Rensselaer Polytechnic Institute

Troy, New York 12181

RPI TECHNICAL REPORT MP-81
ANNUAL REPORT
May 1, 1980 to April 30, 1981

DATA ACQUISITION AND ANALYSIS OF RANGE-FINDING SYSTEMS FOR SPACE CONSTRUCTION

RPI TECHNICAL REPORT MP-81

ANNUAL REPORT FOR

May 1, 1980 to April 30, 1981

DATA ACQUISITION AND ANALYSIS OF RANGE-FINDING SYSTEMS FOR SPACE CONSTRUCTION

NATIONAL AERONAUTICS
and
SPACE ADMINISTRATION

Langley Research Center

GRANT # NAG 1-61

C.N. Shen

RENSSELAER POLYTECHNIC INSTITUTE
TROY, NEW YORK

1981

RPI TECHNICAL REPORT MP-81

ANNUAL REPORT FOR

May 1, 1980 to April 30, 1981

DATA ACQUISITION AND ANAL'SIS OF RANGE-FINDING SYSTEMS FOR SPACE CONSTRUCTION

NATIONAL AERONAUTICS
and
SPACE ADMINISTRATION

Langley Research Center

GRANT # NAG 1-61

C.N. Shen

RENSSELAER POLYTECHNIC INSTITUTE
TROY, NEW YORK

1981

TABLE OF CONTENTS

Abstract

For space missions of future, completely autonomous robotic machines will be required to free the astronauts from routine chores of equipment maintenance, servicing of faulty systmes, etc. and to extend human capabilities in hazardous environments full of cosmic and other harmful radiations. In places of high radiation and uncontrollable ambient illuminations, T.V. camera based vision systems cannot work effectively. However, a vision system utilizing directly measured range information with a time of flight laser rangefinder, can successfully operate under these environments. Such a system will be independent of proper illumination conditions and the interfering effects of intense radiation of all kinds will be eliminated by the tuned input of the laser instrument.

Especially important is the capability of the laser based vision system to afford a 3-dimensional description of the environment. Known objects can be recognized and their correct spatial locations and orientation can be determined with respect to the sensing instrument, and the robot, which can thus perform its tasks in a three dimensional space. Vision systems using 2 dimen- images as the primary input have not met much success in 3-dimensional scene description because of the difficult problems associated with indirect range calculation methods.

Processing the range data according to certain decision, stochastic estimation and heuristic schemes, the laser based vision system will recognize known objects and thus provide sufficient information to the robot's control system which can develop strategies for various objectives.

I. INTRODUCTION

Various schemes for data acquisition, edge detection, plane surface extraction by clustering, and noise removal from the raw input data were discussed earlier in [5]. This report extends the previous work of surface description. A procedure is developed for defining the surfaces by regression planes passing through the three dimensional points clustered by the Hierarchical Clustering Scheme using orthogonal surface slopes along orthogonal directions.

Continuous inner edges are calculated by calculating the possible lines of intersections of the planes found from the surface description procedure. This procedure for calculation of edges of objects can be applied only if the two surfaces forming it are visible from the location of the laser sensor. The edges detected by the Rapid Estimation Scheme, [1], are in the form of scattered points on both sides of the edges and some of the edges especially the inner edges, may be missed, therefore a more elaborate procedure than the Rapid Estimation Scheme alone was required. Thus the inner edges are calculated here from the intersection of best fitted planes. The calculated edges are verified in the case of inner edges also detected by the Rapid Estimation Scheme.

For the description of a scene in terms of meaningful objects, heuristic rules are developed for combining planes and edges found above into consistent subjects so as to define meaningful objects, known or unknown. Based on the convexity properties of plane faceted objects different planes are linked together if there is a convex edge between them. Similarly, half edges with one end on a virtual vertex are linked together if the unit vectors along them are colinear. Heuristic rules are also developed for linking half planes resulting from an occlusion.

Noise removal from the raw data and estimation of range slopes and, in turn, actual surface slopes is achieved by the applicaton of a two-dimensional smoothing algorithm, [2]. The smoothing algorithm used three different smoothness measures which were reduced to a quadratic form in the state vector consisting of the function values, two first partial and the first mixed partial derivatives at each node point of a uniform two-dimensional grid. The weighting matrix was, however, derived by a complex reduction procedure.

To simplify the derivation of the results, the weighting matrix form of the smoothing integrals was avoided and the two-dimensional smoothing problem was formulated as an optimal control problem by incorporating a control term in the recursive model of the spline functions. Essentially the same results were obtained for the smoothing algorithms as in, [2], thus providing mathematical rigor and verification.

To employ different spline function with certain interesting smoothing or edge enhancement properties a general method for the calculation of bases of different spline functions was developed. By using this approach, it is hoped that the derivation of different spline functions will be simplified.

