DTIC FILE COPY

E801833

2

AFATL-TR-88-12

AD-A203 994

Image Processing Language—Phase II

Charles R Giardina
Edward R Dougherty

SINGER
ELECTRONIC SYSTEMS DIVISION
WAYNE, NJ 07474-0975

NOVEMBER 1988

FINAL REPORT FOR PERIOD AUGUST 1985-MAY 1987

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

AIR FORCE ARMAMENT LABORATORY
Air Force Systems Command ▪ United States Air Force ▪ Eglin Air Force Base, Florida

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
Unclassified

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

2b. DECLASSIFICATION / DOWNGRADING SCHEDULE

3. DISTRIBUTION / AVAILABILITY OF REPORT
Approved for Public Release.
Distribution is Unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

5. MONITORING ORGANIZATION REPORT NUMBER(S)
AFATL-TR-88-12

6a. NAME OF PERFORMING ORGANIZATION
Singer

6b. OFFICE SYMBOL *(If applicable)*
CN975

7a. NAME OF MONITORING ORGANIZATION
Air-to-Surface Guidance Branch (AGS)
Advanced Guidance Division (AG)

6c. ADDRESS *(City, State, and ZIP Code)*
Electronic Systems Division
Wayne, N. J. 07474-0975

7b. ADDRESS *(City, State, and ZIP Code)*
Air Force Armament Laboratory
Eglin AFB, FL 32542-5434

8a. NAME OF FUNDING / SPONSORING ORGANIZATION
AFATL/AGS & DARPA/TTO

8b. OFFICE SYMBOL *(If applicable)*

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
F08635-84-C-0296

8c. ADDRESS *(City, State, and ZIP Code)*
AFATL/AGS
Eglin AFB, FL
32542-5434

DARPA/TTO
1400 Wilson Blvd
Arlington,VA22209

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
62602F	2068	06	52

11. TITLE *(Include Security Classification)*
Image Processing Language - Phase II

12. PERSONAL AUTHOR(S)
Charles R. Giardina and Edward R. Dougherty

13a. TYPE OF REPORT
Final

13b. TIME COVERED
FROM Aug 85 TO May 87

14. DATE OF REPORT *(Year, Month, Day)*
November 1988

15. PAGE COUNT
112

16. SUPPLEMENTARY NOTATION
Availability of this report is specified on verso of front cover

17. COSATI CODES

FIELD	GROUP	SUB-GROUP
16	04	
17	07	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
Image Algebra, Image Algorithm, Image Processing, Artificial Intelligence, Pattern Recognition, Image Language

19. ABSTRACT *(Continue on reverse if necessary and identify by block number)*

An image processing language in the form of an algebra is developed to serve as a framework to express image and signal processing algorithms. The operators in the algebra can be expressed in terms of a set of seven primative, or basis operators which have been induced from the underlying algebraic structures of the domain and range spaces. Whereas, the inducement methodology is rather general and its application is not restricted to image processing, the bulk of the report is concentrated on the development of a digital image algebra. In terms of this algebra, operators such as dilation and convolution, which are often defined pointwise, possess closed-form expressions as strings of the seven basic operations. A structured approach to the development of intelligent weapon systems is presented herein with an emphasis in automated image processing algorithm development. These artificial intelligence concepts are implemented in PASCAL.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT
☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL
Bruce T. Buller

22b. TELEPHONE *(Include Area Code)*
904-882-2838

22c. OFFICE SYMBOL
AFATL/AGS

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

PREFACE

This program was conducted by the Singer Company, Electronic Systems Division, 164 Totowa Road, Wayne, New Jersey, 07474-0975, under Contract Number F08635-84-C-0296 with the Air Force Armament Laboratory, Eglin Air Force Base, Florida 32542-5434. Mr. Charles R. Giardina was the principal investigator, and Mr. Edward R. Dougherty was the consultant. Mr. Neal Urquhart (AFATL/AGS) managed the program for the Armament Laboratory. The program was conducted during the period from August 1985 to May 1987.

The authors wish to extend appreciation to both the Air Force Armament Laboratory and the Defense Advanced Research Project Agency for sponsoring the image processing language program. In particular, the efforts of Mr. Neal Urquhart (AFATL/AGS) deserve notice. His recognition of the need for an image algebra has brought the project about, and his counsel has been essential for providing direction.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

SECTION I
INTRODUCTION

A complete image processing language in the form of an image processing algebra is presented herein. It enables image and signal processing algorithms to be written in terms of seven primitive or basis operators. Each of the seven basis operators are induced from the underlying algebraic structures of the domain or range spaces.

The inducement methodology developed in the report is an application of a general theory of inducement which is not restricted to image processing. However, the bulk of the report is concentrated on the development of a digital image algebra. In terms of this algebra, operators such as dilation and convolution, which are often defined pointwise, possess closed-form expressions as strings of the seven basic operators.

The image algebra developed herein has provided a universal environment in which to express image processing algorithms. Moreover, such an environment illustrates standardization of algorithm specification.

The simplicity of the algebra is illustrated by the manner in which a programming language can be based upon the seven basis operators of the digital image algebra.

Recent results in knowledge engineering are also discussed herein. The development of the seven basis operators was based on the work done in Phase I of the Image Processing Language program (Contract #F08635-84-C-0296) which preceded Phase II and was reported by Singer Electronic Systems in the Image Processing Language Final Report for Phase I dated September 1984 - September 1985. An argument is made for a structured approach to the development of intelligent weapon systems, in general, and to automated image processing algorithm development, specifically.

SECTION II

IMAGE ALGEBRA

1. DIGITAL IMAGES

The primary goal of an image algebra is the development of a mathematical environment in which to express the various algorithms employed in image processing. From a practical standpoint, this means that the algorithms should appear as strings in an operational calculus, where each operator can ultimately be expressed as a string composed of some collection of elemental, or "basis" operators and where the action of the string upon a collection of input images is determined by function composition. For instance, rather than defining operations such as convolution and dilation in a pointwise manner, closed-form expressions of these operators in terms of low-level operations that are close to the algebraic structure of the underlying mathematical entities upon which images are modeled are desirable.

In Reference 1, Reference 2 and Reference 3 image processing algorithms are presented through the use of bound matrix representation and block diagrams constructed of various operations upon bound matrices. The intent of the approach is to present a unified framework for the orderly expression of the diverse algorithms that play roles in image analysis. In essence, the block diagram technique serves as a univeral language in which investigators from diverse backgrounds may find a common understanding. In this report, the algebraic foundations of that block diagram language are explored (although the algebraic methodology presented herein is independent of both image processing and bound matrix representation in particular). Speaking analogically, image algebra is to bound matrix algebra as linear algebra is to matrix algebra.

The formulation of a satisfactory image algebra is actually a particular instance of a more general algebraic problem concerning function spaces. Indeed, an image signal is simply a function defined on a subset of some universal set. Thus, once the function space is specified, and the determination of the desired algebra depends on the collection of operations induced from the structure of the domain and range spaces. In digital image processing, the domain space is the extent of the image and is usually considered to be the integral lattice $Z \times Z$. While the range space consists of grey values, it can be represented by either the set R of real numbers or the set Z of integers. Of significance is that when the underlying problem of algebra development is viewed from the appropriate perspective, the particulars of the subject matter involved are of minor importance; rather, it is the methodology of development that is paramount.

2. THE INDUCEMENT METHODOLOGY

An abstract image is defined as a function $f:A \rightarrow Y$, where A is a subset of X and X and Y are arbitrary non-empty sets. Q then denotes the set of all images with domain space X and range space Y. If it is then assumed that both X and Y are algebraic structures, these structures induce corresponding algebraic properties into the set Q: i.e. algebraic operations are inherited in Q because of the algebraic properties in X and Y. Inducement methodology is a standard algebraic technique which is applied to the development of an image and which uses simple algebraic constructs for the purpose of processing images.

Assume Y is a group under the operation +. Then the binary operation <+> is range induced on the function space Q in the following manner:

If $f: A \rightarrow Y$ and $g: B \rightarrow Y$, then we define

$$f \langle + \rangle g: A \cap B \rightarrow Y$$

by

$$(f \langle + \rangle g)(x) = f(x) + g(x)$$

This is the standard procedure, for example, in defining function addition where the domains need not be identical. The inducement procedure leaves open the definition of the intersection of the domains, and this is significant in image processing, where many binary operations involve images with different domains. Moreover, because the operation <+> has been defined by inducement, the group properties of + also induce: <+> is associative and, if the group (Y, +) is commutative, then <+> is commutative. Although the case where (Y, +) is a group, has been considered in detail, the group hypothesis was assumed only for illustration. The methodology of range inducement applies to other algebraic structures as well.

The existence of a transformation group (G, *) consisting of 1-1, onto mappings $t: X \rightarrow X$ is assumed. In other words, if t, s: $X \rightarrow X$ with t, s in G, then the composition t*s is also an element of G, and, in addition, * satisfies the group axioms. G induces a binary mapping Q x G → Q in the following manner:

if $f: A \to Y$ and t lies in G, then

$$\{G\}(f, t): t(A) \to Y$$

by

$$(\{G\}(f, t))(x) = f(t^{-1}(x))$$

The properties of * induce corresponding properties of {G}. For instance, the associativity of * yields

$$(\{G\}(f,(t*r)*s))(x) = f(((t*r)*s)^{-1}(x))$$
$$= f((t*(r*s))^{-1}(x)) = (\{G\}(f,t*(r*s)))(x)$$

and thus {G} is associative with respect to the second variable, i.e., for fixed f, the action of the inducement is associative. This is significant, since in practice it is fixed in G and affects the mapping of

$$\{G;\ t\}: Q \to Q$$

by

$$\{G;\ t\}(f) = \{G\}(f,t)$$

By the associativity of *,

$$\{G;\ (t*r)*s\} = \{G;\ t*(r*s)\}$$

If I is the identity transformation, $I(f) = f$, then

$$\{G;\ t^{-1} * t\} = I$$

If $*$ happens to be commutative, then

$$\{G;\ t*s\} = \{G;\ s*t\}$$

In digital image processing, where $X = Z \times Z$ and $Y = R$, consider the addition operation + on R. Then, if f and g are images with domains A and B, respectively, the range inducement procedure leads to the operation $\langle + \rangle$, where

$$(f \langle + \rangle g)(i,j) = f(i,j) + g(i,j)$$

for all (i,j) pixels in $A \cap B$. Multiplication, maximum, minimum, and division similarly induce $\langle . \rangle$, $\langle \vee \rangle$, $\langle \wedge \rangle$, and $\langle / \rangle$, respectively, except in the case of division, where $f \langle / \rangle g$ is only defined at points of the intersection where g is nonzero.

If consideration is now given to the group T of translations of the grid onto itself, for any (u, v) in $Z \times Z$, there is a translation transformation.

$$\overline{(u,v)}: Z \times Z \xrightarrow[\text{onto}]{1-1} Z \times Z$$

defined by

$$\overline{(u,v)}(i,j) = (i + u,\ j + v)$$

The group operation $*$ is defined by

$$(\overline{(u,v)}*\overline{(m,n)})(i,j) = (i + u + m,\ j + v + n)$$

and inversion is given by

$$\overline{(u,v)}^{-1}\ (i,j) = (i - u,\ j - v)$$

so that

$$\overline{(u,v)}^{-1} = (-u,\ -v)$$

According to the inducement procedure, for any ordered pair (u,v), there is a mapping

$$\{T;\ u,\ v\}: Q \to Q$$

defined by

$$(\{T;\ u,v\}(f))(i,j) = f(\overline{(u,v)}^{-1}\ (i,j)) = f(i-u, j-v)$$

In other words, {T; u, v} does not alter the grey values of f; rather, it simply moves the image along the vector (u,v).

Besides the translation group on Z x Z, the group of symmetries of the square can also be used. This group has two generators, N and D, which are defined by N(i,j) = (-j,i) Rotation, and D(i,j) = (-j,-i), flip, respectively. Geometrically, N represents a 90-degree rotation of the grid in a counterclockwise direction and D represents the flipping of the grid (out of the plane) around the one hundred and thirty five degree line through the origin. The group generated by N and D possesses eight elements. {N} and {D} denote the respective induced image operators.

For those familiar with the bound matrix representation discussed in Reference 1 and Reference 2, <+>, <x>, <y>, <∧>, </>, {*}, {N}, and {D} are more general versions of the commom image processing operations ADD, MULT, MAX, MIN, DIV, TRAN, NINETY, and FLIP, respectively. It is significant that these operations have been naturally induced from the domain and range structures in a manner that is not specific to image processing.

Generally, no operation can be defined on images unless it employs operations that exist in some sense in either the domain or range spaces. After all, the output of any image operator must be an image, and the operation must therefore, in some manner, affect the domain or range of the input. Thus, any image algebra must ultimately be reduced to operators that have been range or domain induced. How many operators can be induced? The number is only limited by the number of possible operators in the domain and range space. Moreover, for analog images the inducement must proceed with $X = R \times R$ and $Y = R$. For analog signals, $X = R$ and $Y = R$. For quantized and digitized images, $X = Z \times Z$ and $Y = Z$.

3. SUBALGEBRAS

Besides the considerations discussed above, there is also a fundamental mathematical (and hence, structural) consideration that must be taken into account when choosing inducement: Are there important subalgebras that can be embedded in the final image algebra? For instance, the domain space X possesses a Boolean set-theoretic structure. It is likely that an isomorphic copy of this structure should exist in the image algebra. For instance, in two-valued morphological algebra, a fundamental operation is the intersection of two constant images: If $f = 1$ on the set A and $g = 1$ on the set B, we desire an image that is 1 on the intersection of A and B, and is undefined elsewhere. If a minimum has been induced in the algebra, i.e. $\langle\wedge\rangle$, then $f\langle\wedge\rangle g$ gives the desired result. However, a union operation on constant images is also required. This is an extended operation in that the domain may be increased in size. The solution appears to be that a union operation on constant images should be simply induced.

A union operation would have the following characteristic: if $f = 1$ on A and $g = 1$ on B, then the output of the union-type operation must be 1 on the union of A and B. But such an operation cannot be induced directly since inducement only defines operations on intersections. At this point it is possible to make use of precisely that fact. The operation (v) is defined as follows:

$$(f\ (\vee)\ g)(x) = \begin{cases} f(x), \text{ if } x \in A - B \\ f(x) \vee g(x), \text{ if } x \in A \cap B \\ g(x), \text{ if } x \in B - A \end{cases}$$

(v) is induced on the intersection; however, it is defined off the intersection in such a way as to include a desired subalgebra within the image algebra. Indeed, given f and g as above $f\ (\vee)\ g = 1$ on the union of A and B and is undefined elsewhere.

Other extended operations can be defined analogously to (v). For instance, defined are an extended minimum, addition, and multiplication, and denoted by $(\wedge)$, $(+)$, and $(\times)$, respectively. All four of the extended range induced operations play significant roles in the formation of image processing algorithms that take the forms EXTMAX, EXTMIN, EXTADD, and EXTMULT, respectively.

In sum, we must augment inducement by operation extension so as to obtain sufficient structure. Indeed, the ultimate desire is to induce all the structure that exists within the domain and range spaces, including important subalgebras. However, for certain image processing applications, the entire structure is not necessary. An important example of this point is morphological image analysis. Indeed, morphological algebra makes up a basic subalgebra of the overall image algebra. What defines the subject of morphology? The answer is straightforward and algebraically it has very little to do with fitting structuring elements and only a secondary relation to neighborhood transformation, instead it is a form of algebra whose primitive elements generate the subalgebra in question.

4. BASIS

The image algebra is defined by the operations within it. Indeed, by linking operations, new, higher-level operations result through function composition. To wit, it is possible to minimum two images, translate the domain of the output by u and v, and then rotate 90 degrees. The total operation is then given, for inputs f and g, by

$$\{N\}(\{T;\ u,\ v\}(f \langle\wedge\rangle g))$$

The question arises as to the choice of a basis of primitive operators in terms of which all other operators can be expressed as macro-operators. Of course, the basis will depend upon the underlying domain and range spaces, as well as those operations that have been induced. Moreover, the basis will not be unique. What we do desire, however, is a spanning capability sufficient to represent the collection of algorithms under consideration, together with minimality of the basis collection. The first requirement means that all operators in a particular domain are macro-operators relative to the basis, and the second means that no basis operator is a macro-operator made up of the other basis operators. Macro-operators are terms in the algebra, they are formed using finite algebraic closure of the basis operators under function composition. A section on macro-operators follows after a review of the seven fundamental image operations.

