N90-25577

EXTENSIONS OF ALGEBRAIC IMAGE OPERATORS: AN APPROACH TO MODEL-BASED VISION[1]

Bao-Ting Lerner[2]
Associate Professor
US Naval Academy
Annapolis, Maryland 21402
and
NASA/Goddard Space Flight Center
Robotics Laboratory
Greenbelt, Maryland 20771

Michael V. Morelli
Electrical Engineering Department
Fairleigh Dickinson University
Teaneck, New Jersey 07666
and
Jackson and Tull Chartered Engineers
NASA/Goddard Space Flight Center
Robotics Laboratory
Greenbelt, Maryland 20771

ABSTRACT

We extend our previous research on a highly structured and compact algebraic representation of grey-level images which can be viewed as fuzzy sets. Addition and multiplication are defined for the set of all grey-level images, which can then be described as polynomials of two variables.

Utilizing this new algebraic structure, we have devised an innovative, efficient edge detection scheme. An accurate method for deriving gradient component information from this edge detector is presented.

Based upon this new edge detection system we have developed a robust method for linear feature extraction by combining the techniques of a Hough transform and a line follower. The major advantage of this feature extractor is its general, object-independent nature. Target attributes, such as line segment lengths, intersections, angles of intersection, and endpoints are derived by the feature extraction algorithm and employed during model matching.

The algebraic operators are global operations which are easily reconfigured to operate on any size or shape region. This provides a natural platform from which to pursue dynamic scene analysis. We discuss a method for optimizing the linear feature extractor which capitalizes on the spatially reconfigurable nature of the edge detector/gradient component operator.

PRECEDING PAGE BLANK NOT FILMED

INTRODUCTION

There have been various algebraic approaches proposed for the description and generation of binary images[2][6]. In this paper we extend our previous work[1][3] on a polynomial architecture based on fuzzy sets for the represention, interpretation and manipulation of grey-scale imagery. An edge detector and a "compass" gradient component operator are defined for this algebraic system, and are employed as the front end of a robust linear feature extractor.

ALGEBRAIC REPRESENTAION OF GREY-LEVEL IMAGES

In a previous paper [1][3] we introduced a polynomial architecture for the description of grey-level (and binary) images. If Z denotes the integers, then an image A may be represented as

$$A = \{(x,y) \in Z \times Z \mid \mu_A(x,y) \in [0,1]\},$$

where $\mu_A(x,y)$, denoting the grey-level of pixel (x,y), is a real number between 0 and 1 with the larger numbers corresponding to the darker grey-levels. In this way A may be viewed as a fuzzy set.

Image Addition and Multiplication:

Let A and B be grey-level images. We define the addition of A and B as:

$$A + B = \Big\{(x,y) \in A \cup B \mid \mu_{A+B} =$$

$$\min[\max(\mu_A(x,y),\mu_B(x,y)),\max(1-\mu_A(x,y),1-\mu_B(x,y))]\Big\} \quad (1)$$

Observe that in the binary case (with $\mu =$ either 1 or 0) the addition reduces to the "exclusive-or" addition defined by Agui, et.al. [2]. That is, in binary the set theoretic addition of images A and B is

$$A + B = (A \cup B) \cap (\overline{A} \cup \overline{B}) \quad (2)$$

where $\overline{A}$ and $\overline{B}$ are the inverse images of A and B respectively.

We define the multiplication of grey-level images A and B by:

$$A * B = \Big\{ \sum_{\substack{(x_1,y_1) \in A, \\ (x_2,y_2) \in B}} (x_1+x_2,y_1+y_2) \mid \mu_{A*B}(x_1+x_2,y_1+y_2) = \min[\mu_A(x_1,y_1),\mu_B(x_2,y_2)] \Big\}$$

$$(3)$$

where $\sum$ denotes the addition performed by (1).

Note that the operations of + and * are commutative and associative, thus making the set of all grey-level images a commutative semigroup under each of these operations.

Polynomial Representation of Grey-level Images:

Under the addition and multiplication operations defined in the previous section, we may represent an image A as the polynomial

$$A = \sum_{(m,n)\in A} \mu_{mn}x^m y^n$$

(4)

where the coefficient is the grey-level of pixel (m,n) and the summation is defined by (1).

ALGEBRAIC EDGE DETECTORS

Agui, et.al [2] described the classical differential operator

$$D1 = (1+x)(1+y)$$

(5)

and applied it to binary images to produce contour-like figures. The action of D1 on the nine pixel square resulted in a "spotty" edge or contour as depicted in figure 1.