II. SUMMARY OF PROGRESS

The first part of this report presents a scene analysis/object recognition system that utilizes directly measured range information from the scene of interest and provides a three-dimensional description of the scene in terms of known and unknown objects in their true relative positions and orientations. The range to a large number of points in the scene is measured by using a time of flight laser rangefinder in which range is obtained from the phase difference between the transmitted and reflected laser beams. The rangefinding system, in fact, measures the spherical coordinates of various points on the surfaces of objects by scanning in the elevation and azimuth directions in a suitably defined coordinate system, Figure 1. The range readings are corrupted with measurement noise.

By uniform increments of the two pointng angles of the laser beam over suitable intervals, a matrix of range values is obtained, the rows and columns of which are indexed with constant values of elevation and azimuthal angles. Range slope in the inpath, i.e., by the variation of elevation angle only, and in the crosspath direction, i.e., by the variation of azimuthal angle only, is defined as the first difference of range divided by the corresponding increment in pointing angles β and θ respectively. The range data is processed along columns and rows to detect the presence of edges in the scanned scene which manifest themselves as sudden changes in range or range slope between adjacent points. The edges are thus extracted by a Rapid State Estimation Scheme (RES), [1], that uses Kalman filters and decision trees.

The observed range data is uniformly spaced in elevation and azimuthal angles, but if projected onto any of the planes xy, yz, or zx shown in Figure 1, the regularity of data spacing is lost. Thus the range and two range slopes

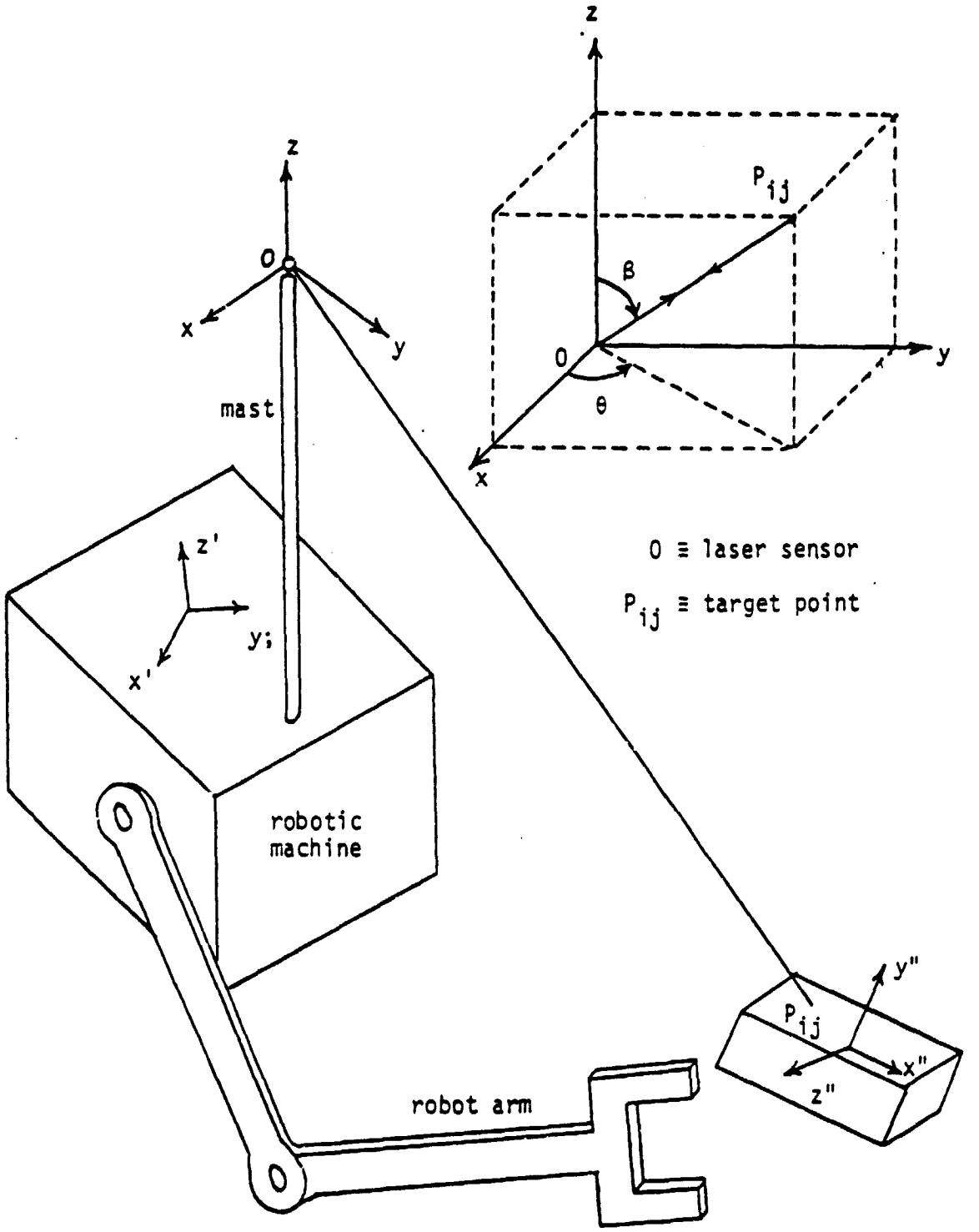


Fig. 1. Coordinate Systems of the Laser Rangefinding System

are smoothed in the original spherical coordinates before any scene analysis procedure is applied. This is accomplished by using a two-dimensional smoothing scheme [2], in the region free of the edges of objects detected above. The smoothed range and two range slopes are transformed to the actual slope of the surfaces of the objects at their observed data points.