With regard to the digital image algebra (DIA), X = Z x Z and Y = R, the table below gives a set of 7 operators that constitute the basis. These seven fundamental operations allow full representation with respect to the field and lattice structures of R, the collection of translations of Z x Z, and the application of the group of symmetries of the square to Z x Z. Moreover, the resulting image algebra possesses isomorphic copies of the Boolean set-theoretic algebra, the bound matrix algebra, and several other algebras.

TABLE 1 - RANGE AND DOMAIN OPERATIONS

RANGE OPERATION	DIA OPERATION
+	<+>
x	<x>
v	(v)
/	</>
DOMAIN OPERATION	**DIA OPERATION**
N	{N}
D	{D}
T	{T}

5. SEVEN FUNDAMENTAL IMAGE OPERATIONS

Commuting diagrams illustrating the nature of the inducement process along with examples of each of the seven basic operations will now be given. The domain of the images f and g in the ensuing discussion are A and B respectively. These sets are subsets of ZxZ and the image space of f and g are the reals R. In subsequent commuting diagrams functions with restricted domains are labeled using the unrestricted function.

The fundamental operation of image addition is defined only on the intersection of the domains of the two images and there it is given by:

$$(f\langle+\rangle g)(i,j) = f(i,j) + g(i,j) \text{ for all } (i,j) \in A \cap B$$

A commuting diagram illustrating this situation is

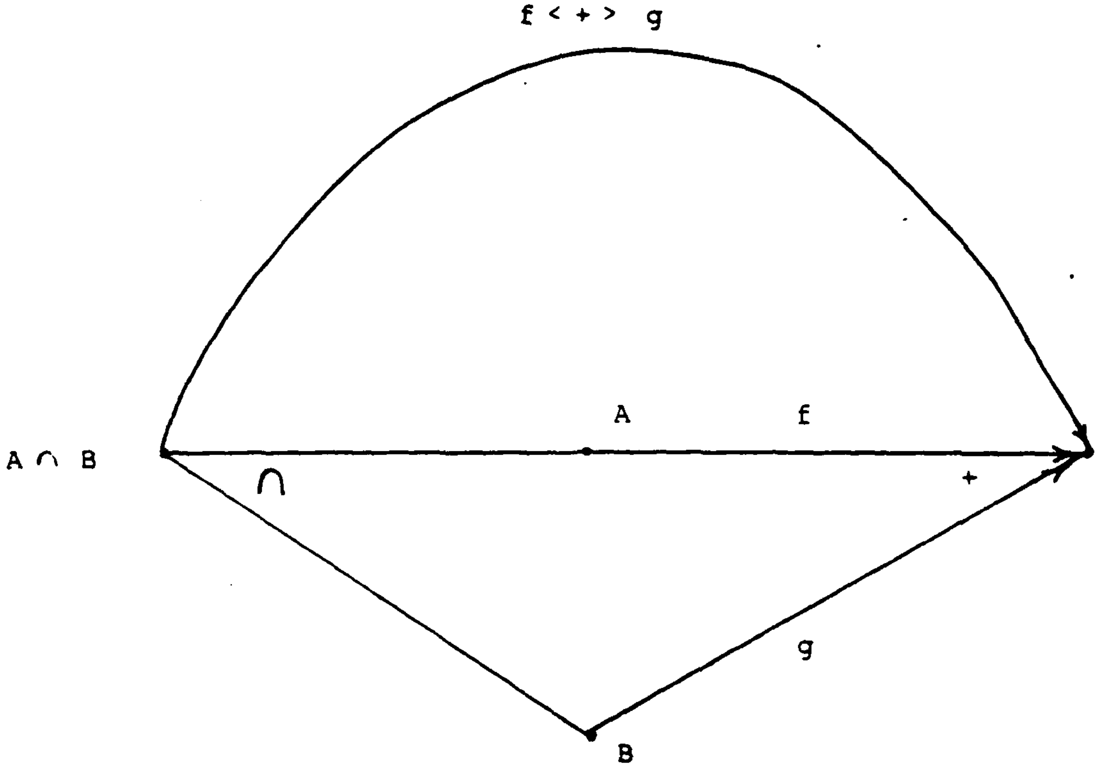

Figure 1 - Range Induced Addition Operation

An illustration of the use of $\langle + \rangle$ is provided utilizing bound matrix representation for images f and g.

Thus, if $$f = \begin{vmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{vmatrix}_{0,0}$$

and $$g = \begin{vmatrix} 3 & -2 & 4 \\ 0 & 2 & 5 \end{vmatrix}_{0,0}$$

then $$f\langle + \rangle g = \begin{vmatrix} 4 & 0 & * \\ 3 & 6 & * \\ * & * & * \end{vmatrix}_{0,0}$$

where * is a symbol to denote "undefined" and matrix subscripts are used to locate the position the first term in the matrix in the integral lattice. All other entries in the matrix have relative locations in the lattice equivalent to their relative positions in the matrix.

Similar to the basic image addition operation, the basic operation of image multiplication is defined only on the intersection of the domain of the given images. Indeed,

$$(f\langle x \rangle g)(i,j) = f(i,j) \times g(i,j) \text{ for all } (i,j) \in A \cap B$$

It follows that the commuting diagram illustrating this operation is similar to the one for image addition.

Using the images f and g previously represented by bound matrices the image multiplication operation results in the image:

$$(f \langle x \rangle g) = \begin{vmatrix} 3 & -4 & * \\ 0 & 8 & * \\ * & * & * \end{vmatrix}_{0,0}$$

Unlike the image addition and multiplication operation the basic operation of image maximum results in an image with domain equal to the union of the domains of the original images. Here,

$$(f(v)g)(i,j) = \begin{cases} f(i,j)vg(i,j) & (i,j) \varepsilon \ A \cap B \\ f(i,j) & (i,j) \varepsilon \ A - B \\ g(i,j) & (i,j) \varepsilon \ B - A \end{cases}$$

The commuting diagram illustrated below can be used in representing this operation. Thus,

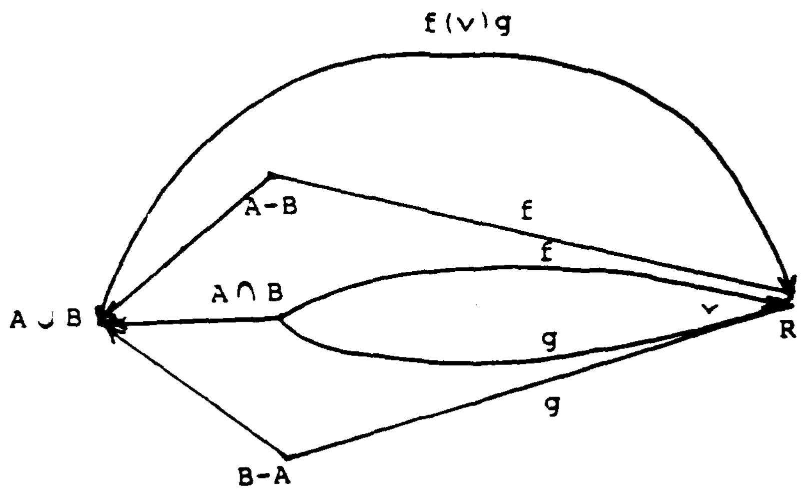

Figure 2 - Range Induced Maximum Operation

The fundamental image maximum operation when applied to f and g as previously given results in the image

$$f \; (v) \; g = \begin{vmatrix} 3 & 2 & 4 \\ 3 & 4 & 5 \\ 5 & 6 & * \end{vmatrix}_{0,0}$$

Image division when applied to f and g results in an image whose domain is some subset of $A \cap B$. Specifically,

$$(f</>g)(i,j) = f(i,j)+g(i,j), \text{ if } (i,j) \in A \cap B \text{ and } g(i,j) \neq 0$$

In terms of commuting diagrams if

$B' = B - \{(i,j) : g(i,j) = 0\}$ then we have

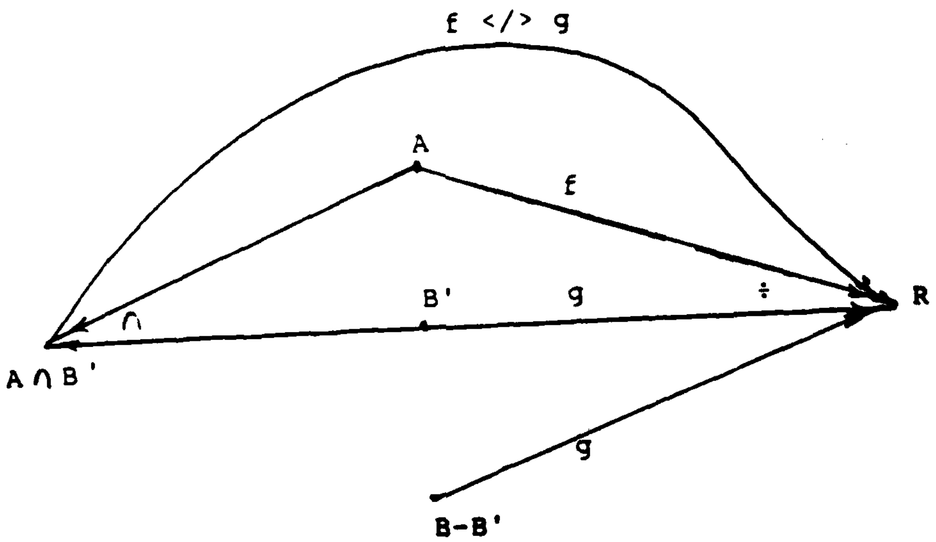

Figure 3 - Range Induced Division Operation

Using f and g as previously defined it follows that

$$f <\!/\!> g = \begin{vmatrix} 1/3 & -1 & * \\ * & 2 & * \\ * & * & * \end{vmatrix}_{0,0}$$

Recall that the last three fundamental operations are domain induced from corresponding one to one onto functions on ZxZ. The fundamental rotation operation $\{N\}$ is such that

$$(\{N\}f)(i,j) = f(j,-i)$$

It is induced from, the 90 degree rotation operation $\{N\}$ which is such that $\{N\}(i,j) = (-j,i)$.

The commuting diagram given next illustrates this operation. In this diagram N(A) is the range of the domain of f provided by the pointwise (counterclockwise) 90 degree rotation N.

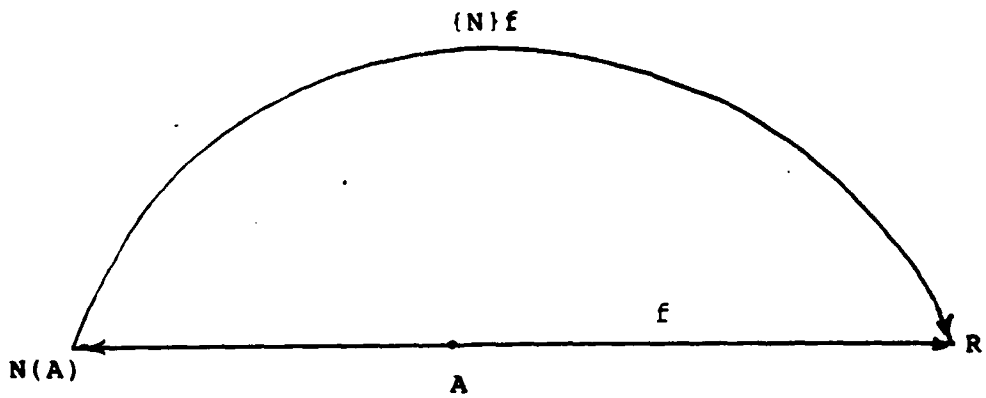

Figure 4 - Domain Induced 90 Degree Rotation Operation

This diagram clearly shows the inducement process is:

$$(\{N\}f)\ (i,j) = f(\{N\}^{-1}(i,j))$$

Using the image f previously given we have

$$\{N\}f = \begin{vmatrix} 2 & 4 & 6 \\ 1 & 3 & 5 \end{vmatrix}_{0,1}$$

Notice that the grey value 1 is the only fixed value since it occurs at the origin. All other grey values are located 90 degrees counterclockwise away from their original location in image f.

The fundamental flip (for reflection) operation {D} is defined by

$$(\{D\}f)\ (i,j) = f(-j,-i)$$

The commuting diagram describing the domain inducement involved in defining this operation follows:

$$\begin{tikzcd} D(A) \arrow[rr, bend left, "\{D\}f"] & A \arrow[l] \arrow[r, "f"] & R \end{tikzcd}$$

Figure 5 - Domain Induced Flip Operation

The flip operation is induced by the idempotent pointwise operation D: $Z \times Z \rightarrow Z \times Z$ where $D(i,j) = (-j\ -i)$. In the above diagram D(A) is the range of the domain of f obtained by utilizing D. If the bound matrix f previously defined is employed utilizing the flip operation we have, earlier f was defined as

$$f = \left| \begin{array}{cc} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{array} \right|_{0,0}$$

Then

$$\{D\}f = \left| \begin{array}{ccc} 1 & 3 & 5 \\ 2 & 4 & 6 \end{array} \right|_{0,0}$$

Notice the similarity of this operation with that of matrix transposition. Indeed, the image flip operation can be described in terms of transpose and the image translation operation which is described in detail next.

The translation operation is domain induced from the successor operation T on the integers. Here

$$T: Z \times Z \rightarrow Z \times Z \quad \text{where}$$

$$T\ (i,j) = (i+1,j)$$

The corresponding domain induced operation defined in the usual way gives

$$(\{T\}f)\ (i,j) = f(T^{-1}(i,j)) = f(i-1,j)$$

The commuting diagram illustrates the inducement process:

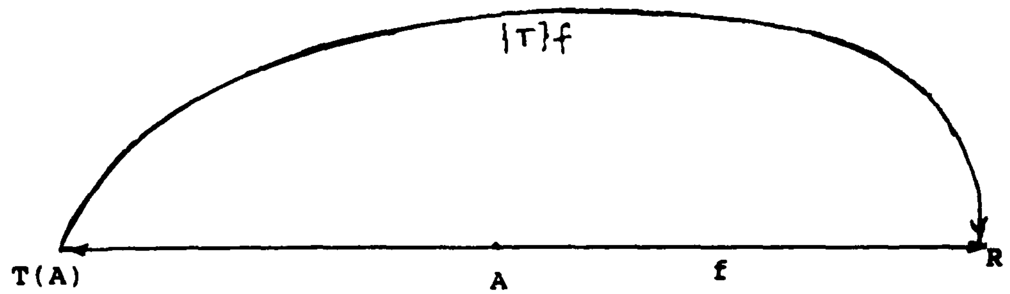

Figure 6 - Domain Induced Translation Operation

Thus for f previously given, we have

$$\{T\}f = \begin{vmatrix} * & 1 & 2 \\ * & 3 & 4 \\ * & 5 & 6 \end{vmatrix}_{0,0} = \begin{vmatrix} 1 & 2 \\ 2 & 4 \\ 5 & 6 \end{vmatrix}_{1,0}$$

6. MACRO-OPERATORS

Now that a collection of fundamental operators has been selected for the Digital Image Algebra (DIA), a library of macro-operators can be given to demonstrate the spanning capability of the basic seven. Among the macro-operator collection to be presented will be low-level mathematical operators, as well as popular image-processing operators such as dilation and convolution. The operators will be presented in the form of a list, with the operator's name, mathematical definition, and basis representation. With regard to the latter, the representations of the higher level operators will employ lower level macro-operators that have already been expressed utilizing the basis primatives.