(a)　　　　　　　　(b)

Figure 1. (a) Simple binary nine pixel image A, (b) the result of D1*A, the differential operator acting on A

We then varied the classical differential operator to the form
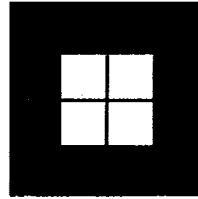
$$D2 = 1+xy$$

(6)

This new, more compact irreducible polynomial operator gave a substantially improved contour enhancement capability. Its action on a nine pixel square appears in figure 2.

Although D2 proved to be a simple, efficient operator for contour enhancement in images with a preponderance of vertical and horizontal lines, its major drawback was the introduction of a bias along the -45 degree line, as can be seen in figure 2. In the next section we shall discuss the derivation of accurate gradient component information from analysis of this bias.

(a)

(b)
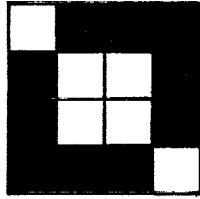
Figure 2. Operation of D2*A.   Figure 3. Operation of D3*A.

To overcome this bias we varied the differential operator on an image A to

$$D3 *A = \{ (m,n)\in (A+Axy)\cup(Ax+Ay) \mid \mu_{D3*A}(m,n)=$$

$$\max[\mu_{A+Axy}(m,n),\mu_{Ax+Ay}(m,n)] \} \qquad (7)$$

Figure 3 depicts the action of D3 on a nine pixel square. Notice the continuous, unbroken contour. The definition of the min-max addition produces a natural threshold value of 0.5. Unlike classical edge detectors which require the determination of the optimal threshold from the statistics of the contoured image, this natural threshold is a constant for all conditions, thus greatly simplifying the production of the edge from the contour. The edge produced by the action of each of the three defined differential operators on a complicated scene is shown in figure 4a - 4c.

(a)
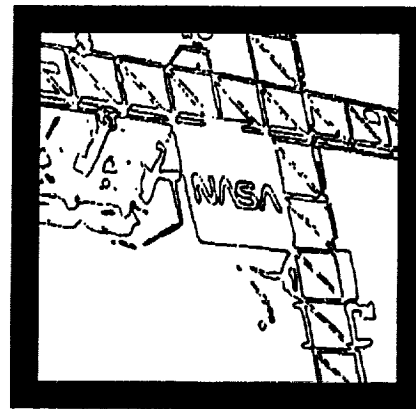
(b)
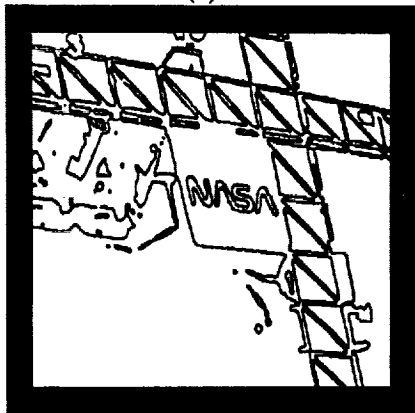
(c)

Figure 4. (a) The operation of D1 on a complex scene. Notice the "spotty" edge. (b) D2 applied to the same scene. This edge is better; however the bias is apparent on the diagonal struts. (c) The operation of D3 on the scene. This operator contains no bias and produces a solid edge.

GRADIENT COMPONENT INFORMATION FROM ALGEBRAIC DIFFERENTIAL OPERATORS

We have derived accurate gradient component information by utilizing the bias (mentioned in previous section) resulting from individual differential operations and the underlying algebraic structure. We interpret the differential operation as being a measure of a feature's spatial alignment with that of the differential operator orientation (the differential vector). A characteristic of the addition (1) is that as the differential vector converges to the feature orientation, the number of resultant edge elements (edgels) produced by that operator decreases. At the point which the differential operator and feature are aligned, no edgels are produced by the operator. Conversely, the population of edgels increases as the differential vector diverges from the feature orientation (Fig 5). There is a strong quantitative as well as qualitative correspondence between the number of edgels produced by a differential operator and the orientation of the feature relative to that differential vector.