Using the fact that the slopes of a plane along any two orthogonal directions and with respect to a reference plane are constant at all of its points, the data points belonging to plane surfaces can be separated. Using the two slopes along orthogonal directions as the two components of a feature vector, the plane surfaces are extracted by using a hierarchical agglomerative clustering procedure, [3], [4].

The continuous edges of a plane faceted objects are then obtained as the intersections of their plane surfaces found above. Using heuristic rules, the edges and planes belonging to individual objects are grouped in an ordered and consistent manner so as to facilitate the object reconstruction.

The characteristics of an object which are important for this purpose are found to be the convexity and colinearity of edges. Methods for determining if an edge is convex or concave and if two edges are colinear are developed. These methods are consistent, regardless of the viewers point of observation. If parts of the image of the workscene in the input information exhibit certain properties, then these parts will be grouped together as an object.

For recognition of known objects, three-dimensional models in the form of ordered lists of features of objects are stored in the computer memory. The essential contents of these models are the specification of number and lengths of edges meeting at a vertex, the spatial angles between these edges, the edges bounding a face, etc. This stored information can then be utilized to recognize known objects and determine their positions and orientations.

The second part of the report deals with the problems of two-dimensional filtering and smoothing of data by using spline functions. A number of approaches to the 2-D data processing problem have been proposed in the literature recently. These algorithms typically fall into two classes, those based on original image model [6-8], and those found by minimizing objective function without explicit model [2], [9-12]. The approach taken here is closer to the technique of numerical spline analysis, that is, no prior image model is assumed. However, the focus here is on achieving a recursive algorithm which can be implemented on-line.

The purpose here is twofold: a) along the same line as developed in [2], the results were extended to a wider class of possible signals; b) the fundamental principles and limitations on recursive smoothing splines are explored, which were not discussed in the previous research.

The state space notation of signal is used for it not only keeps the notation simple, compact, and unique, but also provides the crucial point in the recursive processing. Another factor involved in the formulation of recursive computation is the proper selection of the objective function.

In particular, it is hoped that the expositions will shed some light on the peculiar probmes of recursive algorithms for 2-D smoothing splines, which do not appear in 1-D data processing.

In order to get better data fitting, p.H.p.'s of continuity greater than the first derivatives are desirable. However, the derivation of high order p.H.p. using direct method is quite complex in general. A method using cardinal bases, [13], is also described which results in a systematic, simple, and straightforward calculation procedure.

III. DETAILED PROGRESS REVIEW

The progress made during the past year in the ongoing research is described here. There are essentially two major phases of research, the first one dealing with the three-dimensional scene analysis problem and the other is on the processing of two-dimensional signals.

TASK A. Scene Analysis Based on Laser Rangefinder Data Processing for Space Robotics.

This part of the report deals with the development of a possible vision/environment sensing system that can be used by autonomous robotic systems of future, especially in space probes where proper illumination for T.V. camera based vision systems will be difficult.

1. Feature Extraction and Surface Description

Application of the two-dimensional smoothing algorithm [2], using spline functions, in the regions between the discrete edges, detected by the Rapid Estimation Scheme, [1], yields smoothed values of range r_{ij} and the two range slopes $\frac{\partial r_{ij}}{\partial \beta_{ij}}$ & $\frac{\partial r_{ij}}{\partial \theta_{ij}}$ and the measured pointing angles β_{ij} and θ_{ij}, at an observed point P_{ij}, Figure 2a, the actual slope of the surface at P_{ij} with respect to some coordinate planes xy, yz, zx of the laser rangefinder and the three pairs of reference directions employed are as shows by $\left\{\frac{dz}{dx}, \frac{dz}{dy}\right\}$, $\left\{\frac{dx}{dy}, \frac{dx}{dz}\right\}$ and $\left\{\frac{dy}{dz}, \frac{dy}{dx}\right\}$.

a. Segmentation of Range Data: Recursive Hierarchical Clustering

The idea behind the transformation from range slopes to the actual surface slopes of the objects is that the data points belonging to a plane surface will show some slopes along two fixed orthogonal directions and thus can be separated from the rest of the points. Hence, for all of the observed data points, two

slopes along any two fixed directions w·r·t a reference plane cnan serve to separate the total observed data into plane surfaces. However, there can arise a serious problem due to some slopes becoming singular or achieving arbitrarily large magnitudes. This will happen when the corresponding unknown plane has a high inclination w·r·t the reference plane (Figure 2b).

