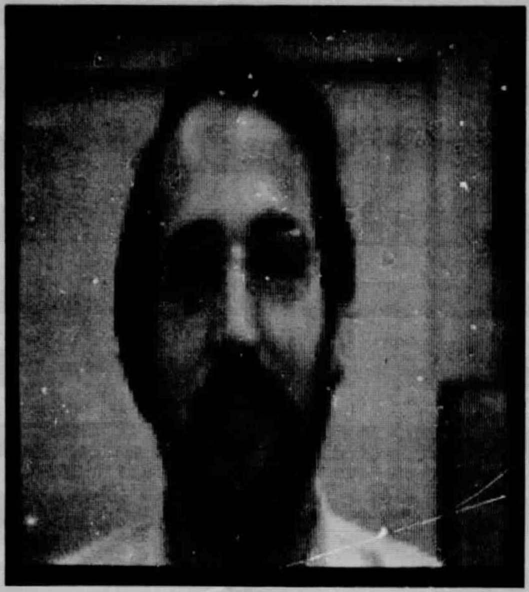


Fig. 11. A face

Fig. 12. Segmentation with option 1

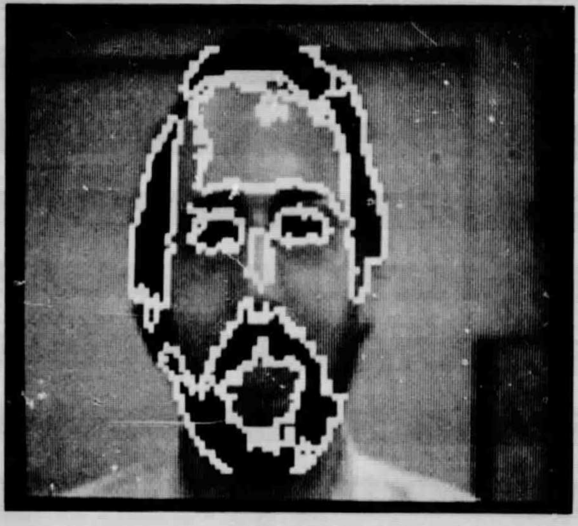


Fig. 13. Segmentation with option 2 — conservative

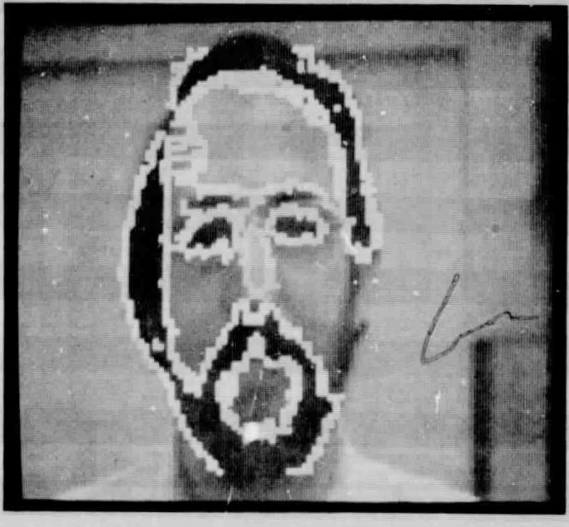


Fig. 14. Segmentation with option 2 — liberal

Fig. 15. Segmentation with option 3

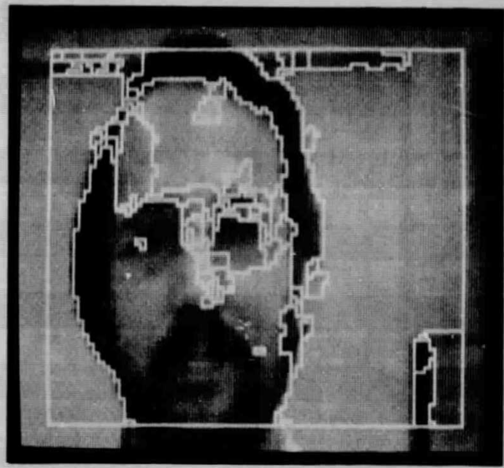


Fig. 16. Scanning region grower segmentation — liberal

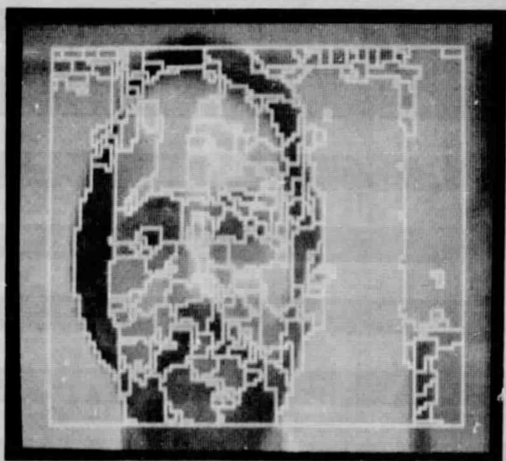


Fig. 17. Scanning region grower segmentation — conservative

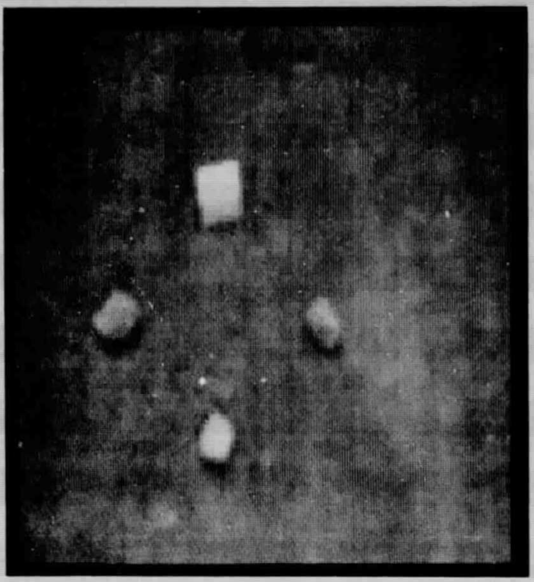


Fig. 18. Some objects on the floor

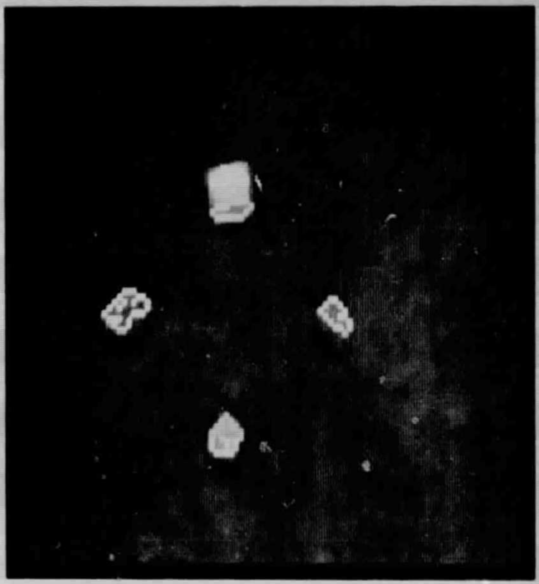


Fig. 19. Segmentation with option 1

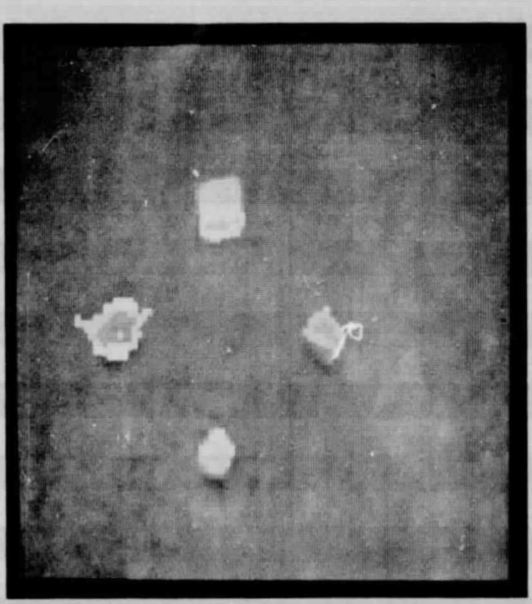


Fig. 20. Segmentation with option 2 — conservative

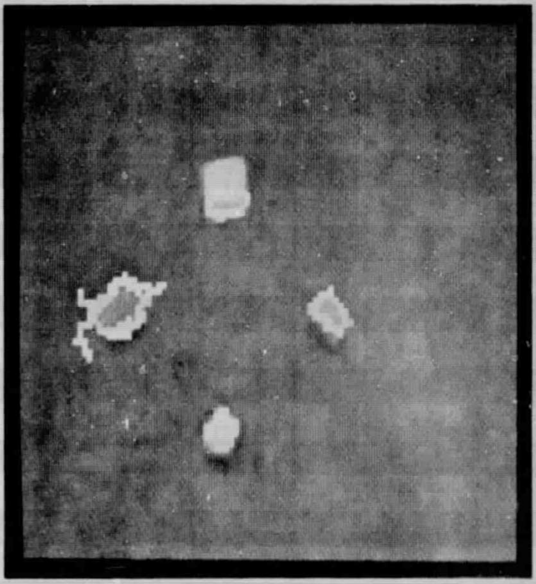


Fig. 21. Segmentation with option 2 — liberal

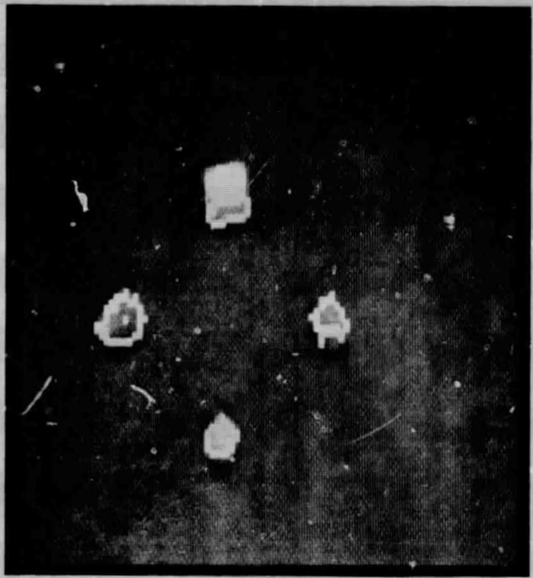


Fig. 22. Segmentation with option 3

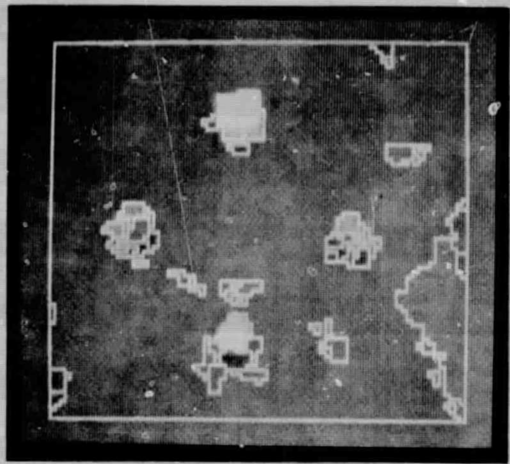


Fig. 23. Scanning region grower segmentation — conservative

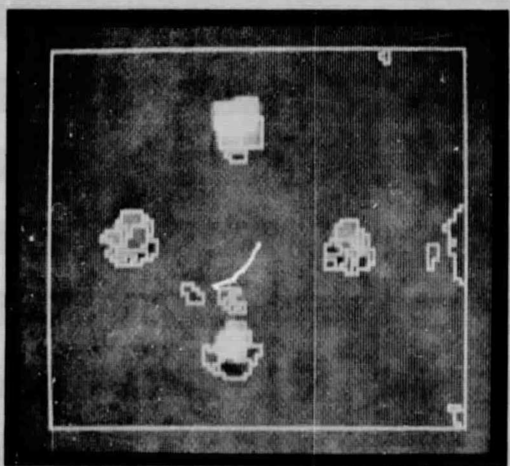


Fig. 24. Scanning region grower segmentation — liberal

REFERENCES