The differential operation is symmetric and unsigned, hence four operators are required to identify all unique (non-redundant) orientations. We define the four component operators as:
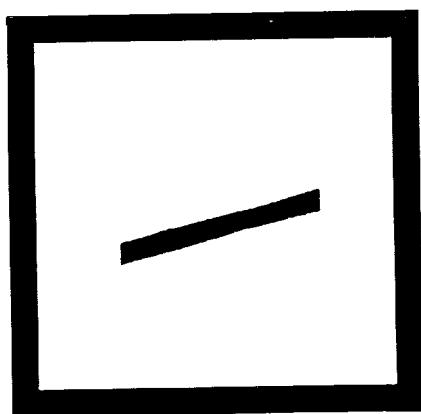
$$G_x = (1+x) \tag{8}$$

$$G_y = (1+y) \tag{9}$$

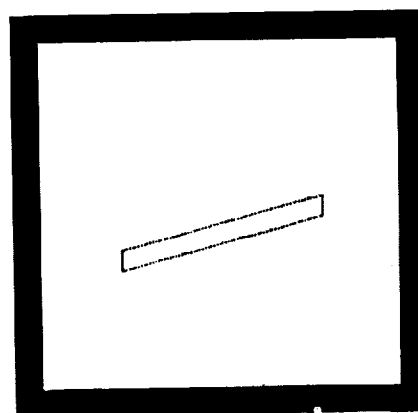$$G_{xy} = (1+xy) \tag{10}$$

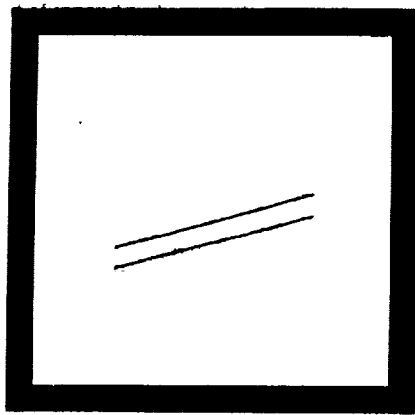$$G_{xy^{-1}} = (1+xy^{-1}) \tag{11}$$

Notice that the operators (8), (9), and (10) were used previously to define D2 and D3. The operation of each of these component operators on a simple binary test image is shown in figure 5a - 5e.
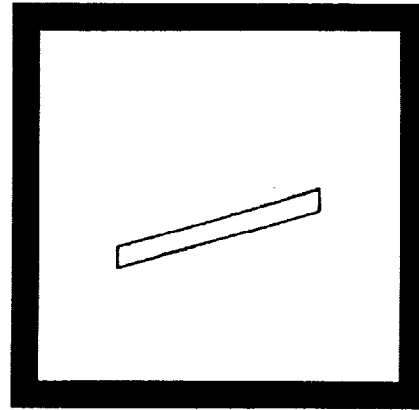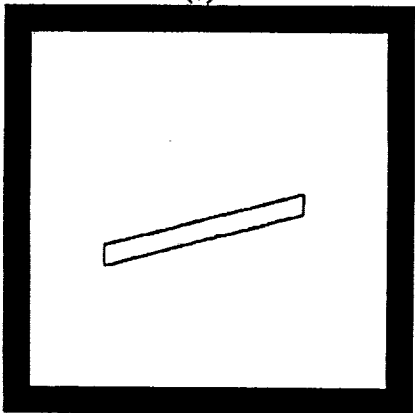

(a)　　　　　　　　　　　　　　　(b)

(c)

(d)

(e)

Figure 5 (a) A test pattern oriented
15 degrees relative to the abscissa.
This was generated by a Bresenham
algorithm. (b) The operation of Gx,
(c) Gy, (d) Gxy, (e) Gxy^{-1}. The results
of the edgel count for each operation:

Gx   : 148
Gy   : 516
Gxy   : 390

Gxy^{-1} : 662

The image is operated on by each of the gradient operators and the number of edgels contributed by each component is compiled. The resultant component operator with the fewest contributions is the closest to the feature orientation, and is identified as the primary component (Gp). The component operator which contributes the next to the least edgels is identified as the secondary component (Gs). Gp and Gs bound the sector in which the feature is oriented. To determine the actual orientation of the feature relative to the primary component, the inverse trigonometric relationship (12) between the edgel count of Gp and the edgel count of its orthogonal component operator (Gp⊥) is employed.

$$\text{orientation} = \arctan(\text{Gp count}/\text{Gp}\perp \text{ count}) \qquad (12)$$

As an example, consider the test image data of figure 5. The test line is draw with a Bresenham algorithm at an orientation of 15 degrees from the horizontal axis. From the data listed in figure 5 we identify : Gp = Gx, Gs = Gxy and Gp⊥ = Gy. The relative measure from the Gp by (12) is 16.0 degrees (It should be pointed out that accuracy of this operation degrades at very low spatial resolution.)