Thus for avoiding the problem of singularities in the transformed surface slopes it was decided to calculate the same along six different directions, Figure 2a. For a particular observed point only a pair of surface slopes w·r·t one of the coordinate planes is calculated. Possible pairs of slopes are:

$$\left\{\frac{dz_{ij}}{dy_{ij}}, \frac{dz_{ij}}{dx_{in}}\right\}, \left\{\frac{dy_{ij}}{dx_{ij}}, \frac{dy_{ij}}{dz_{ij}}\right\} \quad \text{and} \quad \left\{\frac{dx_{ij}}{dy_{ij}}, \frac{dx_{ij}}{dz_{ij}}\right\} \tag{1}$$

As can be noted, only three of the above quantities are unique. Denoting the range slopes $\frac{\partial r_{ij}}{\partial \theta_{ij}}$ by V_{ij}^{c} and $\frac{\partial r_{ij}}{\partial \beta_{ij}}$ by V_{ij} three of the above surface slopes may be written in terms of the measured and smoothed quantities:

$$S_1 \triangleq \frac{dz}{dy} = \frac{-(\tan\beta)V - r}{[\cos\theta - r \tan\beta \sin\theta / V^c]V - r\tan\beta\cos\theta} \tag{2}$$

$$S_2 \triangleq \frac{dz}{dx} = \frac{-[\tan\theta + r/V]V^c}{r\cos\theta + \sin\theta(1 - r\tan\beta/V)V^c} \tag{3}$$

$$S_3 \triangleq \frac{dy}{dx} = \frac{[\tan\theta + r/V^c]V + r\tan\theta\tan\beta}{[1 - r\tan\theta/V^c]v - r\tan\beta} \tag{4}$$

where each of the above expressions connect all the quantities at a single point. The remaining three surface slopes in Equation 1 may be obtained as the reciprocals of these quantities.

Thus, with each observed point of the scene, a two-dimensional slope vector is associated indexed by the particular plane w·r·t which the two slopes were calculated. Processing the whole data in this manner, 6 two-dimensional arrays corresponding to six different slopes are obtained. These arrays contain blank entries for points at which the particular slope was not calculated.