Throughout the development of the macro-operator or term library, two isomorphisms, I1 of the Reals (R), with the set of images defined only at the origin and I2 the collection of subsets of Z x Z with the set of images having constant value 0, are defined. The constant image defined on the set A with value k will be denoted by k_A. In particular, k_0 will denote the image possessing the single grey value k at the origin. Moreover, if the following assumption is made: given the constant image 0_A, with A finite, the elements of A can be employed as an index set. A list of fundamental macro-operators similar to those defined in the Phase I report, follows. Where indicated, proofs showing that these operations are indeed terms within the algebra are given in the examples which are in the following section.

a. Domain Extractor. Given an image f defined on A, $K(f)=0_A$. Using the identification I2, K(f) = A.
Basis Representation: $K(f) = 0_{ZxZ} \langle x \rangle f$

26

b. Existential Operator. Given a set 0_A and a real number k_0, $E(k_0, 0_A) = k_A$. Because of the identifications I1 and I2, $E(k, A) = k_A$.
Basis Representation: $E(k, A) = k_{ZxZ} \langle + \rangle 0_A$

c. Additive Inverse. $-f$
Basis Representation: $-f = E(-1, K(f))\langle x \rangle f$

d. Extended Minimum. $f(\wedge)g$
Basis Representation: $f(\wedge)g = -\{(-f)(v)(-g)\}$

e. Extended Addition. $f(+)g$
Basis Representation:
$f(+)g=(f\langle + \rangle g)(v)\{(G\langle / \rangle G)\langle x \rangle f\}(v)\{(H\langle / \rangle H)\langle x \rangle g\}$ where
$G = E\{1,K(f)\}(x)E\{0,K(g)\}$, $H = E\{1,K(g)\}(x)E\{0,K(f)\}$

f. Extended Multiplication f (x) g
Basis Representation: Same as f (+) g, except employ f<x>g

g. Directly Induced Maximum. f <v> g
Basis Representation:
f <v> g = (f(v)g) <+> K(f) <+> K(g)

h. Directly Induced Minimum. f <Λ> g
Basis Representation:
f <Λ> g = (f(Λ)g) <+> K(f) <+> K(g)

i. Scalar Multiplication: Given an image f and a real number k, kf denotes the usual scalar multiplication of an image by a scalar. In fact, there are two images involved since k is actually k_0.
Basis Representation: kf = E{k, K(f)} <x> f

j. Absolute Value. |f|
Basis Representation: |f| = f <v> (-f)

k. Complementation. For the constant image 0_A, we define the complement image $C(0_A) = 0_{A}c$.
Basis Representation: $C(0_A)=0_{ZxZ}$ </> $(1_{ZxZ}(+)E(-1,A))$

l. Thresholding. A common image processing operation is thresholding. Given an image f and a scalar k, the thresholded image $T_k(f)$ is defined by

$$\{T_k\ (f)\}(i,j) = \begin{cases} 1, & \text{if } f(i,\ j) \geq k \\ 0, & \text{if } f(i,\ j) < k \end{cases}$$

Basis Representation: Let g = f <+> E{-k, K(f)}. Then
$T_k(f)$ = (-{(g<Λ>E{0,K(g)})</>(g(Λ)E{0,K(g)})})(+)E(1,K(g))

m. Selection. Given an image f and a set B, the selection operator S returns f restricted to the set B.
Basis Representation: S(f,B) = f <+> 0_B

n. Restriction. Given images f and g, R(f,g) is restricted to the domain of g.
Basis Representation: R(f,g) = S(f, K(g))

o. Extension. Given a primary image f and a secondary image g, the extension of f by g, H(f,g), is equal to f on K(f) and g on K(g) - K(f).
Basis Representation: H(f,g) = f (+) (E{0, K(f)} (x) g)

p. Translation. Perhaps the single most important operation in image processing algorithms is translation. The basis operator {T} yields a unit translation in the positive x direction. Consequently, the concatenation of {T} n times, $\{T\}^n$, yields a translation of n units in the positive x direction. More generally, we require a representation of the macro-operator {T; u,v}, which produces a translation of u units in the x direction and v units in the y direction.
Basis Representation:

Positive Y translation	$\{T;\ 0,\ 1\} = \{N\}^3\ \{T\}\{N\}$
Negative X translation	$\{T;\ -1,0\} = \{N\}^2\ \{T\}\{N\}^2$
Negative Y translation	$\{T;\ 0,\ 1\} = \{N\}\{T\}\{N\}^3$

q. Offset. Given a real number k and an image f, the offset of f by k, f + k, is the image obtained from f by adding k to each grey value.
Basis Representation: f + k = E{k, K(f)} <+> f

r. Minkowski Addition (Dilation). Given two images f and g possessing finite domains, the grey-scale dilation of f and g is defined by:

$$f \circledcirc g = \mathrm{EXTMAX}_{(i,j)\ \varepsilon\ \mathrm{DOMAIN}(g)}(\mathrm{TRAN}(f{:}i,j)+g(i,j))$$

(see Reference 3). Since we have identified the subsets of ZxZ with the 0-valued constant images, the set-theoretic Minkowski addition is given by the same definition. Basis Representation: Since EXTMAX, TRAN, and DOMAIN are bound matrix versions of (v), {T;_,_}, and K, respectively, the definition is, in essence, a basis representation. Specifically,

$$f \circledcirc g = (v)_{(i,j)\ \varepsilon\ K(g)}\{\{T;i,j\}(f)+g(i,j)\}$$

s. Minkowski Subtraction. Given two images f and g with finite domains, the grey-scale Minkowski subtraction is defined by

$$f \ominus g = (\wedge)\mathrm{MIN}_{(i,j)\varepsilon\ \mathrm{DOMAIN}(g)}\{\mathrm{TRAN}(f;i,j)+g(i,j)\}$$

(see Reference 3). As with Minkowski addition, the corresponding set-theoretic erosion is given by the same definition.
Basis Representation:

$$f \ominus g = \langle\wedge\rangle_{(i,j)\varepsilon\ K(g)}\{\{T;i,j\}(f)+g(i,j)\}$$

t. Dot Product. Given two images f and g possessing a common finite domain, i.e. A, f · g is defined as the usual sum-of-products dot product. Should the images not possess a common domain, the dot product is undefined. Basis Representation:

$$f \cdot g = \langle + \rangle_{(i,j)\ \varepsilon\ \ K(f)} \{T;\ -i,\ -j\}(f\ \langle x \rangle\ g)$$

u. Correlation. The correlation of two images f and g posessing finite domains is given pointwise by

$$(f \bullet g)(i,j) = \ {}_{\substack{(p,q)\ \ K(f)\\ (p-i,q-j)\ \ \ K(g)}} f(p,q)g(p-i,q-j)$$

Basis Representation:

$$f \bullet g = \ {}_{(r,s)\ \varepsilon\ (+)\ \ K(g)}\ g(r,s)\{T;\ -r,-s\}(f)$$

where, rigorously, under the identification I1,

$$g(r,s) = S(\{T;\ -r,\ -s\}(g),\{(0,0)\})$$

v. Moving-Average Filter. The usual moving-average filter, as defined in Reference 1 is essentially a convolution: however, there are two conventions (which guarantee that the domain of the filtered image is given by the morphological opening of the input domain by the mask domain): (1) The second input g is a mask, which simply means it is defined at the origin. (2) If the domain of the translated mask {T; i,j}(g) is not a subset of the primary input f, then the filter is undefined at (i,j). While the first convention does not affect the basis representation, the second does. Instead of (+), <+> is employed. The notation F(f,g) denotes the output of the filter.

w. Image Vector Norms. Given a vector of images, say

$$\vec{f} = (f, f_2, \ldots, f_m)$$

various norms on $\vec{f}$ are defined (see Reference 1). For the sake of simplicity, consider only the supremum norm, and we will suppose that the f_j possess a common domain. Then

$$||\vec{f}||_\infty (i,j) = \max\{|f_1(i,j)|, |f_2(i,j), \ldots, |f_m(i,j)|\}$$

Basis Representation:

$$||\vec{f}||_\infty = \underset{j=1}{\overset{m}{\langle v \rangle}} \quad |f|$$

x. Gradient-Type Edge Detector. Various edge detection algorithms employ digital image gradients, such as the Prewitt and Sobel gradients. An edge image is produced by filtering by two gradient masks, applying a norm to the resulting image vector and then thresholding. Restricting our attention to the supremum norm, a basis representation for the entire procedure is given by

$$E(f,g,h,t) = T_t(\ ||(F(f,g),F(f,h)||_\infty \ )$$

where f is the input image, g and h are directional mask images and t is the threshold value.

7. MORPHOLOGICAL BASIS

As noted previously, the morphological algebra is a subalgebra of the image algebra (assuming the images under consideration possess finite domains). As such, it is defined precisely by the choice of its basis elements. Specifically, we define the morphological basis to the collection of operators

$$M = \{(N),\ T,\ \langle\wedge\rangle,\ (\vee)\}$$

From the operators in M, it is possible to construct the two-valued dilation and erosion. Keep in mind that two-valued morphology involves operations on subsets of Z x Z; of the form 0 {A}. Noting that {T; u, v} is a macro-operator relative to {N} and {T}, and hence relative to M, we have the Minkowski addition (dilation)

$$0\{A\} \oplus 0\{B\} = \underset{(i,j)\,\varepsilon\,0\{B\}}{(\vee)} \{T;\ i,j\}(0\{A\})$$

the Minkowski subtraction

$$0\{A\} \ominus 0\{B\} = \underset{(i,j)\,\varepsilon\,0\{B\}}{\langle\wedge\rangle} \{T;\ i,j\}(0\{A\})$$

and the erosion

$$\text{ERODE}(0\{A\},\ 0\{B\}) = 0\{A\} \ominus \{N\}^2\ (0\{B\})$$

For the grey-scale morphological algebra, we employ the grey-scale morphological basis:

$$M' = M \cup \{+, -\}$$

where + is the offset operator (macro letter q.) and - is the additive inverse operator (macro letter c.). Although $\langle\wedge\rangle$, +, and - are macro-operators relative to the full basis, they are much simpler than the operators from which they are derived. It is therefore necessary to demonstrate that the grey-scale dilation and erosion are macro-operators relative to M.

First note the domain extractor K (macro letter a.) can be constructed from the operators of M:

$$K(f) = 0_{Z \times Z} \langle\wedge\rangle (f \ (v) \ (-f))$$

In addition the "reflection" macro-operator J, reflects an image through the origin: $\{J(f)\}(i,j)=f(-i,-j)$. The basis representation is:

$$J(f) = \{N\}^2(-f)$$

Referring to the original basis representations for Minkowski addition (macro letter r.) and Minkowski subtraction (macro letter s.), it can be seen that both representations only employ operators from M. Moreover, erosion is given by

$$\text{ERODE}(f,g) = f \ \ominus \ J(g)$$

Two points should be noted. The two-valued representations are special cases of the grey-scale representations. Moreover, in both settings, the opening and closing are immediately expressible in terms of Minkowski addition and subtraction.

$$\text{OPEN}(f,g) = (f \ \ominus \ J(g)) \ \oplus \ g$$

$$CLOSE(f,g) = - OPEN(-f, -g)$$

In the sum, for images possessing finite domains, morphological algebra is a subalgebra of the overall digital image algebra. The importance of recognizing its basis is that the basis determines the extent of the subalgebra, and thus its domain of applicability.

SECTION III

MACRO OPERATIONS REPRESENTATIONS

In order to demonstrate the operational nature of the 24 macro-operator representations given earlier, a walk-through using sample images is provided throughout the section. The lettering scheme will correspond to the lettering of the macro-operator representation from Section II, paragraph 6. Proof of several operations are also provided at the end of the appropriate examples.

a. Domain Extractor K. Let

$$f = \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2}$$

According to the basis representation of K,

$$K(f) = 0_{ZxZ} \langle x \rangle f = \begin{vmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & * \end{vmatrix}_{0,2}$$

which, under the identification of zero-images and subsets of ZxZ is precisely the domain of f. Note that the induced multiplication operator <x> only yields an output on the intersection of the domains, and since the domain of f is a subset of the domain of 0_{ZxZ}, the resulting image is defined only on the domain of f.

b. Existential Operator E. Let

$$A = \{(0, 0), (0,1), (1,0), (1,1)\}$$

Then under the identification I2,consider the image

$$0_A = \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix}_{0,1}$$

Suppose the constant k = 2. Under the identification I1,

$$k = (2)_{0,0}$$

According to the basis representation of E,

$$E(k, A) = E(2_0, 0_A) = 2_{ZxZ} \langle+\rangle 0_A$$

$$= 2_{ZxZ} \langle+\rangle \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix}_{0,\ 1}$$

$$= \begin{vmatrix} 2 & 2 \\ 2 & 2 \end{vmatrix}_{0,1}$$

As in the preceding representation for K, the fact that A is a subset of ZxZ results in the output image having domain A.

c. Additive Inverse –. Let f be as in (a.). Then, according to the basis representation and the identifications I1 and I2.

$$-f = E\{-1, K(f)\} \langle x\rangle f$$

$$= \begin{vmatrix} -1 & -1 & -1 \\ -1 & -1 & -1 \\ -1 & -1 & * \end{vmatrix}_{0,2} \langle x\rangle \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2}$$

$$= \begin{vmatrix} -2 & -3 & -1 \\ 0 & 1 & -2 \\ -1 & -4 & * \end{vmatrix}_{0,2}$$

d. **Extendend Minimum ($\wedge$).** Let f be the image given in example a and let

$$g = \begin{vmatrix} 2 & 1 \\ 0 & 1 \\ 2 & 3 \\ 5 & 1 \end{vmatrix}_{0,3}$$

According to the basis representation,

$$f \; (\wedge) \; g = -\{(-f) \; (v) \; (-g)\}$$

$$= - \left[\begin{vmatrix} -2 & -3 & -1 \\ 0 & 1 & -2 \\ -1 & -4 & * \end{vmatrix}_{0,2} (v) \begin{vmatrix} -2 & -1 \\ 0 & -1 \\ -2 & -3 \\ -5 & -1 \end{vmatrix}_{0,3} \right]$$

$$= - \begin{vmatrix} -2 & -1 & * \\ 0 & -1 & -1 \\ 0 & 1 & -2 \\ -1 & -1 & * \end{vmatrix}_{0,3} = \begin{vmatrix} 2 & 1 & * \\ 0 & 1 & 1 \\ 0 & -1 & 2 \\ 1 & 1 & * \end{vmatrix}_{0,3}$$

e. **Extended Addition (+).** Let f and g be as given previously. Since the basis representation of (+) is rather long, we will break the present illustration into pieces as shown in Section IIb, paragraph e and then put it all together to illustrate extended addition. Let

$$F = f \; \langle + \rangle \; g = \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2} \langle + \rangle \begin{vmatrix} 2 & 1 \\ 0 & 1 \\ 0 & 1 \\ 2 & 3 \\ 5 & 1 \end{vmatrix}_{0,3}$$

$$= \begin{vmatrix} 2 & 4 \\ 2 & 2 \\ 6 & 5 \end{vmatrix}_{0,2}$$

$$G = E\{1,\ K(f)\}\ (\wedge)\ E\{0,\ K(g)\}$$

$$= \left|\begin{matrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & * \end{matrix}\right|_{0,2} \langle \wedge \rangle \left|\begin{matrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{matrix}\right|_{0,3}$$

$$= \left|\begin{matrix} 0 & 0 & * \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & * \end{matrix}\right|_{0,3}$$

$$H = E\{1,\ K(g)\}\ (\wedge)\ E\{0,\ K(f)\}$$

$$\left|\begin{matrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{matrix}\right|_{0,3} \quad (\wedge) \quad \left|\begin{matrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & * \end{matrix}\right|_{0,2}$$

$$= \left|\begin{matrix} 1 & 1 & * \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & * \end{matrix}\right|_{0,3}$$