Notice that (12) generates no sign information, only a relative measure from the primary component vector. The correct placement of this measure from the primary vector (one side or the other) is fixed by the secondary component. Hence letting the global orientation of Gx be zero degrees, the global orientation of the feature in figure 5 is (0+orientation)=16.0 degrees.

A major advantage of this edge detector/gradient component system is that it inherently is a global operation; however reconfiguring it as a spatially symmetric or asymmetric local operation is a trivial matter. This property allows for the compilation of the gradient components over any size and shape regions. In the last section we shall discuss how this reconfiguration is perfectly suited for dynamic scene processing.

LINEAR FEATURE EXTRACTOR

The algebraic edge detector/gradient component operator has been employed as the front end of a linear feature extraction system which is based on the Hough transform. Our primary goal is to establish and maintain object independence and system invariance to object rotation, translation and scaling in the image. The system takes as input the edge and gradient component information (as discussed in the next section) and produces a list of line segments.

The Hough transform is a technique for parameterizing binary image points into a family of curves representing the analytical equations of image features [4][5]. We employed a linear transform which is based upon the form

$$\rho = x*\cos(\theta)+y*\sin(\theta), \tag{13}$$

where x and y are the coordinates of the edgel, θ is the evaluated orientation relative to the x axis (evaluated in the range of $-\pi/2$ to $\pi/2$), and ρ is the normal vector to the line of that orientation. The evaluation of (13) produces the transform parameter space.

We have developed an adaptable and reliable method for processing the Hough transform parameter space to extract unique image feature equations[3]. The analytical feature representations are then utilized to locate actual feature segment endpoints via a line following/segment locating mechanism.

The processing sequence of the linear feature extraction system is shown in figure 6a - 6e. Details of this system can be referenced in [3].

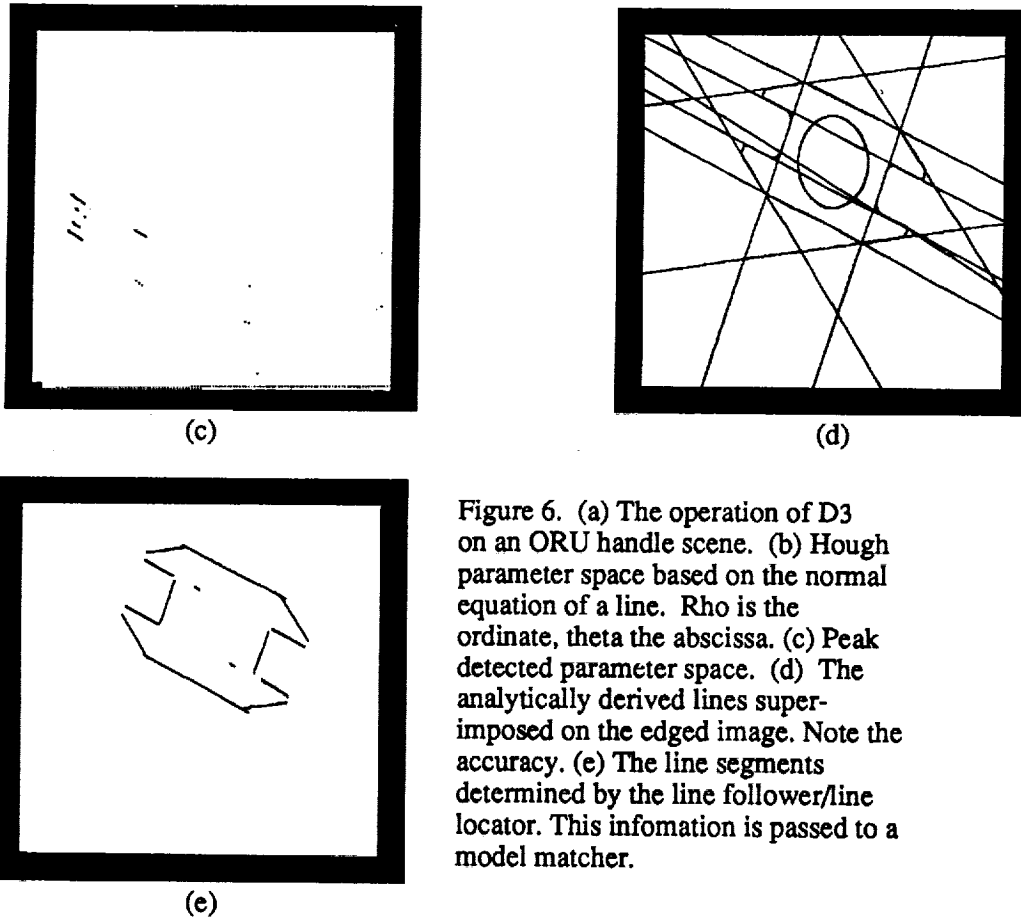(a)   (b)