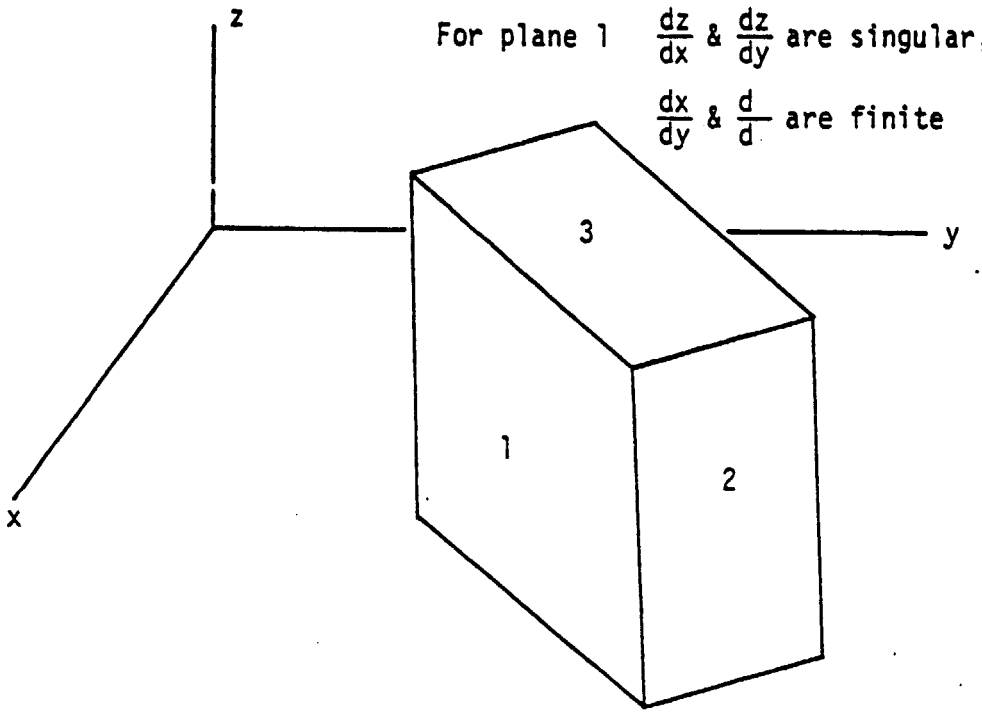


Fig. 2b Singularities in surface slopes

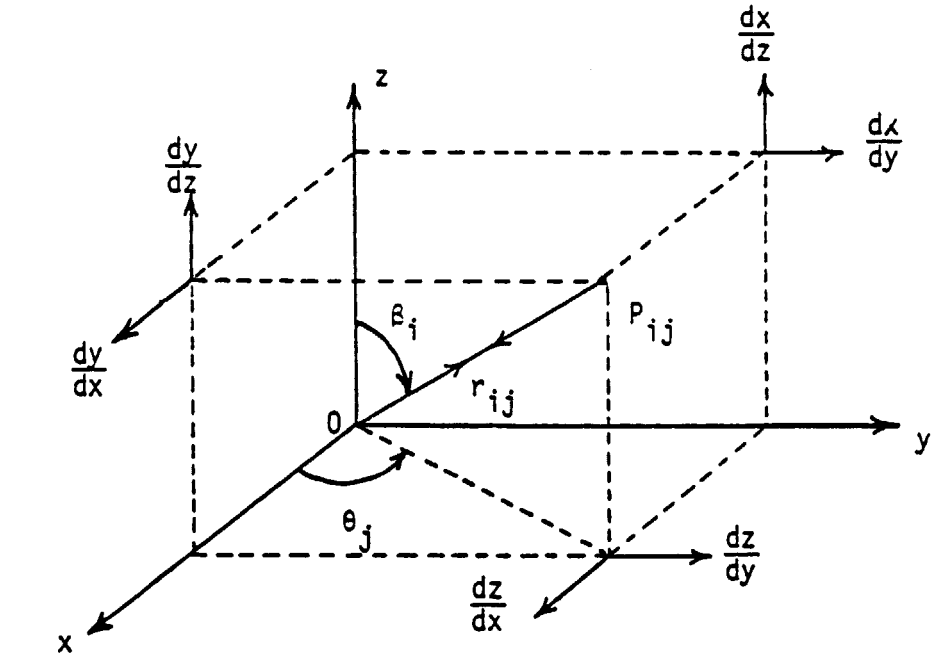


Fig. 2a Reference directions of six different surface slopes at a point

The observed data points are separated into different groups for different planes surfaces by using a recursive hierarchical clustering scheme reported earlier, [5].

b. Surface Estimation: Least Squares Plane Fitting

The output of the recursive clustering algorithm [5], consists of three different sets of clusters, depending on which pair of surface slopes was used to group the data points. For a cluster of three-dimensional points using surface slopes $\frac{dz}{dx}$ and $\frac{dz}{dy}$ a linear relation $z = a + bx + cy$, (a plane) is supposed to exist between their rectangular coordinates (x,y,z). Given 'n' number of points, the best values of the parameters a,b, and c are obtained by minimizing the squared error $\sum_{i=1}^{n} (z_i - z)^2 = \sum_{i=1}^{n} (z_i - a - bx_i - cy_i)^2$ (5)

Differentiating this error w·r·t a,b, and c, three euqations are obtained:

$$\frac{\partial E}{\partial a} = \sum_{i=1}^{n} (z_i - a - bx_i - cy_i)\ (-2) \tag{6}$$

$$\frac{\partial E}{\partial b} = \sum_{i=1}^{n} (z_i - a - bx_i - cy_i)\ (-2x_i) \tag{7}$$

$$\frac{\partial E}{\partial c} = \sum_{i=1}^{n} (z_i - a - bx_i - cy_i)\ (-2y_i) \tag{8}$$

Setting all derivatives equal to zero for $a = \hat{a}$, $b = \hat{b}$, and $c = \hat{c}$, the so-called normal equations of the least squares plane are:

$$\Sigma z_i = n\hat{a} + \hat{b}\ \Sigma x_i + \hat{c}\ \Sigma y_i \tag{9}$$

$$\Sigma x_i z_i = \hat{a}\ \Sigma x_2 + \hat{b}\ \Sigma x_i^2 + \hat{c}\ \Sigma x_i y_i \tag{10}$$

$$\Sigma y_i z_i = \hat{a}\ \Sigma y_i + \hat{b}\ \Sigma x_i y_i + \hat{c}\ \Sigma y_i^2 \tag{11}$$

where all summations are from i=1,....,n. Solution of this set of linear equations defines the best fitting plane as the triple $(\hat{a}, \hat{b}, \hat{c})$.

c. Continuous Edges: Calculation and Verification

The edge points detected by the Rapid Estimates Scheme, [1], are widely spaced, the spacing being dependent on the distance from the laser sensor and the increments of pointing angles used. Also because of measurement noise in the range readings there occur some false detections at few points. From Figure 3, it is clear that the outer edges will be detected easily, however, some of the inner edges may be missed for which the change of slope is small or gradual. Thus R.E.S. alone is not completely sufficient for extracting edges of the objects.

Having described the surfaces in the scene by the least squares regression planes, the continuous and exact inner edges can be obtained by calculating their lines of intersections. It is to be noted that this approach can be applied only for the inner edges where both the intersecting planes are visible to the laser sensor. For outer edges the best estimate is a straight line passing through the edge points lying in a three-dimensional space.