Then

$$(G \;</>\; G)\ \langle x \rangle f = \left|\begin{matrix} * & * & 1 \\ * & * & 1 \end{matrix}\right|_{0,2} \langle x \rangle \left|\begin{matrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{matrix}\right|_{0,2}$$

$$= \left|\begin{matrix} * & * & 1 \\ * & * & 2 \end{matrix}\right|_{0,2} = \left|\begin{matrix} 1 \\ 2 \end{matrix}\right|_{2,2}$$

$$(H \;</>\; H)\ \langle x \rangle\ g = (\ 1\ \ 1\ )_{0,3} \quad \langle x \rangle \left|\begin{matrix} 2 & 1 \\ 0 & 1 \\ 2 & 3 \\ 5 & 1 \end{matrix}\right|_{0,3}$$

$$= (\ 2 \quad 1\ )_{0,\ 3}$$

Finally, according to the basis representation, the extended addition is given by:

$$f\ (+)\ g = F\ (v)\ \{(G\ </>\ G)\ <x>\ f\}\ (v)\ \{(H\ </>\ H)\ <x>\ g\}$$

$$= \left|\begin{matrix} 2 & 4 \\ 2 & 2 \\ 6 & 5 \end{matrix}\right|_{0,2} \quad (v) \quad \left|\begin{matrix} 1 \\ 2 \end{matrix}\right|_{2,2} \quad (v)\ \left(\begin{matrix} 2 & 1 \end{matrix}\right)_{0,3}$$

$$= \left|\begin{matrix} 2 & 1 & * \\ 2 & 4 & 1 \\ 2 & 2 & 2 \\ 6 & 5 & * \end{matrix}\right|_{0,3}$$

f. Extended Multiplication (x). The operator (x) possesses the same basis representation as <+> except that we employ f (x) g in place of f (+) g. Thus, utilizing the same f and g definitions as in the previous illustration, we let

$$F = f\ <x>\ g = \left|\begin{matrix} 0 & 3 \\ 0 & -3 \\ 5 & 4 \end{matrix}\right|_{0,2}$$

Using G and H from the preceding case, the extended addition is given by:

$$f\ (x)\ g = F\ (v)\ \{(G\ </>\ G)\ <x>\ f\}\ (v>\ \{(H\ </>\ H)\ <x>\ g\}$$

$$= \left|\begin{matrix} 0 & 3 \\ 0 & -3 \\ 5 & 4 \end{matrix}\right|_{0,2} \quad (v) \quad \left|\begin{matrix} 1 \\ 2 \end{matrix}\right|_{2,2} \quad (v) \quad \left(\begin{matrix} 2 & 1 \end{matrix}\right)_{0,3}$$

$$= \left|\begin{matrix} 2 & 1 & * \\ 0 & 3 & 1 \\ 0 & -3 & 2 \\ 5 & 4 & * \end{matrix}\right|_{0,3}$$

g. Directly Induced Maximum $\langle \vee \rangle$. According to the basis representation, if we let f and g be as previously, then

$$f \langle \vee \rangle g = (f \ (\vee) \ g) \langle + \rangle \{K(f) \langle + \rangle K(g)\}$$

$$= \left| \begin{matrix} 2 & 1 & * \\ 2 & 3 & 1 \\ 2 & 3 & 2 \\ 5 & 4 & * \end{matrix} \right|_{0,3}$$

$$\langle + \rangle \left[\left| \begin{matrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & * \end{matrix} \right|_{0,2} \langle + \rangle \left| \begin{matrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{matrix} \right|_{0,3} \right]$$

$$= \left| \begin{matrix} 2 & 1 & * \\ 2 & 3 & 1 \\ 2 & 3 & 2 \\ 5 & 4 & * \end{matrix} \right|_{0,3} \langle + \rangle \left| \begin{matrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{matrix} \right|_{0,2}$$

$$= \left| \begin{matrix} 2 & 3 \\ 2 & 3 \\ 5 & 4 \end{matrix} \right|_{0,2}$$

h. Directly Induced Minimum $\langle \wedge \rangle$. Because of the similarity to $\langle \vee \rangle$, we will omit an example in this case.

i. Scalar Multiplication $\cdot$. Let f be as above and $k = 2$. Then the basis representation gives

$$2f = 2 \cdot f = E\{2, K(f)\} \langle x \rangle f$$

$$= \begin{vmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & * \end{vmatrix}_{0,2} \langle x \rangle \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2}$$

$$= \begin{vmatrix} 4 & 6 & 2 \\ 0 & -2 & 4 \\ 2 & 8 & * \end{vmatrix}_{0,2}$$

j. Absolute Value || . Again using f, according to the basis representation,

$$|f| = f \langle v \rangle (-f)$$

$$= \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2} \langle v \rangle \begin{vmatrix} -2 & -3 & -1 \\ 0 & 1 & -2 \\ -1 & -4 & * \end{vmatrix}_{0,2}$$

$$= \begin{vmatrix} 2 & 3 & 1 \\ 0 & 1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2}$$

k. Complementation C. Let

$$0_A = \begin{vmatrix} 0 & 0 \\ 0 & 0 \end{vmatrix}_{0,1}$$

Applying the basis representation for complementation and using complementary bound matrix notation

$$C(0_A) = 0_{ZxZ} \langle / \rangle (1_{ZxZ} \ (+) \ E\{-1, A\})$$

$$= 0:_{ZxZ} \langle/\rangle (1:_{ZxZ} \ (+) \begin{vmatrix} -1 & -1 \\ -1 & -1 \end{vmatrix}_{0,1}) = \begin{vmatrix} * & * \\ * & * \end{vmatrix}^{c}_{0,1}$$

Note that 1_{ZxZ} (+) E{-1, A} yields an image which is 0 on A and -1 elsewhere. Hence, when we divide 0_{ZxZ} by this image, all pixels in A have undefined gray value, whereas those pixels outside of A still have value 0. The complementary bound matrix notation is used to denote the latter occurrence.

1. Thresholding. To demonstrate the threshold representation we let f be as previously and k = 2. First, utilizing the notation in the original specification,

$$g = f \langle + \rangle E\{-2, K(f)\}$$

$$= \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2} \langle + \rangle \begin{vmatrix} -2 & -2 & -2 \\ -2 & -2 & -2 \\ -2 & -2 & * \end{vmatrix}_{0,2}$$

$$= \begin{vmatrix} 0 & 1 & -1 \\ -2 & -3 & 0 \\ -1 & 2 & * \end{vmatrix}_{0,2}$$

Let

$$G = g \langle \wedge \rangle E\{0, K(g)\}$$

$$= g \langle \wedge \rangle \begin{vmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{vmatrix}_{0,2} = \begin{vmatrix} 0 & 0 & -1 \\ -2 & -3 & 0 \end{vmatrix}_{0,2}$$

According to the basis representation,

$$T_2 (f) = (-\{G \langle/\rangle G\}) \ (+) \ E\{1, K(g)\}$$

$$= \begin{vmatrix} * & * & -1 \\ -1 & -1 & * \\ -1 & * & * \end{vmatrix}_{0,2} (+) \begin{vmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & * \end{vmatrix}_{0,2}$$

$$= \begin{vmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & * \end{vmatrix}_{0,2}$$

Proof of term representation.

$h = -\{(g \langle\wedge\rangle E\{0,K(g)\}) \langle/\rangle (g \langle\wedge\rangle E\{0,K(g)\})\}(i,j)$

Then $h(i, j) = -1$ if and only if $g(i, j) < 0$; otherwise it is undefined. Thus,

$$(h \ (+) \ E(1, K(g))\} = \begin{cases} 0, & \text{if } g(i, j) < 0 \\ 1, & \text{if } g(i, j) \geq 0 \end{cases}$$

But $g(i,j) = f(i, j) - k$

m. Selection S. Let f be as above and let

$$B = \{(0, 0), (0, 1), (0, 2), (0, 3)\}$$

then, using the basis representation

$$S(f, B) = f \langle+\rangle 0_B$$

$$= \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2} \langle+\rangle \begin{vmatrix} 0 \\ 0 \\ 0 \\ 0 \end{vmatrix}_{0,3} = \begin{vmatrix} 2 \\ 0 \\ 1 \end{vmatrix}_{0,2}$$

n. Restriction R. Let f and g be as in previous illustrations. Using the basis representation,

$$R(f, g) = S(f, K(g))$$

$$= S\left\{ \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2} \begin{vmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{vmatrix}_{0,3} \right\} = \begin{vmatrix} 2 & 3 \\ 0 & -1 \\ 1 & 4 \end{vmatrix}_{0,2}$$

o. Extension H. Let f and g be as in the preceding representation. Then

$$E\{0, K(f)\} \ (x) \ g = \left|\begin{matrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & * \end{matrix}\right|_{0,2} (x) \left|\begin{matrix} 2 & 1 \\ 0 & 1 \\ 2 & 3 \\ 5 & 1 \end{matrix}\right|_{0,4}$$

$$= \left|\begin{matrix} 2 & 1 & * \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & * \end{matrix}\right|_{0,3}$$

Thus, the basis representation for H yields

$$H(f, g) = f \ (+) \ (E\{0, K(f)\} \ (x) \ g)$$

$$= \left|\begin{matrix} 2 & 1 & * \\ 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{matrix}\right|_{0,2}$$

p. Translation {T}. {T; 1, 0} is in the basis. To demonstrate the basis representations of the other three unit translations, we shall employ the image:

$$f = \left|\begin{matrix} * & * & 3 & 2 \\ * & * & * & * \end{matrix}\right|_{0,1}$$

where we have deliberately not employed a minimal representation of the image so that the bottom-left pixel, which is star-valued and located at the origin, will remain within the frame of the matrix, the reason being to better illustrate the translation and rotation effects. The first representation is

$$\{T;\ 0,\ -1\}(f) = \{N\}^3\{T\}\{N\}(f)$$

$$= \{N\}^3\ \{T\} \begin{vmatrix} 2 & \ast \\ 3 & \ast \\ \ast & \ast \\ \ast & \ast \end{vmatrix}_{-1,3}$$

$$= \{N\}^3 \begin{vmatrix} 2 \\ 3 \\ \ast \\ \ast \end{vmatrix}_{0,3}$$

$$= \{N\}^2 \begin{pmatrix} 2 & 3 & \ast & \ast \end{pmatrix}_{-3,0}$$

$$= \{N\} \begin{vmatrix} \ast \\ \ast \\ 3 \\ 2 \end{vmatrix}_{0,0}$$

$$= \begin{pmatrix} \ast & \ast & 3 & 2 \end{pmatrix}_{0,0}$$

The second representation is

$$\{T;\ -1,\ 0\}(f) = \{N\}^2\ \{T\}\{N\}^2\ (f)$$

$$= \{N\}\{T\}^2\{N\} \begin{vmatrix} 2 & \ast \\ 3 & \ast \\ \ast & \ast \\ \ast & \ast \end{vmatrix}_{-1,3}$$

$$= \{N\}^2\{T\} \begin{vmatrix} \ast & \ast & \ast & \ast \\ 2 & 3 & \ast & \ast \end{vmatrix}_{-3,0}$$

$$= \{N\}^2 \begin{vmatrix} \ast & \ast & \ast \\ 2 & 3 & \ast \end{vmatrix}_{-2,0}$$

$$= \{N\} \begin{vmatrix} \ast & \ast \\ \ast & 3 \\ \ast & 2 \end{vmatrix}_{0,0}$$

$$= \begin{vmatrix} * & 3 & 2 \\ * & * & * \end{vmatrix}_{0,1}$$

The third translation representation is illustrated by

$$\{T: 0, 1\}(f) = \{N\}\{T\}\{N\}^3 (f)$$

$$= \{N\}\{T\}\{N\}^2 \begin{vmatrix} 2 & * \\ 3 & * \\ * & * \\ * & * \end{vmatrix}_{-1,3}$$

$$= \{N\}\{T\}\{N\} \begin{vmatrix} * & * & * & * \\ 2 & 3 & * & * \end{vmatrix}_{-3,0}$$

$$= \{N\}\{T\} \begin{vmatrix} * & * \\ * & * \\ * & 3 \\ * & 2 \end{vmatrix}_{0,0}$$

$$= \{N\} \begin{vmatrix} * & * & * \\ * & * & * \\ * & * & 3 \\ * & * & 2 \end{vmatrix}_{0,0}$$

$$= \begin{vmatrix} * & * & 3 & 2 \\ * & * & * & * \\ * & * & * & * \end{vmatrix}_{0,2}$$

Proof of term representation:

$$\begin{aligned} \{N\}^3 \{T\}\{N\} (f)(i,j) &= \{N\}^3\{T\}(f)(j,-i) \\ &= \{N\}^3(f)(j - 1, -i) \\ &= \{N\}^2(f)(-i + 1) \\ &= \{N\}(f)(-j+1,i) \\ &= f(i,j -1) \\ &= \{T;0,1\}(f)(i,j) \end{aligned}$$

$$
\begin{aligned}
\{N\}^2\{T\}\{N\}^2(f)(i,j) &= \{N\}^2\ \{T\}(f)(-i,-j) \\
&= \{N\}^2(f)(-i-1,\ -j) \\
&= f(i+1,\ j) \\
&= \{T;\ -1,\ 0\}(f)(i,j)
\end{aligned}
$$

$$
\begin{aligned}
\{N\}\{T\}\{N\}^3(f)(i,j) &= \{N\}\{T\}(f)(-j,i) \\
&= \{N\}(f)(-j-1,i) \\
&= f(i,j+1) \\
&= \{T;\ 0,\ -1\}\ (f)(i,j) \\
&= \{D\}(f)(-j-1,\ -i) = f(i,\ j+1)
\end{aligned}
$$

q. Offset +. Once again consider the same image f, and let $k = 2$, Then

$$f + 2 = E\{2,\ K(f)\} <+> f$$

Because of the similarity to scalar multiplication, the details shall not be given.

r. Minkowski Addition ⊕. To demonstrate the representation of the grey-scale Minkowski addition, let

$$f = \begin{vmatrix} 4 & 1 \\ 5 & 0 \end{vmatrix}_{0,1}$$

and

$$g = \begin{vmatrix} * & 1 \\ * & 2 \end{vmatrix}_{0,1}$$

Then, since $K(g) = \{(1,\ 0),\ (1,\ 1)\}$, according to the representation,

$$f \oplus g = (\{T;\ 1,\ 0\}(f) + 2)\ (v)\ (\{T;\ 1,\ 1\}(f) + 1)$$

$$= \begin{vmatrix} * & 6 & 3 \\ * & 7 & 2 \end{vmatrix}_{0,1} (v) \begin{vmatrix} * & 5 & 2 \\ * & 6 & 1 \\ * & * & * \end{vmatrix}_{0,2}$$

$$= \begin{vmatrix} * & 5 & 2 \\ * & 6 & 3 \\ * & 7 & 2 \end{vmatrix}_{0,2}$$

s. Minkowski Subtraction $\ominus$. Using the same images as in the preceding illustration, we simply interchange <∧> for <v> to obtain

$$f \ominus g = \begin{vmatrix} * & 6 & 3 \\ * & 7 & 2 \end{vmatrix}_{0,1} \langle\wedge\rangle \begin{vmatrix} * & 5 & 2 \\ * & 6 & 1 \\ * & * & * \end{vmatrix}_{0,2}$$

$$= \begin{vmatrix} 6 & 1 \end{vmatrix}_{0,1}$$

t. Dot Product $\cdot$. Let

$$f = \begin{vmatrix} 2 & 3 & 4 \end{vmatrix}_{0,0}$$

and

$$g = \begin{vmatrix} 1 & -3 & 2 \end{vmatrix}_{0,0}$$

According to the basis representation,

$$f \cdot g = \{T;\ 0,\ 0\}(f \langle x\rangle g) \langle +\rangle \{T;\ -1,\ 0\}(f \langle x\rangle g)$$

$$\langle +\rangle \{T;\ -2,\ 0\}(f \langle x\rangle g)$$

$$= \begin{vmatrix} 2 & -9 & 8 \end{vmatrix}_{0,0} \langle +\rangle \begin{vmatrix} 2 & -9 & 8 \end{vmatrix}_{-1,0}$$

$$\langle +\rangle \begin{vmatrix} 2 & -9 & 8 \end{vmatrix}_{-2,0}$$

$$= \begin{vmatrix} 1 \end{vmatrix}_{0,0}$$

Since (0, 0) is the only pixel in the intersection of the three domains. Under the identification I1, $(\ 1\ )_{0,0}$ is identified with the real number 1, which, of course, is the dot product of f with g.