(c)

(d)

(e)
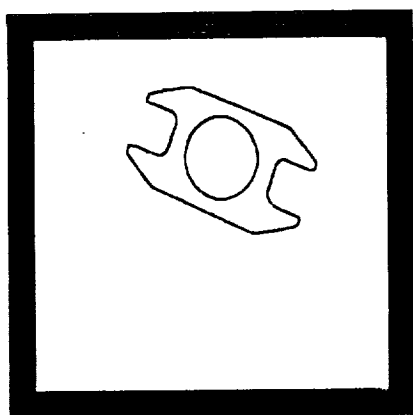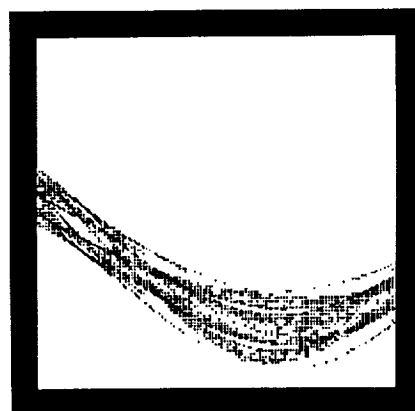
Figure 6. (a) The operation of D3 on an ORU handle scene. (b) Hough parameter space based on the normal equation of a line. Rho is the ordinate, theta the abscissa. (c) Peak detected parameter space. (d) The analytically derived lines super-imposed on the edged image. Note the accuracy. (e) The line segments determined by the line follower/line locator. This infomation is passed to a model matcher.

DYNAMIC SCENE ANALYSIS

The major drawback of the Hough transform is the large amount of computation that it requires. It can be seen from (13) that the number of iterations will equal the number of edgels multiplied by the theta resolution. It is apparent that the optimization of the transform requires the reduction of the parameter space theta resolution as well as the area of the image which is transformed. The intent is to perform this optimization in a way which does not violate the goals of object independence and system invariance to rotation, translation and scaling. This can be accomplished by evaluating the image only in desired areas and at high theta resolution at selected orientations. This optimizing information can be derived, in a dynamic sence, from previous processing. In previous work [3] we discuss the optimization of the Hough parameter space via the utilization of information from a geometric world model.

We have recently developed a mutually optimizing system in which spatial and gradient information from the algebraic edge/gradient component operator optimizes the Hough transform, and the confirmed object feature information is used to redefine the size and shape of the algebraic operator. In this way the highest degree of accuracy, optimization and confirmation is achieved.

694

The unique ability to redefine the size and shape of the algebraic edge detector/gradient component operator provides a natural environment to perform this optimization.

CONCLUSION

The algebraic approach provides an innovative methodology as well as a sound mathematical framework for image representation and manipulation. It is significant that this algebraic system provides insight into the morphological structure of the image itself, as well as the operators on the image. Real-time implementation of these operations can be easily realized due to the efficient min-max nature of our algebraic system. The simple extraction of the gradient component information from the polynomial structure demonstrates the strong potential of this approach.

ACKNOWLEDGEMENTS

REFERENCES

[1]   B. Lerner and H. Thomas, "Polynomial Representation of Grey-level Images", SPIE Applications of Artificial Intelligence VI, Vol. 937, April 1988, pp. 104-116.

[2]   T. Agui, M. Nakajima, and Y. Arai, "An Algebraic Approach to the Generation and Description of Binary Pictures", IEEE Trans.on PAMI, Vol. PAMI-4, No. 6, Nov. 1982, pp. 635-641.

[3]   B. Lerner, M. Morelli, and H. Thomas, "Applications of Algebraic Image Operators to Model-Based Vision", SPIE Applications of Artificial Intelligence VII, Vol. 1095, March 1989, pp. 36-47.

[4]   D. H. Ballard and C. M. Brown, *Computer Vision*, Englewood Cliffs, N.J. : Prentice-Hall 1982.

[5]   J. Illingworth and J. Kittler, "A Survey of the Hough Transform", Computer Vision, Graphics, and Image Processing, No 44, 1988, pp. 87-116.

[6]   G.X. Ritter and P.D. Gader, "Image Algebra Techniques for Parallel Image Processing", Journal of Parallel and Distributive Computing 4(5), March 1987.