1. Feldman, J., and Yakimovsky, Y., "Decision Theory and Artificial Intelligence: I). A Semantics-Based Region Analyzer," Artificial Intelligence Journal, 5, 1975.

2. Harlow, C., and Eisenbeis, S., "The Analysis of Radiographic Images," I.E.E.E. Transactions on Computers, Vol. C-22, 1973.

3. Wintz, P., and Gupta, J., Multi-Image Modeling, TR-EE-1974-24. Purdue University, West Lafayette, Indiana, 1974.

4. Tenenbaum, J., and Barrow, H. G., Experiments in Interpretation Guided Segmentation, TM-123. Stanford Research Institute, Menlo Park, Calif., March 1976.

5. Yakimovsky, Y., Cunningham, R., DABI: A Non-Deterministic Data Base for Image Understanding, Technical Memorandum 33-773. Jet Propulsion Laboratory, Pasadena, California, May 1976.

6. Yakimovsky, Y., "Scene Analysis Using a Semantic Base for Region Growing," Memos AIM-209 (STAN- -73-380). Stanford University Artificial Intelligence Laboratory, Palo Alto, Calif., June 1973.

7. Wall, R., The Gray Level Histogram for Threshold Boundary Determination in Image Processing with Application to Scene Segmentation in Human Chromosome Analysis, PhD dissertation. School of Engineering, University of California at Los Angeles, 1974.