Proof of term representation. Since each translation places f(i,j)g(i,j) at the origin,

$$(f \cdot g)(0,0) = {}_{(i,j)}\ \Sigma\ {}_{K(f)} f(i,j)g(i,j)$$

Thus, under the identification I1, the representation is accurate if $(f \cdot g)(p,q)$ is undefined for all $(p,q) \neq (0,0)$. But $f \cdot g$ is defined at the point (p,q) if and only if $(p,q)_{(i,j)\ \varepsilon\ K(f)}\ K(f) - (i,j)$
But the latter intersection is the erosion of K(f) by K(∞) and, since K(f) is finite, this erosion consists only of the origin.

u. Correlation ●. To demonstrate the correlation representation employ the images f and g used above (s.). Noting that

$$K\ (g) = \begin{vmatrix} 0 & 0 & 0 \end{vmatrix}_{0,0}$$

we have

$$f \bullet g = 1\{T;\ -1,\ -1\}(f)\ (+)\ 2\{T;\ -1,\ 0\}(f)$$

$$= \begin{vmatrix} 4 & 1 \\ 5 & 0 \end{vmatrix}_{-1,0} \quad (+) \quad \begin{vmatrix} 8 & 2 \\ 10 & 0 \end{vmatrix}_{-1,-1}$$

$$= \begin{vmatrix} 8 & 2 \\ 14 & 1 \\ 5 & 0 \end{vmatrix}_{-1,-1}$$

Proof of term representation h denotes the proposed basis representation, then h is undefined at (i,j) if and only if $f \bullet g$ is undefined there. Specifically, h is undefined at (i,j) if and only if, for any $(r,s) \in K(g)$, $(i,j) \notin K(\{T;-r,-s\}(f))$.

This in turn is equivalent to

$$K(g) \cap K(\{T;\ -i,\ -j\}(f))$$

which is itself equivalent to

$$K(\{T;\ i,\ j\}(g)) \cap K(f)$$

On the other hand, $f \bullet g$ is undefined at (i,j) if and only if $(r-i,s-j) \notin K(g)$ for all (r,s) in K(f), but this is clearly equivalent to the above intersection being null. The proof is completed by showing that whenever the correlation is defined, $f \bullet g = h$. Changing variables in the definition of $f \bullet g$ yields

$$
\begin{aligned}
(f \bullet g)(i,j) &= \sum_{\substack{(m,n)\ \in\ K(g) \\ (m+i,n+j)\ \in\ K(f)}} f(m+i,\ n+j)g(m,n) \\
&= \sum_{\substack{(m,n)\ \in\ K(g) \\ (m+i,n+j)\ \in\ K(f)}} (\{T;\ -m,\ -n\}(f))(i,j)g(m,n) \\
&= h(i,j)
\end{aligned}
$$

v. Moving-Average Filter. Let f and g be defined as

$$
g = \begin{vmatrix} 2 & 4 \\ 1 & * \end{vmatrix}_{0,1} \qquad f = \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2}
$$

Then, according to the conventions regarding the basis representation,

$$F(f,g) = 2\{T;\ 0,\ -1\}(f)\langle+\rangle 1\{T;\ 0,\ 0\}(f)\langle+\rangle 4\{T;\ -1,\ -1\}(f)$$

$$= \begin{vmatrix} 4 & 6 & 2 \\ 0 & -2 & 4 \\ 2 & 8 & * \end{vmatrix}_{0,1} \langle+\rangle \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2}$$

$$\langle+\rangle \begin{vmatrix} 8 & 12 & 4 \\ 0 & -4 & 8 \\ 4 & 16 & * \end{vmatrix}_{-1,1}$$

$$= \begin{vmatrix} 16 & 9 \\ -3 & 10 \end{vmatrix}_{0,1}$$

w. Image Vector Norm. The supremum norm representation is demonstrated by using f as above and

$$h = \begin{vmatrix} 8 & -3 & 0 \\ 4 & 7 & 1 \\ 6 & -1 & * \end{vmatrix}_{0,1}$$

and the image vector (f, h). According to this representation,

$$\|(f,\ h)\|_{\infty} = |\ f\ |\ \langle v\rangle\ |\ h\ |$$

$$= \begin{vmatrix} 2 & 3 & 1 \\ 0 & -1 & 2 \\ 1 & 4 & * \end{vmatrix}_{0,2} \langle v\rangle \begin{vmatrix} 8 & 3 & 0 \\ 4 & 7 & 1 \\ 6 & 1 & * \end{vmatrix}_{0,2}$$

$$= \begin{vmatrix} 8 & 3 & 1 \\ 4 & 7 & 2 \\ 6 & 4 & * \end{vmatrix}_{0,2}$$

x. Gradient-type Edge Detection. The basis representation is demonstrated using Prewitt edge detector masks defined by g and h as shown

$$g = \begin{vmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{vmatrix}_{0,0}$$

and

$$h = \begin{vmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{vmatrix}_{0,0}$$

f represents the observed image upon which the mask will be applied.

$$f = \begin{vmatrix} 0 & 1 & 0 & 6 & 5 & 6 \\ 0 & 1 & 0 & 5 & 6 & 6 \\ 1 & 0 & 0 & 6 & 6 & 6 \\ 2 & 0 & 1 & 5 & 6 & 6 \end{vmatrix}_{0,3}$$

Moreover let t (threshold) = 9. Then

$$F(f,g) = \begin{vmatrix} * & * & * & * & * & * \\ * & -1 & 15 & 17 & 1 & * \\ * & -2 & 15 & 17 & 2 & * \\ * & * & * & * & * & * \end{vmatrix}_{0,3}$$

$$F(f,\ h) = \begin{vmatrix} * & * & * & * & * & * \\ * & 0 & 1 & -1 & 0 & * \\ * & -2 & 0 & -1 & 0 & * \\ * & * & * & * & * & * \end{vmatrix}_{0,3}$$

$$||(F(f,g),\ F(f,H))||_{\infty} = \begin{vmatrix} 1 & 15 & 17 & 1 \\ 2 & 15 & 17 & 2 \end{vmatrix}_{1,2}$$

Thresholding with t = 9 yields

$$E(f,g,h\ 9) = \begin{vmatrix} 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{vmatrix}_{1,2}$$

This completes the step-by-step trace of the basis representation illustration.

SECTION IV

UNIVERSAL IMAGE ALGEBRA STRUCTURE

1. INDUCEMENT METHODOLOGY

As discussed earlier,the inducement methodology is quite general; indeed, we saw how a simple group operation + in the range space leads at once to an induced operation $\langle + \rangle$. In addition the existence of any translation group in the domain leads to a domain induced operation on images. In the current section the methodology will be formalized into a general algebraic structure, to be termed the universal image algebra. Moreover it will be shown that the structure applies ipso facto to other image models besides the digital image algebra.

2. TYPES OF STRUCTURES

Consider a transformation group S and a lattice commutative ring I with identity, where I possesses a partial multiplicative inverse operation. The set of images is then given by

$$X = \bigcup_{A \subset S} I^A$$

the collection of a functions with domains in S and values in

I. The elements of I will be called grey values and the elements of S will be called pixels. S and I will be called the domain set and range set, respectively.

a. Examples of possible domain sets are

(1) the xy plane RxR

(2) the usual integer lattice ZxZ

(3) the hexagonal (or some other tiled) lattice

(4) the collection of pixel squares in RxR (which lead to the sampled data images)

(5) the orientation histogram on a Gaussian sphere

(6) R^n, $n > 1$ where $R^n = RxRxRxR...R$

(7) Z^n, $n > 1$

b. Examples of grey values (the elements constituting the range set I) are

(1) real numbers (as in the digital image algebra)

(2) complex numbers

(3) Galois numbers (numbers from a finite field)

(4) vectors whose components consist of any of the above types

(5) possibility-valued distributions

3. GENERAL STRUCTURES

In order to analyze the general structure, it is useful to characterize the types of fundamental operators involved. In the digital image algebra, there are 7 basic operators, 3 of which are domain induced and 4 of which are range induced. There are also 3 very basic "structural" operators, which, because of the identifications of real numbers with images defined only at the origin and of subsets of ZxZ with zero-valued images, can be represented as macro-operators.

These structural operators are the domain extractor K, the parameter extractor G, and the existential operator E. Whereas the first and last of these have been expressed in the macro-operator section, the parameter extractor has yet to be defined. For any image f, G(f) gives the grey value at the G origin. Rigorously, if $f(0, 0) = a$, then under the identification I1,

$$G(f) = (\ a\ )_{0,0} = a_0$$

If f is not defined at the origin, G(f) is still defined: in such a case it is the null image. A basis representation for G is given by $G(f) = f <+> 0_0$.

At this point, we shall present the fundamental domain induced, range induced and structural operators for a number of image-algebraic structures. As a result of different algebraic structures in the domain and range sets, it might very well be that some change in the basis elements might occur; however, as will be seen in subsequent paragraphs such changes, if they exist at all, are straightforward and not extensive.

The first case we consider is the extension of the seal field of grey values into the complex field. In this case, we assume that the grey values are complex numbers and hence

$$X = \bigcup_{A \subset Z \times Z} C^A$$

The extension occurs naturally in image processing due to the use of Fourier transforms (among other complex techniques). Since C is a field (like R), we should expect that the structure of X is similar in the complex case to the usual digital algebra setting. Grey values are of the form a + ib and images are of the form f = u + iv, where u and v are the real and imaginary parts of the image, respectively. For instance, if

$$f = \left| \begin{matrix} 3 + i5 & 4 - i \\ 2 & 5 + i3 \end{matrix} \right|_{-1,-1}$$

then

$$\text{Real } f = \left| \begin{matrix} 3 & 4 \\ 2 & 5 \end{matrix} \right|_{-1,-1} = \text{Re } f$$

and

$$\text{Imaginary } f = \left| \begin{matrix} 5 & -1 \\ 2 & 3 \end{matrix} \right|_{-1,-1} = \text{Im } f$$

The structural mappings must reflect the complex range. Thus,

$E\ (a + ib, A) = (a + ib)_A$

$G(f) = u(0, 0) + iv(0, 0)$, if f is defined at the origin

Under the identification K(f) is the domain of A.

Insofar as the range induced mappings are concerned, if we assume that f and g are elements of C^A and C^B, respectively, then f <+> g, f <x> g, and f </> g are defined in the same manner as in the real case. However, there is difficulty with f (v) g because the complex field is not ordered. To arrive at a maximum operation, the maximum v is induced on the real and imaginary parts separately. To wit,

$$(f\langle v\rangle g)(x,y)=\{u_f(x,y) \vee u_g(x,y)\} + i\{v_f(x,y) \vee v_g(x,y)\}$$

for (x, y) in the intersection of the domains of f and g, where we have let u_f and u_g denote the real parts of f and g, respectively. The extended union is formed as usual by letting f (v) g equal f on K(f) - K(g) and g on K(g) - K(f).

Since the complex conversion occurs only in the range, the domain induced mappings {T}, {N}, and {D} are unaffected.

In continuous image processing, the domain set is RxR. Thus, the collection of images is

$$X = \bigcup_{A \subset R\times R} R^A$$

The definition of both the structural and range induced mappings are the same as in the digital image algebra setting; however, because of the ability to continuously translate and rotate, the domain inducement is altered, even though in the case of {T}, the defining relation appears the same. In any event, the domain induced mappings are given by

$$\{T;\ s,\ t\}(f)(x,y) = f(x - s,\ y - t)$$

$$\{N;\ f,\theta\}(f)(x,\ y) = f(x \cos\theta - y \sin\theta,\ x \sin\theta + y \cos\theta)$$

$$\{D;\ f,\theta\}(f)(x,y) = f(-x\ \cos\theta + y\ \sin\theta,\ x\ \sin\theta + y\ \cos\theta)$$

As noted earlier, in various settings the basis may have to be altered. In the digital grid, we only required the basic 90 deg operator {N} to generate the group of rotations

$$\{(N),\ (N)^2,\ (N)^3,\ (N)^4\}$$

There does not exist any generator in the continuous case. As a consequence, the basis must contain {N; ., .} and {D; ., .}, both of which are functions of two variables, an image and a real number (radian). The finite group theoretic elegence and simplicity are lost due to the nature of the model. Should, however, we consider a finite subgroup of images resulting from the actions of {N; .,θ} or {D; .,θ}, the resulting group is isomorphic to either the cyclic group $C^n\ (R^2)$ or the dihedral group $D^{2n}\ (R^2)$, respectively, n = 1, 2,.... In the terms of inducement $C^n(R^2)$ is induced by application of the matrix

$$\begin{vmatrix} \cos \dfrac{2\pi}{n} & -\sin \dfrac{2\pi}{n} \\ \sin \dfrac{2\pi}{n} & \cos \dfrac{2\pi}{n} \end{vmatrix}$$

(relative to the usual basis). $D^{2n}(R^2)$ occurs by application of the preceding matrix in conjunction with

$$\begin{vmatrix} 1 & 0 \\ 0 & -1 \end{vmatrix}$$

(relative to the usual basis).

If we consider image processing using sampled data systems, then

$$X = \bigcup_{A \subset \overline{Z \times Z}} R^{A}$$

, where $\overline{ZxZ}$ denotes the collection of unit squares centered at the grid points, where each square contains its left and bottom boundary, but does not contain its top and right boundary. Algebraically, we treat sampled data images as if they were digital images. Ignoring topological considerations, such an approach is legitimate. Thus, the induced and structural mappings are the same as in the digital case. If we do not ignore topological considerations, then there does not exist any rotation or reflection type mappings other than the identity.

Next consider multi-spectral image processing, where the domain set is ZxZ and the range set is R, the set of Euclidean n-vectors. The usual vector addition is defined as

$$a + b = \begin{vmatrix} a_1 + b_1 \\ a_2 + b_2 \\ \vdots \\ a_n + b_n \end{vmatrix}$$

where

$$a = \begin{vmatrix} a_1 \\ a_2 \\ \cdot \\ \cdot \\ \cdot \\ a_n \end{vmatrix} \qquad b = \begin{vmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ b_n \end{vmatrix}$$

and multiplication in I is given by

$$ab = \begin{vmatrix} a_1 b_1 \\ a_2 b_2 \\ \cdot \\ \cdot \\ \cdot \\ a_n b_n \end{vmatrix}$$

A commutative ring with identity results

$$1 = \begin{vmatrix} 1 \\ 1 \\ \cdot \\ \cdot \\ \cdot \\ 1 \end{vmatrix}$$

Moreover the structure possesses zero divisors. Indeed, for $n = 2$,

$$\begin{vmatrix} 0 & 1 \\ 0 & 0 \end{vmatrix} \cdot \begin{vmatrix} 1 & 0 \\ 0 & 0 \end{vmatrix} = \begin{vmatrix} 0 \\ 0 \end{vmatrix} = 0$$

the zero element in I. To obtain a lattice, we use the operations v (max) and $\wedge$ (min) defined by

$$a \vee b = \begin{vmatrix} a_1 \vee b_2 \\ \vdots \\ a_n \vee b_n \end{vmatrix}$$

and

$$a \wedge b = \begin{vmatrix} a_1 \wedge b_1 \\ \vdots \\ a_n \wedge b_n \end{vmatrix}$$

respectively. A partial multiplicative inverse operation is defined by

$$a \div b = \begin{vmatrix} a_1 + b_1 \\ \vdots \\ a_n + b_n \end{vmatrix}$$

providing b_i is not 0, for $i = 1, 2, \ldots, n$. Given the above algebraic description of I, the multispectral image algebra can be described by

$$X = \bigcup_{A \subset Z \times Z} (R^n)^A$$

The structural mappings are given by

$$E\left\{ \begin{vmatrix} t_1 \\ t_2 \\ \cdot \\ \cdot \\ \cdot \\ t_n \end{vmatrix}, A \right\} = \begin{vmatrix} t_{1_A} \\ t_{2_A} \\ \cdot \\ \cdot \\ \cdot \\ t_{n_A} \end{vmatrix}$$

$$G(f) = \begin{vmatrix} f_1(0, 0) \\ f_2(0, 0) \\ \cdot \\ \cdot \\ \cdot \\ f_n(0, 0) \end{vmatrix}_{0,0}$$

where we assume that (0, 0) is in the domain of f. Lastly, K(f) gives the domain of f in ZxZ. Of course, under the identification I's K(f) is actually an image defined on A whose values are all given by the zero vector, the additive identity in I.