In Figure 4, the plane ϕ has been defined uniquely by its unit normal vector $\underline{n} = p\hat{i} + q\hat{j} + r\hat{k}$ and its distance from the origin, shown as d. $\hat{i}$, $\hat{j}$, $\hat{k}$ are the unit vectors and p, q, r are the components of $\underline{n}$ along these directions. The equation of the plane is written as $\underline{n}\cdot\underline{r} = d + px + qy + rz$

$$\underline{n}\cdot\underline{r} = d + px + qy + rz \tag{12}$$

which can be put in the form

$$z = \frac{d}{r} - \frac{p}{r}x - \frac{q}{r}y \tag{13}$$

$$= a + bx + cy \tag{14}$$

where $z = \frac{d}{r}$, $b = -\frac{b}{r}$, $c = -\frac{a}{r}$

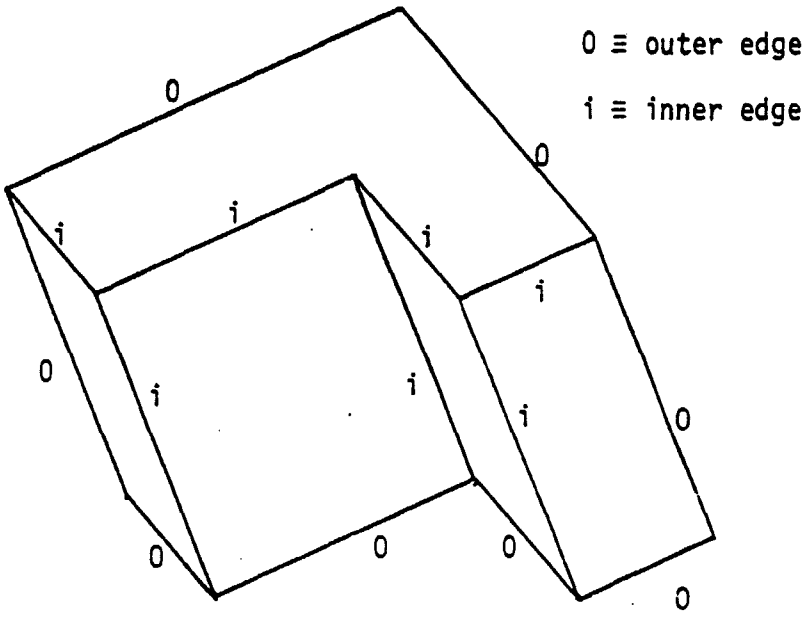


Fig. 3 Inner and outer edges of an object

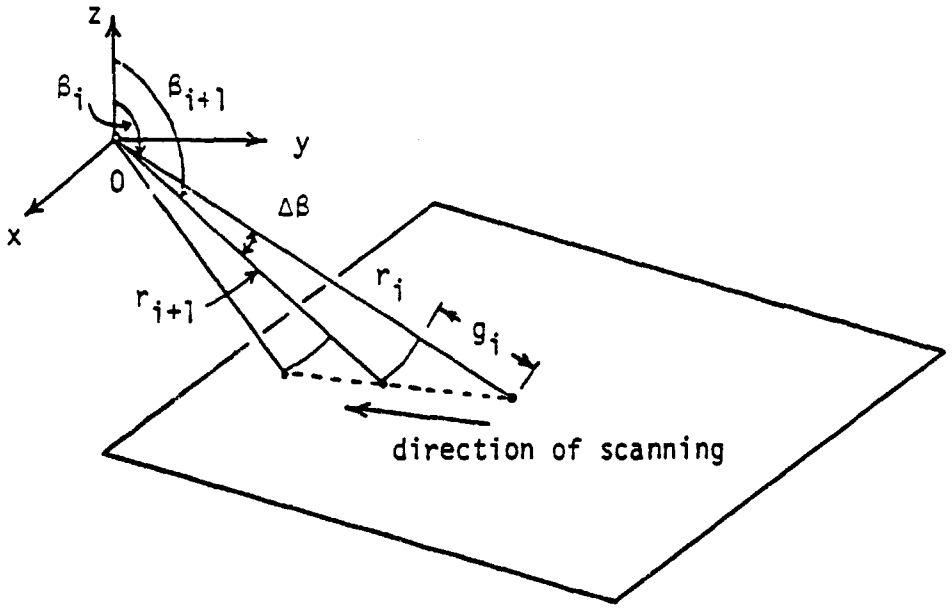


Fig. 3a Laser beam following a flat surface

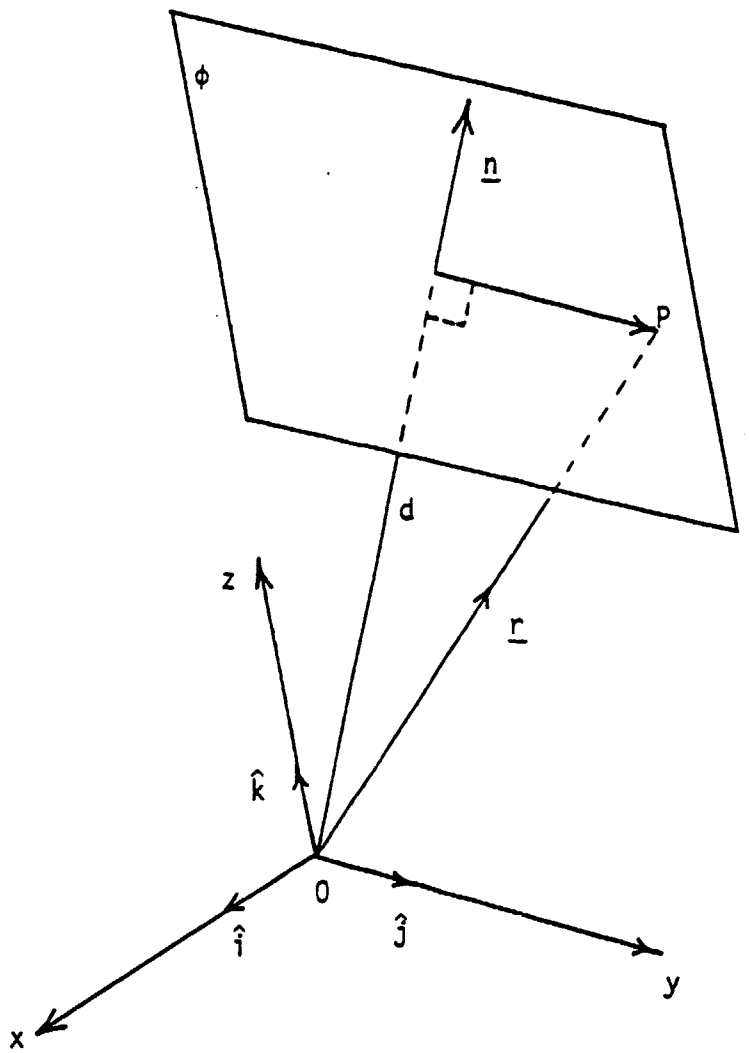


Fig. 4 A plane defined by its normal vector and distance from the origin

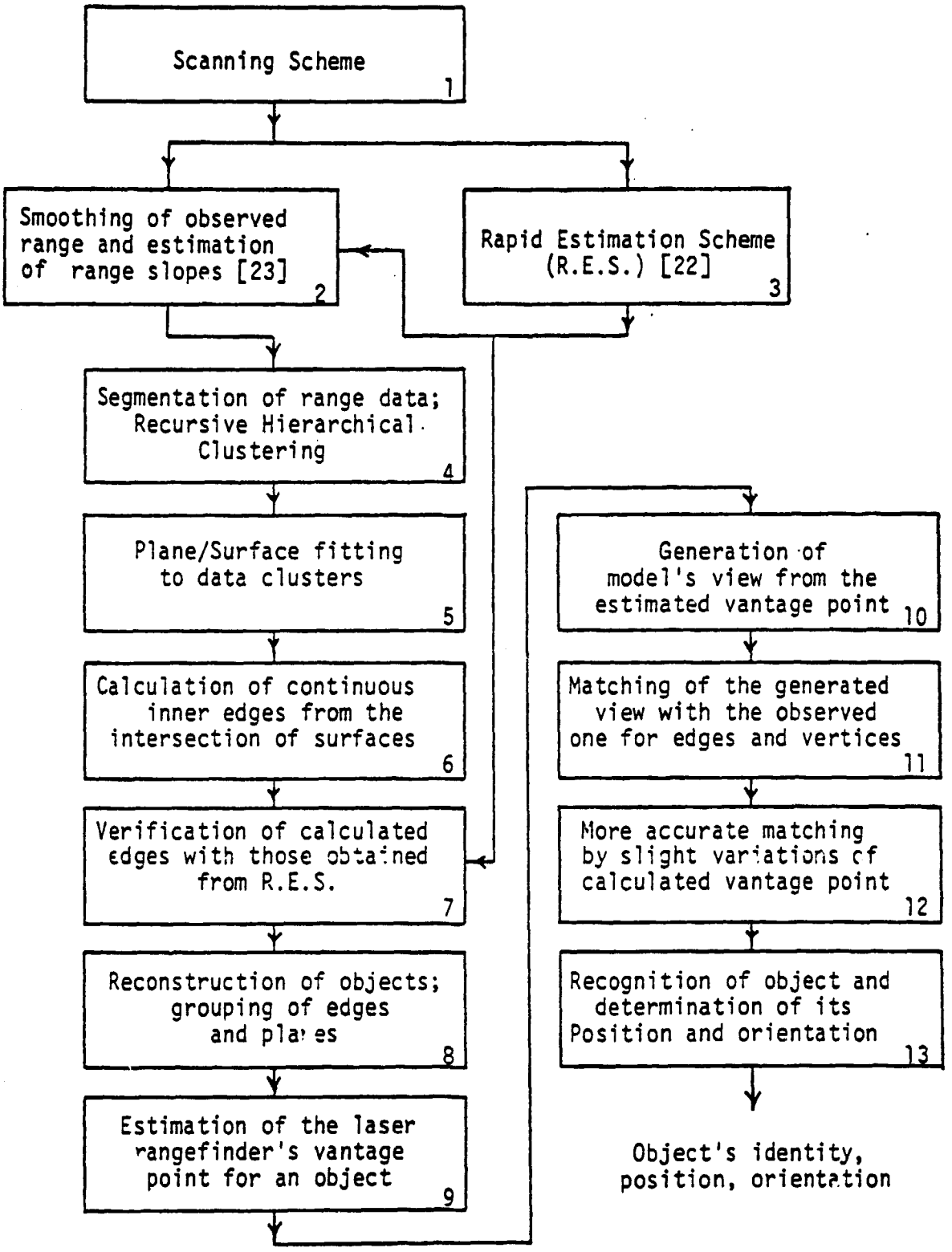


Fig. 5 Flow diagram of the 3-Dimensional Scene Analysis System

In Equation 14, parameter $a = \frac{d}{r}$ is a measure of the distance of the plane P from the origin. The equations of the planes obtained from the least squares method of surface estimation are in the form of triples $[\hat{a}, \hat{b}, \hat{c}]$.