8. Yakimovsky, Y., "Boundary and Object Detection in Real World Images" (to be published in J.A.C.M.). Also in Proceedings of the 4th Joint Conference on Artificial Intelligence, 1975, p.695.

9. Brice, C., and Fennem, C., "Scene Analysis Using Regions," Artificial Intelligence Journal, 1, 1970.

10. Yakimovsky, Y., and Karspeak, M., "Sequential Decision Based Edge Detector," Proceedings of the Conference on Computer Graphics, Pattern Recognition and Data Structures, Los Angeles, Calif., May 1975.

11. Yakimovsky, Y., Rayfield, M., and Eskenazi, R., "RAPID: A Random Access Picture Digitizer Display and Memory System," Technical Memorandum 33-772. Jet Propulsion Laboratory, Pasadena, California, May 1976.

12. Barrow, H., and Popplestone, R., "Relational Descriptions in Picture Processing," Machine Intelligence, Vol. 6, Edinburgh University Press, 1971.

13. Pavlidis, T., "Fuzzy Representation as Means of Overcoming the Overcommitment of Segmentation," Proceedings of the Conference on Computer Graphics Pattern Recognition and Data Structure, May 1975, p. 215.

14. Tenenbaun, J., Barrow, H., and Wejl, S., "Research in Interactive Scene Analysis." Final report S.R.I. Project 8721. Stanford Research Institute, Menlo Park, Calif., January 1976.

NASA—JPL—Coml. L.A. Calif