Range inducement is analogous to the digital image case except that n-vectors that constitute the range set are utilized. For instance,

$$(f \langle + \rangle g)\ (i,j) = (\ f_k\ (i,j) + g_k(i,\ j)\ )$$

where the notation indicates that the vectors are added componentwise. Since the domain is ZxZ, domain inducement proceeds as in the digital image algebra.

Returning to the general case cited at the outset of the section, certain algebraic properties are seen to hold. For the sake of completeness, before proceeding to the next theorem, we state the definition of a module. V is a module over K if

M1) For all u and v in V, u + v is in V and V is an Abelian group

M2) For all v in V and k in K, kv lies in V and

a) $(k + 1)v = kv + 1v$

b) $k(u + v) = ku + kv$

c) $(k1)v = k(1v)$

d) $1v = v$

Theorem. If the operations <+> and <x> in the universal image algebra are restricted to images in $V = I^A$, A fixed in S, then the resulting structure is a module over I.

Proof. We need to verify the properties constituting the definition of a module.

M1) By construction, <+> is closed. Moreover, (V, <+>) is an

Abelian group by inducement.

M2) For any t in I and f in V, tf lies in V since

$$tf = E\{t, A\} \langle x\rangle f$$

a) $(s+t)f = (s + t)_A \langle x\rangle f = (s_A \langle +\rangle t_A) \langle x\rangle f$

$= (s_A \langle x\rangle f) \langle +\rangle (t_A \langle x\rangle f) = sf + tf$

where the distributivity of $\langle x\rangle$ over $\langle +\rangle$ is justified by inducement.

b) $t(f \langle +\rangle g) = t_A \langle x\rangle (f \langle +\rangle g) = (t_A \langle x\rangle f) \langle +\rangle$

$(t_A \langle x\rangle g) = tf \langle +\rangle tg$

c) $(ts)f = (ts)_A \langle x\rangle f = (t_A \langle x\rangle s_A) \langle x\rangle f$

$= t_A \langle x\rangle (s_A \langle x\rangle f) = t(s_A \langle x\rangle f) = t(sf)$

where the inducement justifies the use of the associative law with respect to $\langle x\rangle$.

d) $1f = 1_A \langle x\rangle f = f$

To state the next theorem, we require the definition of an associative algebra. V is an associative algebra over the field K if

A1) V is a vector space over K.

A2) For all u, v in V, uv lies in V and

a) $(u + v)w = uw + vw$

b) $w(u + v) = wu + wv$

c) $t(uv) = (ku)v = u(kv)$, for any t in K

d) $u(vw) = (uv)w$

If (d) does not hold, then V is called a nonassociative algebra.

Theorem. If the operations <+> and <x> in the universal image algebra are restricted to images in $V = I^A$, for A a subset of S, and for I a field, then the resulting structure is an associative algebra over I.

Proof. The theorem follows at once by range inducement. For instance, note that For A2 (c),

$$t(f \langle x\rangle g) = t_A \langle x\rangle (f \langle x\rangle g) = (t_A \langle x\rangle f) \langle x\rangle g$$
$$= (f \langle x\rangle t_A) \langle x\rangle g = f \langle x\rangle (t_A \langle x\rangle g)$$
$$= f \langle x\rangle (tg)$$

As an example of a situation where the previous theorem does not apply, consider multi-spectral imaging. If X is fixed to a specific set of pixels, the restriction is a module over the grey value set R^n; however, it is not a vector space and is therefore not a nonassociative (or associative) algebra.

In the primary digital image algebra, the domain set is ZxZ. In order to process images defined on spatial, or higher dimension points, Z^n must be considered. The space of images is then

$$X = \bigcup_{A \subset Z^n} R^A$$

where R is the real field. In this setting, each grey value is of the form $f(x_1, x_2, \ldots, x_n)$. For such images the existential operator takes the form

$$E(t, A) = t_A,$$

where, for any point $(x_1, x_2, \ldots, x_n)$ in A,

$$t_A(x_1, x_2, \ldots, x_n) = t.$$

The parameter extractor is defined by

$$G(f) = f(0, 0, \ldots, 0)$$

and the domain extractor is the usual $K(f) = 0_A$, where A is the domain of f.

Since the range space is R, the range inducement follows the customary pattern. For instance,

$$(f <+> g)(x_1, x_2, \ldots, x_n)$$
$$= f(x_1, x_2, \ldots, x_n) + g(x_1, x_2, \ldots, x_n)$$

For n > 2, the domain inducement is much more complicated than in the case n = 2. Essentially, the problem lies with the transformation-group structure of Z^n. For instance, in the case n = 3, images are of the form f(x, y, z) and we have the following domain induced mappings holds true

$$\{T;\ i, j, k\}(f)(x, y, z) = f(x - i, y - j, z - k)$$

$$\{N_1\}(f)(x, y, z) = f(x, z, -y)$$

$$\{N_2\}(f)(x, y, z) = f(y, z, x)$$

$$\{D_1\}(f)(x, y, z) = f(-x, y, z)$$

$$\{D_2\}(f)(x, y, z) = f(x, -y, z)$$

$$\{D_3\}(f)(x, y, z) = f(x, y, -z)$$

For multi-spectral images whose grey values are m-vectors in R^m and whose pixels are elements of Z_n, the previous domain space analysis can be applied to the original multi-spectral case in which images were defined on points of ZxZ. Since the conjunction of the two cases is straightforward, the grey value is of the form

$$\begin{vmatrix} f_1(x_1, x_2, \ldots, x_n) \\ f_2(x_1, x_2, \ldots, x_n) \\ \vdots \\ f_m(x_1, x_2, \ldots, x_n) \end{vmatrix}$$

where $(x_1, x_2, \ldots, x_n)$ is an element of Z^n.

Other settings are possible; however they are similar to those already introduced and will be given herein. As mentioned at the outset of the section, the domain set S is a transformation group and the range set I is a lattice commutative ring with identity that possesses a partial multiplicative inverse operation.

SECTION V

ARTIFICIAL INTELLIGENCE FOR IMAGE ALGEBRA

1. THE ARCHITECTURAL TOPOLOGY OF AUTOMATED IMAGE ALGEBRA ALGORITHM DEVELOPMENT

The integration of artificial intelligence and expert system technologies, with the image processing algorithms development has been ad hoc and at this time is not a well defined process. There appears to be no clear direction or approach currently being pursued to either analyze image processing algorithms for their constitutent components and embedded expertise or to elicit from image processing experts, their strategies for effective image processing (pattern recognition, classification, detection, etc.). Further, even if the component pieces characterizing expert image processing could be formalized and represented, their interaction and reintegration into decisive new algorithms in at least a semi-automated way requires further research and development.

The thrust of this section is to clearly identify the necessary architectural topology of an automated image processing algorithm development process based upon the image algebra. Architectural topology refers not only to the basic building blocks of the process and their interconnections but to the degree of closeness and connectivity of the technologies underlying the automated algorithm development process. With such a patterned architectural approach, the reasoning strategy underlying automated image processing algorithm development will become evident. Feed back and fine tuning during the algorithm development process will then be seen to be the sufficient condition for the final design of decisive solutions to the automated target detection, classification, recognition, and decision problem of algorithm development. The final demonstration of the ability of an automatically generated algorithm will, in general, require running it against classified data. Various other aspects in developing the total capability of the algorithm generator will of necessity be of a classified nature; for example, the interview of certain image processing experts.

The starting point for the elaboration of the underlying technologies involved is the newly emerging field of knowledge engineering. The perspective of this section is that knowledge engineering is distinct from artificial intelligence in that it is a branch of systems engineering which utilizes aspects of computer science in the overall process of engineering intelligent systems. As such, knowledge engineering utilizes expert systems technology in addition to bringing other scientific disciplines to bear on the problem, namely, cognitive and systems science and the scientific and engineering disciplines underlying pattern recognition, classification, detection, and other aspects of image processing.

Knowledge engineering of the image processing algorithm development process must consider such problematic issues as multilevel expert systems, knowledge and metaknowledge elicitation from experts, knowledge engineering of coherent collections of methodological tools as they appear in the literature, and the implementation of expert knowledge in specialized hardware and software architectures. With respect to image processing specifically and integrated signal processing in general, the role of neural net model based computing architectures must be considered (parallel array memories). Additionally, recently explored rigorous unified mathematical structures for both domain independent expert based systems and image processing algebras and algorithms must be integrated in a clear way by the knowledge engineering process.

Fundamental results developed by authors and co-researchers in the area of multilevel expert systems and knowledge engineering (References 4 through 15) have led to new paradigms for interfacing expert systems with relatively unsophisticated users. These results have also pointed the way to potential hardware synthesis of expert systems. Research into both cognitive and system modelling have led to the discovery of primitives which are the basis for formalization of knowledge engineering elicitation techniques. These primitives are perceived to be fundamental in the architectural framework for semi-automated knowledge elicitation modules to be used in developing knowledge bases for multilevel expert systems involved in automated image processing algorithm development. The discovery of these principles has significantly increased the possibility for the actuation of a high level research and development facility for the automated design of target detection/classification algorithms and autonomous target recognition systems. Of paramount importance is the development of the capability to elicit the strategy from expert algorithm designers and/or algorithms that experts have designed. Further research and development is needed on these techniques which are capable of both eliciting and reproducing the expert's strategy.

The remainder of the section concentrates on two aspects of designing a facility for automated image processing algorithm development. The first is knowledge engineering as developed by the authors and co-researchers. The second is the architectural topology and research and development philosophy of a facility designed to bring the full potential of the emerging field of knowledge engineering and allied technologies to bear on the problem of increasing the performance and impact of image processing based weapon systems and the effectiveness of associated personnel. It is essential that both aspects of the problem be taken into consideration and integrated. In general, state of the art, sophisticated systems have a human being in the loop.

2. KNOWLEDGE ENGINEERING FOUNDATIONS FOR IMAGE PROCESSING ALGORITHM DEVELOPMENT

Any system is a product of its time. System builders necessarily operate against a background of competing ideas and controversies and must also confront the limitations of their resources. Summarizing experiments and formulating a simplified theory necessarily involves stepping outside of this rich historical process and committing oneself to a vantage point with a high possibility of success.

This report links the results of research investigations into the morphology and operational methodology of knowledge engineering to image processing specifically, and more generally to other technologies. Basically, an inductive modelling approach to establishing a foundation for knowledge engineering has been pursued which results in the recognition of three distinct areas of expertise as being fundamental for the scientific development of knowledge engineering. These areas are knowledge elicitation, knowledge formalization, and knowledge representation.

Each area has a distinct collection of fundamental knowledge, methodologies, organizing principles and presuppositions associated with it. It is the union of these areas individually in addition to transformational procedures from one area to the other which appears to constitute the field of knowledge engineering. This defining perspective of knowledge engineering will be shown to be both pragmatic and an appropriate consequence of a natural growing convergence between psychology, systems theory, cybernetics and computer science. It is essential that the distinction between knowledge formalization and knowledge representation be made. Knowledge formalization has to do with the conceptual organization and structuring of knowledge whereas knowledge representation deals with the form in which the computer implementation of the knowledge takes place.

The growing convergence between psychology, systems theory and computer science disciplines have interacted in such a way as to become major mutual influences on one another's developments. The cognitive psychology paradigm is based on information processing concepts applied to human behavior (References 16 and 17). General systems theory (Reference 18) and cybernetics (Reference 19) are based on modeling the organism/person as an information processing entity. Fifth generation computers (Reference 20) offer natural human-computer interaction using techniques of knowledge processing derived from artificial intelligence studies in computer science. (Reference 21)

There are compelling reasons for the decoupling of knowledge elicitation, formalization, and representation which will be given shortly. In a recent paper delimiting expert systems (Reference 22), Gregory points out that there is a fundamental difference between an expert system designed for diagnosis versus one designed for operational planning. In the latter case the possibility of the environment acting as a conscious agent in competition with the expert system must be considered. This requires that the capability to create a template, a morphological grid, which can be imposed on a problem for successful solution with the resources available to the knowledge based system be developed. The type of information needed to solve this kind of problem is essentially system theoretic and cybernetic in nature.

The use of different perspectives of the domain must be incorporated in the knowledge base. A cybernetic representational system must be able to compute similarity, distinction, and conflict and to facilitate the resolution of conflict and ambiguity when it arises (Reference 22). It is precisely these types of computational measures which are currently being researched and developed in system science, mathematics, engineering and philosophy (References 4, 23, 24, 25, 26, 27, 28 and 29). With the potential for formalizing the meta-thinking of an expert (i.e. higher order domain independent reasoning but not necessarily category independent) present in the current formulations of the system theorist Klir (Reference 4), the decoupling of knowledge formalization from knowledge representation becomes even more desirable. The role of morphology (Reference 30) as a knowledge formalization tool is another area which is applicable to the foundational frame proposed (Reference 8).

However, image processing specific methodology must still be elaborated. These methodologies must give the knowledge engineer a particular context within which an expert's statement is to be analyzed. A knowledge engineer must be trained to respond to both the form and content of an expert's statement.

The derived and developed definitions of form and content are the definitions of deletions, lack of referential indices, unspecified verbs, nominalizations, universal quantifiers, modal operators, cause and effect, etc.

In this section the specific procedures involved in the elicitation of image expert knowledge are outlined in principle. The methodology of fine tuning, and further elaboration on research advances in the application of the knowledge elicitation phase of knowledge engineering is beyond the scope of this work.

The following commentary is given in brief form to convey the essential elements of the communication skills of a knowledge engineer. The full requisite variety of the communication in the sense of Ashby (Reference 31), Bandler (Reference 32), Bandler and Grinder (Reference 33 and 34), Klir (Reference 4), and Reese (Reference 10 and 11) is almost impossible to communicate by the written word unless one gives special attention to the nuances of wording and non-verbal communication.

ELICITING INFORMATION

Expert	Knowledge Engineer	Content
*Deletion		
A statement with excluded or missing referent.		
"I erode."	What is it that you erode?	deleted information
"I'm not quite sure."	About what?	deleted information

Both of these statements taken independently are examples of deletion. If however, they were taken as sequential dialog, experienced knowledge engineers trained in the techniques of Neutral Linguistic Programming (NLP) would alter their elicitation strategy when a question directed towards reconnecting the expert with their underlying reality experience failed and elicited a statement of the same form (deletion). Understanding that some of the presuppositions of a field may also be operational organizing principles in that they direct the methodology within context, the particular sensory channel in which the expert is experiencing difficulty in accessing information must be determined. Consider the following interchange.

"I erode."	What is it that you erode?	deleted information

"I'm not quite sure."

Obviously, the expert is having difficulty accessing the desired information. It is assumed of course that the knowledge engineer is pacing the expert and has not broken rapport.

"I'm not quite sure."	How, specifically, do you know you are not sure?	sensory channel of difficulty
"It's not clear to me."	Can you show me an example of what you you erode?	deleted information within sensory channel

This last example shows that a how question directed at the appropriate time in attempting to restore a deletion can yield the sensory channel in which the access difficulty is being manifested by the expert. Once determined, the knowledge engineer can return to the job at hand. Without further comment at this time, the sensory based information channels used by the expert can also be determined by knowledge elicitation techniques which are designed to yield information from the non-verbal communication (behavior) of the expert.

Other examples of deletion which are encountered with image processing experts and are of a slightly different nature are those which involve a deletion of either a principle of evaluation or the elements from which the principle is derived. This is a principle of basic set theory where a class is defined by either a principle or enumeration of the class.

"This is a better approach."	Better than what other approaches? or By what criteria?	defining principle used
"Convolving images is best in this case."	By what criteria? Compared to what?	defining principle used

*Lack of Referential Index

A statement with an unspecific noun or pronoun.

"It is the most important part of the image which I look at first."	Specifically, which part is the most important?	specific reference
"They give me the specifications."	who exactly?	specific person(s)

*Unspecified Verbs

Verbs not specific with regard to the underlying process as to how, when, where.

"Dr. Matheron taught me how to do this."	How?, When?, Where?	specific reference to underlying process
"I intuit the right move when I design the algorithm."	How, specifically, does your intuition work in this case?	specific reference to underlying process

In response to questions of how?, the expert may respond with a complex equivalent of the unspecified verb which requires further knowledge engineering analysis.

Nominalization

An active ongoing process is made static by conversion of a verb into a noun.

"First, I get foreground backing."	How could the foreground back you?	reestablish dynamics of action
"The critical issue is pixel traversal in the image."	How do you traverse the pixels?	reestablish dynamics of verb action
"We use operational imaging."	How do you operationally image?	reestablish dynamics of verb action

Limits of the Expert's Model of Knowledge

Universal Quantifiers

Statements that preclude exceptions or alternatives by using generalizations of an all or nothing nature. The strategy is to elicit an existential quantifier.

"We never use heuristics."	Never? Has there ever been a time or place when ...?	exception or counter-example
"We have a rule that all subimages of the image are checked."	All subimages? Is there ever a situation when ..?	exception to rule special case rule

Modal Operators

A statement which involves modal operators of necessity, sufficiency, possibility, etc. These sentences have the potential for rich logical structure and within the context of knowledge engineering require the introduction of questioning techniques that elicit measures of uncertainty, conflict, chaos, etc. In this report only the modal operator of necessity is treated from an NLP perspective. Basically, the response of the knowledge engineer will be of two types. ' What stops you? ' is intended to elicit past experience that supports the modal operator and ' What would happen if...? ' is intended to place the expert in the future to examine possible consequences of replacing the modal operator.

"I can't use that feature in this image."	What stops you?	past experience
	What would happen if you did?	possible outcomes, consequences
"I have to use a smaller kernel for prediction."	What will happen if you don't?	possible outcomes, consequences
	What stops you from using a larger kernel	past experience

Semantic Ill-formedness

Statements in this class are essentially statements of belief for which the underlying experiential data supporting the belief is missing. When questioned, these beliefs either have no basis, have a basis which the knowledge engineer then knows, or are of the form of judgments unique to the expert's experience but not necessarily supported by other expert's experience.

Mind Reading

An assumption on the part of either the expert or the knowledge engineer that the other individual "knows" what is being thought or felt without direct communication.

"I'm sure you can see how I analyze the image."	How, specifically, can you be sure I see how you analyze?	complex equivalent of source of information

Cause and Effect

Statements which contain a belief that a specific action on the part of one person causes a specific response in another person. The knowledge engineer questions how does X cause Y. Note that cause and effect beliefs for all other cases is preferably handled under the category of eliciting information, specifically through the use of an unspecified verb.

"You are frustrating me during this elicitation process."	How do I frustrate you?	complex equivalent of action
"His elicitation process threatens me."	How dces his elicitation process make you feel threatened?	imagined action

Lost Performative

Statements from the expert's model of the world which are projected as actual facts about the world. These may be statements which are not agreed to by all experts in the same field as being an actual fact about the world.

"Convolution is the only good way to find features.	How do you know? According to whom? For whom?	source of opinion or strategy
"This is the only way an expert in imaging works."	How do you know? According to whom?	source of opinion or strategy

The knowledge engineer monitors the expert's reasoning strategy by keeping track of the current type of knowledge structure the expert is accessing (source, data, behavior, structure) and either anticipating or eliciting the next node in the strategy and the methodology used to get there. The knowledge engineer can anticipate the path to be followed based upon the type of problem the expert is trying to solve. At the highest level the authors identified several generic problem solving strategies typically handled by experts.

3. ARCHITECTURE RESEARCH AND DEVELOPMENT PLAN FOR A FACILITY FOR AUTOMATED IMAGE PROCESSING ALGORITHM DEVELOPMENT

In the previous sections of the report, a foundation for the process of knowledge engineering was formulated. This is a prerequisite for a managed, scientific approach to automated algorithm development in any field. Current approaches are ad hoc, attempting to improve heuristics, gain insight, etc. with no structured approach. The consistent development of cost effective, intelligent weapon systems, whose intelligence performance exceeds performance requirements, requires knowledge engineering not shot in the dark "artistic" breakthroughs; although, they are appreciated when they occur.

A knowledge engineering approach is always looking towards the future for new developments in the underlying sciences on which knowledge engineering is built. For example, current approaches to knowledge representation include if-then rules, frames, semantic nets, finite state automatons, etc. However, in the not too distant future, knowledge will be represented in neural model networks. Even now, the first cousins of these neural net models are being patented: the PAM (parallel array memory) boards. A knowledge engineering approach to automated image processing algorithm development aggressively pursues the latest information in the cognitive, system, and computer sciences. An ad hoc approach takes a more experimental perspective.

The following is a design for an effective, efficient facility for the development of intelligent weapon systems based upon the premise of a knowledge engineering approach to automated image processing algorithm development. Note that in principle the design would encompass integrated signal processing (vision, sonar, kinesthetic sensor, etc.). In each of the groupings outlined, there is a current body of knowledge and a research and development program as well as a protyping component. The full discussion and elaboration on the complete facility is well outside the scope of this report.

4. FUNCTIONAL PRIMITIVES FOR AN INTELLIGENT SYSTEM CENTER

The following functional decomposition of an Intelligent System Center into two laboratories and their constituent working groups is based upon a analysis of what has to be done in order to produce intelligent weapon systems, within current resources, reasonable cost, and with a high probability of success.

INTELLIGENT SYSTEM CENTER

Knowledge Engineering Laboratory

Knowledge Elicitation Group

* Elicitation of Knowledge Source & Strategies

  **Human Experts
  **Literature
  **Other Life Forms

Knowledge Formalization Group

* Formalization of Knowledge Source & Strategies

  **Basic Knowledge Structures
  **MetaKnowledge Structures and the MetaHierarchy
  **Categories of Knowledge Structures
  **Strategies
  **MetaStrategies
  **Uncertainty Criteria for Decision
  (probability, possibility, theory of evidence, min/max entropy, etc.
  **Reconstructability
  **Morphology
  **Logic

Knowledge Representation Group

* Knowledge Representation in Varied Modalities

  **Rules, Semantic Nets, Frames
  **Inference Engines
  **ADA based GST Frames
  **MultiLevel Representations
  **Automata
  **Neural Networks

**Cellular Models
**Submodalities

Advanced Technology Transfer Group

* Identification and Transfer of Sources of Knowledge to the Knowledge Engineering Center

    **(interdisciplinary consulting group)

INTELLIGENT SYSTEM CENTER

Intelligent System Prototype Laboratory

Intelligent System Software Engineering Group

* Software Engineering of Intelligent Systems

  **ADA based Expert Systems
  **Experimental Expert Systems
  **On Board Expert Systems
  **Neural Net based Expert Systems

Integrated Signal Processing Group

* Signal Processing Algorithm Development & Performance Evaluation

  **Image Processing
  **Sonar Processing
  **Select Band Limited Processing

Advanced Physical Architecture Group

* Functionally Specific Hardware Architectures

  **Distributed RISC computers
  **Parallel Array Memory Boards & Neural Net Architectures
  **Optical Computing
  **Sensor Interfacing

Intelligent System Engineering Group

* System Engineering of Intelligent Prototypes from Knowledge Elicitation Stage through Synthesis of Intelligent Prototype

  **Workload, Capacity, Performance and Service Requirements
  **Workload Characterization

**Capacity Planning
**Feasibility of Functional Synthesis
**Benchmark and Test Evaluation
**Feedback and Fine Tuning

As indicated previously, the complete elaboration in detail of how these two centers would operate and interface within the Intelligent Systems Laboratory and the necessary resources to bring them to fruition are outside the scope of this report. However, in order to clarify to some degree the types of work carried out, the following sample of work items is listed. Obviously, the actual work content is far more extensive and detailed and is known in principle:

a) Work items:

(1) The transfer of multilevel expert system technology to the field of image processing

(2) The hardware design of an image processing computer based on image algebra concepts developed by the Singer Corporation.

(3) Investigation of extensions of image algebra concepts to form complete instruction sets for functionally specific image algebra computers.

(4) The design of a standard operating environment for a fully automated multilevel expert image processing algorithm development facility. This ADA based standard operating environment is based upon a distributed architecture.

(5) The back end level of the expert system consists of:

- o functionally specific image processing computers whose instructions sets are precisely the primitive operators of image algebras

- o a library of macro transforms and control structures to implement the image processing functions. These functions can later be implemented in hardware in the distributed architecture. As parallelism and functionality emerge from elicited strategies and from monitored usage of image processing programs (monitoring to be done in the front end), this will induce the design of the distributed architecture of the back end.

(6) The front end level of the expert system consists of an expert system incorporating problem solving expertise of human image processing experts and the communication mechanisms necessary to interface with the image algebra computers in the distributed architecture.

(7) Development of elicitation strategies and techniques to extract expert strategies from image processing programs and from the image processing experts themselves. Eventually research and development in this area should lead to an automatic knowledge elicitation expert system module incorporated into the image processing facility. This module will allow the expansion of the knowledge base by interviewing experts and analyzing samples of their algorithm designs.

(8) Elicited strategies will be incorporated into the front end of the image processing expert system using mathematically rigorous methods of metaknowledge representation. These methods are currently being researched by the authors and co- researchers.

(9) Analysis of current image processing algorithms and more complex image processing programs based on specific sets of image processing algorithms. These analyses are to be carried out in several modes including a complete translation to image algebra machine primitives. The latter will aid in further research into the design of control instructions for the image algebra machine.

(10) A thorough investigation of the use of measures of uncertainty in target detection/recognition algorithms. These measures are based upon entropy, probability, possibility, and the theory of evidence as well as fuzzy set theory. If measures of uncertainty are a part of an image processing expert's stategy, they will be elicited and incorporated into the front end of the multilevel expert system. If they are not, further research into new target detection/recognition algorithms utilizing measures of uncertainty should be initiated. Results from reconstructability analysis developed in general systems theory should be quite useful here.

(11) The transfer of technology from various disciplines to benefit automatic target recognition algorithm development. Typical areas to be transferred include stereology, syntactical recognition procedures, mathematical morphology, random geometry,

constructive function theory, and potential theory, among others.

(12) Deep research into totally innovative methods of formalizing, representing, and solving time critical target recognition problems. Innovation here means exploring abstract areas of mathematics generally unaccessible via traditional engineering approaches. Research directions in this area have been motivated by successful development of image algebras by researchers. Potentially, results from this research may lead to distinct machine architectures such as data-flow, systolic, and wave-front array processors being applied to difficult autonomous target recognition problems.

(13) Research and development of adaptive signal processing techniques in conjunction with measures based upon statistics and uncertainty principles.

As a specific example of the type of output of one of the groups in the knowledge engineering center, the knowledge elicitation group, the following preliminary list of rules was generated. These were based on a review of an introductory presentation on image processing algorithms given by Dr. Benjamin M. Dawson, Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology. It is important to realize that this presentation was intended as an introduction to image processing. Again, these rules represent the information given in the presentation, no more, no less.

b) Knowledge Based Automated Image Processing Algorithm Development Rules

(1) Classification of rules

Categories of image processing algorithms: point proccesses, area processes, geometric proccesses frame processes, image based versus symbolic methods, linear versus nonlinear, knowledge level used, knowledge about pixel structures.

(2) Goals of rules

Improve appearance, highlight information, measurement of image elements, classification or matching of image elements, recognition of items in the image.

Image measurement makes few assumptions about elements in image. Classification and recognition require successively more knowledge about what can appear in the image. Knowledge level used in an algorithm can range from simple assumptions about the physics of image formation to specific world knowledge about possible items in the scene. Note: Most image processing algorithms are compounds of other more primitive algorithms. Therefore, a knowledge of which algorithms to apply and in what order to apply them in order to reach a processing goal (classification by goal).

(3) Preliminary rules

(a) PERFORMANCE: If image transformation is a point process then use look up table for computational efficiency

(b) CONTRAST: If image is to be brightened (darkened) then add (subtract) to (from) pixel value.

(c) CONTRAST: If shading is to be corrected or pixel values to be smoothly changed in an area then estimate the shading functions and use point process with pixel location that computes the inverse of the shading function to eliminate (correct) the shading.

(d) CONTRAST: If contrast of areas is to be highlighted or adjusted then smoothly change the pixel values in the area.

(e) CONTRAST: If contrast at center of image is to be increased and edges to be faded then

```
pixel-out-value =
       pixel-in-value*k*exp(-(x*x/l+y*y/l))-m
```

where k,l and m are constants which adjust the extent and amount of change and x ranges from -xsize/2,xsize/2 and y ranges from -ysize/2,ysize/2.

(f) CONTRAST: If simple contrast enhancement is desired then generate intensity histogram, clip histogram on hi/low criteria, set up point process to set pixels below low criteria to 0 and above high criteria to 255. All other pixels are multiplied to increase their value so that they span the range of 0 to 255.

(g) OUTPUT: If a range of monochrome pixel values is to be highlighted then use pseudocoloring.

(h) AREA PROCESS: If it is desired to remove noise, reduce blurring, smooth an image, perform object detection by matching images, measure image properties, determine object edges, perform spatial filtering, then area processing is used.

(i) CONVOLUTION: If spatial filtering is required then perform convolution.

(j) CONVOLUTION: If finding image features then perform convolution.

(k) PERFORMANCE: If convolving an area of size X by Y with a kernal of size n by m then performance is X*Y*n*m multiplies and adds.

(l) PROBLEM: If using recursive or infinite response filter then it is difficult to understand or predict filter response.

(m) PROBLEM: If convolving then border may be unintelligible.

(n) IMPLEMENTATION PROBLEM: If convolving then pixel value resolution level may overflow or underflow.

(o) IMPLEMENTATION SOLUTION: If values overflow, then scale result.

(p) IMPLEMENTATION PROBLEM: If using convolution then there are implementation issues.

(q) IMPLEMENTATION PROBLEM: If area processing is being performed then there is a high likelihood of encountering issues of both internal accuracy and interpretation at edge of image.

(r) META KNOWLEDGE: If applying convolution then use it as a matched filter or spatial filter.

(s) META KNOWLEDGE: If using a matched filter then convolution kernel is essentially a small image of what is to be detected or amplified.

(t) CHARACTERIZATION: If an edge exists then there is a sudden increase or decrease in image intensity.

(u) DETECTION: If attempting to perform vertical edge detection then use convolution (correlation) kernel

$$\begin{vmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{vmatrix}$$

followed by a yes/no decision based on threshold point. If attempting to perform horizontal edge detection then use convolution (correlation) kernel

$$\begin{vmatrix} -1 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \end{vmatrix}$$

followed by a yes/no decision based on threshold point.

(v) METAKNOWLEDGE: If searching for a pattern then use convolution kernel "Kernel Pattern" followed by a yes/no decision based on pattern criteria.

(w) SOLUTION: If selection of high spatial frequencies is desired then use kernel

$$\begin{vmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{vmatrix}$$

(x) CHARACTERIZATION: If edge exists then it has high spatial frequency.

(y) SOLUTION: If locating higher spatial frequencies in image is required to be selectively boosted, then use kernel

$$\begin{vmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{vmatrix}$$

This may cause image to look sharper and noisier

(z) If kernel matches lower spatial frequencies then image blurs or smooths.

(aa) If a certain band of frequencies is to be selected and detected then build a kernel that selects that frequency. The art of convolving is in choosing the kernel with the proper frequency characteristics.

(ab) If a better signal-to-noise ratio is needed for detection of image elements or detection of features with less computation, then consider nonlinear area processes.

(ac) If strength and orientation of edges in image is rough then use Sobel operators.

(ad) If Sobel operators are applied then use intensity for edge strength and color for orientation in a two dimensional display.

(ae) If Sobel operators are employed they are a form of first derivative (oriented) edge finders.

(af) If employing Sobel operators, the two kernels X and Y are

$$X: \begin{vmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{vmatrix} \qquad Y: \begin{vmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{vmatrix}$$

Then

$$\text{strength} = \{ X*X + Y*Y \}^{.0.5}$$

$$\text{orientation} = \arctan\{Y/X\}$$

(ag) If machine vision algorithms are being developed then consider the first step with certainity (x) to be a Sobel gradient.

(ah) If using Sobel operators and performance is too computationally intensive then consider an approximation of the Sobel kernels.

(ai) If there is spot noise in image then use a median filter.

(aj) Classification: If determining the constituents of an element of an image.

Additional knowledge is needed.

(ak) If there is a connected group of pixels with the same value then the group is an element.

(al) If processing is computationally intense then booleanize (threshold) tje image, search top to bottom looking for areas that have more than N connected pixels.

(am) If thresholding then small spots of noise may be introduced.

(an) If classifying, record number of pixels in area and use for further classification.

(ao) If finding all connected pixels in an element then use a recursive search.

(ap) If image-element classification scheme is desired use both area and point processes to develop list of properties for a multi-dimensional space and partition space to perform the classification.

(aq) If spatial arrangement of pixels os to be changed then use geometric processes.

(ar) If image correction for distortion by camera optics or viewpoint is required then use geometric processes.

(as) If rotation, stretching, translation or warp of image is needed then use geometric process.

(at) If geometric transform is employed this introduces possible gaps between output pixels.

(au) If gaps between output pixels are undesirable then invert (reverse) the mapping equations and scan the destination area. At each point of the destination area use the inverted equations to fetch a source point. Source points generated by the inverse map that lie between pixels in the source area must have their value approximated by using the value at the nearest source pixel (possibly producing sudden intensity changes and a blocky appearance) or by interpolating source values (yielding smoother results).

(av) If pixel represents a rectangular rather than square area then compensate by adjusting the transformation equations.

(aw) If capturing a static image then sum N consecutive image frames and divide sum by N to reduce noise introduced by sensor.

(ax) If noise is gaussian and uncorrelated frame to frame then improvement in signal to noise ratio will be of order square root of N.

(ay) If motion detection is needed then subtract frames to approximate a time differentiation.

A pragmatic knowledge engineering approach to the literature of pattern recognition, image processing, integrated signal processing, detection, classification, recognition, etc., with the goal of elicitation of rules and context markers for the use of the various rules and methodologies would be a first step in organizing and putting together a large knowledge base which is a precursor to any serious attempt at automated image processing algorithm development. Along with this, the elicitation of expert image processing algorithm developers must be done. Clearly, certain aspects of this elicitation by the very nature of the work involved must be classified at a high level.

As an example, in the first case, the list in Table 2 of image processing transforms was developed during Phase I of the project in order to elicit the image algebra primitives and as a basis for which a knowledge engineering approach to level 3 image processing could be mounted. Some very direct questions from the knowledge engineering perspective can be asked.

- In which contexts should each transform be used?
- What is the certainity of the result as a function of the certainity of the inputs or as a function of the transform itself?

- Within categories of transforms, why would one transform be chosen over others? Which tend to be more robust independent of context? How is robustness measured? What are the performance issues and computational complexity issues? Are backup approximations available? Can they be generated from a uniform procedure?

- Additional questions from a knowledge engineering approach to automated image processing algorithm development are listed below:

  What are the naturally occurring sequences of these transforms across current image processing algorithms? What are the precedence orderings? When decomposed into image algebra primitives, what are the naturally occurring sequences of primitives? When used in image processing algorithms, what are the decision criteria associated with post transform application? Are their default *decision criteria* associated with the use of these transforms? What is the criteria by which an expert chooses the use of one transform over another? What are the approximations used if performance criteria are stringent? Which can be rewritten (or modeled) as neural net applications?

TABLE 2 IMAGE PROCESSING TRANSFORMS

Alpha Conditional Bisector
Array Grammars
Asynchronous Interaction
Background Subtraction
Bandwith Compression Via Iterative Histogram Modification
Bernstein Polynomial Approximation
Best Plane Fit (BPF, Sobel, Roberts, Prewitt, Gradient)
Boundary Finder
Boundary Segmenter
Chain Code Angle Determiner
Closing (Black and White)
Closure Operation
Connection Operator
Connectivity Number
Convex Hull
Convexity Number
Convolution Transform
Co-occurrence Matrix
Cumulative Angular Deviant Fourier Description
Cue Transform
Digitilization
Dilation (Black and White)
Directional Gradient Transform
Discrete Cosine Transform
Discrete Fourier Transform
Discrete K-L Feature Selection
Discrete Picture Transform
Edge Detection By Gradient
Erosion (Black and White)
Fields Without Interaction (Black and White)
Fourier Feature Normalization
Fourier Features
Frei-Chen Thresholding Strategy
Geometric Correction
Gibbs Ensemble (Black and White)
Gradient Directed Segmentation
Gradient Edge Operators
Grey Scale Correction
Grey Scale Histogram
Grey Scale Transformation
Haar Transform (Haar Functions)
Hadamard Transform (Walsh Functions)
Heukel Edge Operator

TABLE 2 IMAGE PROCESSING TRANSFORMS (continued)

Hierarchial Edge Detection
Histogram Equalization
Hit and Miss Transformation
Hotelling Transform (Karhunen-Loeve Transform)
Hough Method for Line Detection
Image Coding by DPCM
Intersection Function (Black and White)
Kirsch Operator
Linear Erosion
Linear Filtering
Local Neighborhood Transform
L-U Decomposition
Magnitude Gradient Transform
Markov Random Field (Black and White)
Medial Axis Skeleton
Minkowski Functionals
Moments of Silhouette
Morphological Covariance
Noise Removal by Smoothing
Opening (Black and White)
Parallel Interactive Scene Labeling
Perimeter Estimation by Dilation
Planar Size Distribution
Pyramid
Projection Estimation by Dilation
Quad Trees (and Binary Trees)
Quantization
Random Field
Raster Tracking
Region Growing
Relaxation Labeling
Rotation Invariant Field (Black and White)
Sequential Thinning
Serial Relaxation
Shape Grammars
Simple Boundary Segmenter
Size Criteria
Size Distribution in Length
Skeleton
Skiz Transform (Exoskeleton)
Slant Transform
Spoke Filter
Stacked Image Data Structure
Straight Line Detection by Linear Filter

TABLE 2 IMAGE PROCESSING TRANSFORMS (CONCLUDED)

Subtraction of Laplacian
Superslice
Superspike
Syntactic Noise Reduction
Template Matching
Template Matching by Cross Correlation
Thickening
Thinning (Morphological)
Thresholding
Transform Encoding
Translation Transform
Tree Grammars
Tree Search Labeling Algorith
Umbra Transform
Variable Sized Hexagons
Walsh Feature Representation
Zucker and Hammel Three Dimensional Edge Operators

SECTION VI
CONCLUSION

A mathematical environment has been developed which allows for the expression of various algorithms employed in image processing. Here, algorithms appear as strings in an operational calculus, where each operator is ultimately expressed as a string composed of some collection of elemental, or "basis" operators. The action of the string upon a collection of input images is determined by function composition. Operations such as convolution and dilation usually determined in a pointwise manner are given as closed-form expressions in terms of low-level operations of the algebra.

A unified framework for the orderly expression of the diverse algorithms that play roles in image analysis has been given. Indeed, it provides a theory behind the block diagram technique which serves as a universal language in which investigators from diverse backgrounds may find a common understanding.

The view taken herein is that images are part of a function space. Once the function space is specified, the determination of the desired algebra depends on the collection of operations we wish to induce from the structure of the domain and range spaces. In digital image processing, the domain space is usually considered to be the integral lattice ZxZ, while the range space is either the set, R, of real numbers or the set, Z, of integers. Of significance is that when the underlying problem of algebra development is viewed from the appropriate perspective, the particulars of the subject matter involved are of minor importance; rather, it is the methodology of development that is paramount.

Advanced Artificial Intelligence and knowledge engineering research have been presented with the goal of introducing a structured approach to the development of intelligent weapon systems in general and to automated image processing algorithm development specifically. Current interest in level 3 image processing (integration of knowledge and intelligence into image processing algorithms) dictates a cost effective approach with high probability of success. Emerging results in knowledge engineering, cognitive science, system science, and computer science appear to indicate that a formal approach to the engineering of intelligent systems in on the horizon. The design of an intelligent system laboratory is recommended and an initial outline of such a facility is given. Research, development, and prototype support for such a facility is highly recommended.

REFERENCES

1. E. Dougherty and C. R. Giardina, Matrix Structured Image Processing, Prentice Hall, NJ, 1987.

2. E. Dougherty and C. R. Giardina, Image Processing Continuous to Discrete, Volume 1, Prentice Hall, NJ, 1987.

3. E. Dougherty and C. R. Giardina, Morphological Methods in Image and Signal Processing, Prentice Hall, NJ, 1988.

4. G. J. Klir, Architecture of Systems Problem Solving, Plenum Press, New York, 1985.

5. R. A. Orchard, E. J. Reese and M. R. Tausner, "Foundations of Knowledge Engineering," Proceedings of the 4th International Symposium on Modelling and Simulation, University of Arizona, Tuscon, AZ., January 1987.

6. R. A. Orchard, M. R. Tausner, "The Foundational Role of General Systems Formalizations in Knowledge Engineering," Proceedings of the 1987 Conference of the International Society for General Systems Research, Budapest, Hungary, 1987.

7. R. A. Orchard, M. R. Tausner, "Transformation of Knowledge in General Systems Formalizations to Expert System Representations," Proceedings of the 1987 Conference of the International Society for General Systems Research, Budapest, Hungary, 1987.

8. R. A. Orchard and M. R. Tausner, "Foundations of Knowledge Engineering," Vol. 1: Fundamental Principles, Prentice Hall, Englewood Cliffs, NJ, 1989. (To be published in 1989.)

9. R. A. Orchard, E. J. Reese and M. R. Tausner, "Foundations of Knowledge Engineering," Vol. 2: Fundamental Principles of Knowledge Elicitation, Prentice Hall, Englewood Cliffs, NJ, 1989.

10. E. J. Reese, Knowledge Elicitation Notes, Knowledge Engineering Seminar at the College of Staten Island, Institute for Advanced Knowledge Engineering, Sun Institute of NLP, Indian Rocks Beach, FL, 1986.

11. M. A. Reese, Knowledge Elicitor's Notebook, Draft made available by the author, 1986, Institute for Advanced Systems, 1200 Rt. 23, Butler, NJ 07442.

12. M. R. Tausner, "A Taxonomy of Reasoning Strategies for Expert Systems," 1985 Proceedings of the International Conference on the Management and Performance Evaluation of Computer Systems, Dallas, Texas, December 1985.

13. M. R. Tausner and R. A. Orchard, "On the Design Of A Knowledge and Expert Based System for General System Problem Solving," Proc. 1985 SGSR International Conference, Los Angeles, CA, May 1985, pp. 369-376.

14. M. R. Tausner and R. A. Orchard, "A Formalism For Expert Systems With Front-End/Back-End Architecture," Proceedings 1986 SGSR International Conference, Philadelphia, Pennsylvania, May 1986.

15. M. Tausner, Formalization and Representation of Expert Systems, University Microfilms, Ann Arbor, MI, 1986.

16. W. K. Estes, The Information Processing Approach to Cognition: A Confluence of Metaphors and Methods, W. K. Estes, Ed., Handbook of Learning and Cognitive Processes Volume 5: Human Information Processing, Lawrence Erlbaum, 1978.

17. A. Newell and H. A. Simon, Human Problem Solving, Prentice-Hall, NJ, 1972.

18. L. Von Bertalanffy, General System Theory, Penguin, London, 1968.

19. N. Wiener, Cybernetics, Wiley, New York, 1948.

20. T. Moto-Oka, Ed., Fifth Generation Computer Systems, North-Holland, Amsterdam, 1982.

21. B. R. Gaines and M. L. G. Shaw, "Hierarchies of Distinctions as Generators of System Theories," A. W. Smith, Ed., Proceedings of the Society for General Systems Research International Conference, Intersystems, CA.

22. D. Gregory, "Defining Expert Systems," IEEE Transactions on Systems, Man, and Cybernetics, SMC-16, No. 6, 1986.

23. R. Cristensen, Entropy Minimax Source Book, Vol. 1: General Description, Entropy Ltd., Lincoln, MA, 1980.

24. M. Higashi and G. J. Klir, "Measures of Uncertainty and Information Based Possibility Distributions," International Journal of General Systems, 9, No. 1, 1983.

25. M. Higashi and G. J. Klir, "On the Notion of Distance Representing Information Closeness: Probability and Possibility Distributions, "International Journal of General Systems, 9, No. 2, 1983.

26. G. A. Shafer, A Mathematical Theory of Evidence, Princeton University Press, Princeton, NJ, 1976.

27. S. Watanabe, Pattern Recognition: Human and Mechanical, John Wiley, NY, 1985.

28. R. R. Yager, "A Foundation for a Theory of Possibility," Journal of Cybernetics, Vol. 10, Nos. 1-3, 1980.

29. L. A. Zadeh, "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes," IEEE Transactions on Systems, Man, and Cybernetics, SMC-1, No. 1, 1973.

30. F. Zwicky, Discovery, Invention, Research Through the Morphological Approach, Macmillan, Toronto, 1966.

31. W. R. Ashby, An Introduction to Cybernetics, Wiley, NY, 1956.

32. L. Bandler, They Lived Happily Ever After, Meta Publications, Meta Publications, Cupertino, CA, 1978.

33. R. Bandler and J. Grinder, The Structure of Magic, Vol. 1, Science and Behavior Books, Palo Alto, CA, 1975.

34. R. Bandler and J. Grinder, The Structure of Magic, Vol. 2, Science and Behavior Books, Palo Alto, CA, 1976.

BIBLIOGRAPHY

Boulding, K.L., "General Systems Theory - The Skeleton of Science," Management Science, Vol. 2, 1956. Reprinted in General Systems, 1956.

Chandrasekaran, B., "Generic Tasks in Knowledge Based Reasoning: High-Level Building Blocks for Expert System Design," IEEE EXPERT, Vol. 1, No. 3, 1986.

Dilts, R., Grinder, J., Bandler, R., Bandler, L.C., and DeLozier, J., Neuro-Linguistic Programming: Vol 1, The Study of the Structure of Subject Experience, Meta Publications, Cupertino, CA, 1980.

Foerster, H. Von, "On Self-Organizing Systems and Their Environments," Self-Organizing Systems, Yovitz and Cameron, Eds., Pergamon, NY, 1960.

Gaines, B. R. and Shaw, M. L. G., "Emotion: A Systemic Development of the Personal Construct Model," Proceedings of the Personal Construct Network Conference, Cincinnati, OH, June 22-24, 1984.

Hart, A., Knowledge Acquisition for Expert Systems, McGraw - Hill, NY, 1986

Hayes-Roth, F., Waterman, D. A., and Lenat, D. B., (Eds.) Building Expert Systems, Addison-Wesley, Reading, MA, 1983.

Kelly, G. A., The Psychology of Personal Constructs, Norton, NY, 1955.

Klir, G. J., An Approach to General Systems Theory, Van Nostrand Reinhold, NY, 1969.

Lofgren, L. "Complexity of Description of Systems: A Foundational Study," International Journal of General Systems, Vol. 3, No. 4, 1977.

Minsky, M., "A Framework for Representing Knowledge," in The Psychology of Computer Vision, P. Winston, Ed., McGraw-Hill, NY, 1975.

Pask, G., Scott, C. E. and Kalikourdis, D.,"A Theory of Conversations and Individuals," International Journal of Man-Machine Studies, Vol. 5, 1973.

Pask, G., "Developments in Conversation Theory," International Journal of Man-Machine Studies, Vol. 3, 1980.

Pask, G. and Gregory, D., "Conversational Systems," Human Productivity Enhancement, J. Zeidner, Ed., Praeger, NY, 1986.

Rauch-Hindin, W. B., Artificial Intelligence in Business, Science, and Industry, Vol. 1, Prentice-Hall, 1986.

Zadeh, L. A., "Fuzzy Sets," Information and Control, Vol. 8, No. 3, 1965.