Thus two planes $[a_1, b_1, c_1]$ and $[a_2, b_2, c_2]$ will intersect if

$$\frac{b_1}{b_2} \neq \frac{c_1}{c_2}$$

$\frac{b_1}{b_2} = \frac{c_1}{c_2}$ indicates that the two planes in question are parallel to each other and hence do not intersect. All the inner edges can, therefore, be obtained by the above procedure after the equations of the two planes forming it have been obtained.

2. Object Reconstruction and Recognition

In a paper on two-dimensional scene analysis [14], Guzman used heuristic rules to decompose a line drawing into objects. These heuristic rules were based on the types of vertices which occur in a line drawing.

Here we present a heuristic scheme for object reconstruction and formation based on input data containing depth information. This scheme will reconstruct plane faceted objects from a workscene described as edges, faces, and vertices in cartesian coordinates.

a. Object Reconstruction Using Heuristic Rules

Planes will be collected into the same object if they satisfy these rules:

1. Intersect at a convex edge.
2. Contain colinear half edges which have adjacent coplanar faces.
3. Intersect at a concave edge where one of the vertices of the edge is on the outside edge and the concave edge is not colinear with the outside edge.

These heuristics are pictorially described in Figure 6, where  indicates that the two faces connected are grouped into the same object. Figure

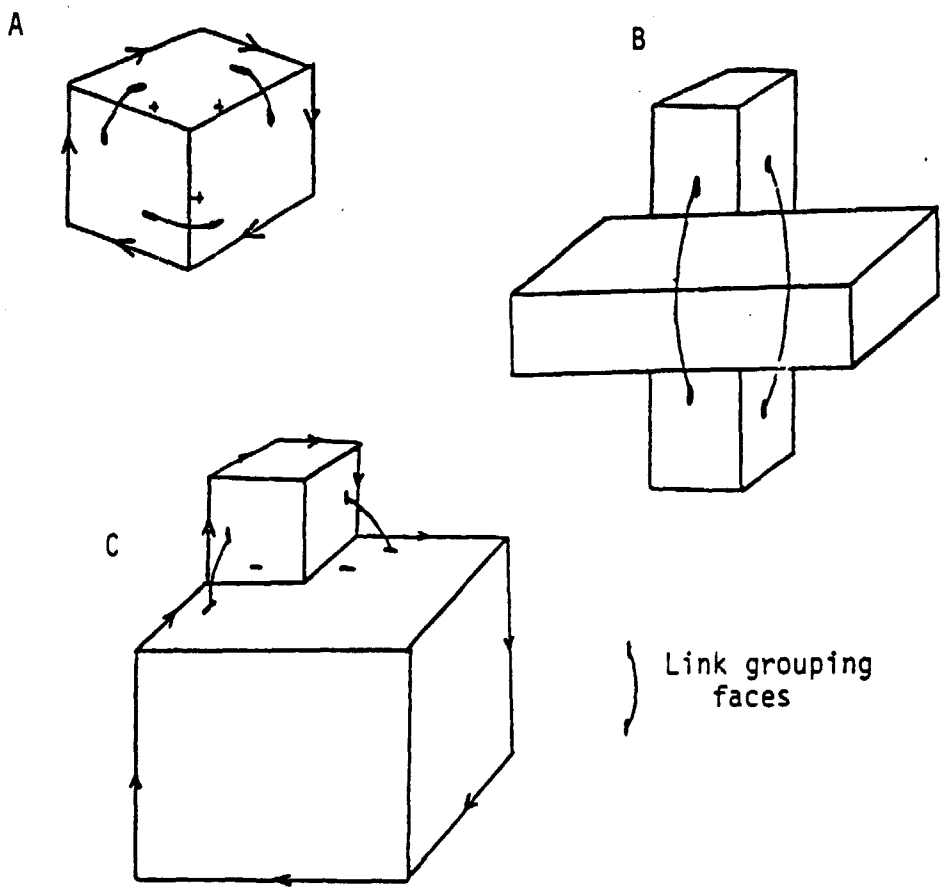


Figure 6

Heuristic Rules

A. Faces linked because they intersect at a convex edge

B. Faces linked due to coplanarity

C. Faces lined because they intersect at a concave edge where one of the edges vertices fall on an outside boundary

6A shows faces which are grouped because of the convex edge between them. Figure 6B shows faces which are grouped because they contain colinear edges with adjacent coplanar faces. Figure 6C shows an example of grouping according to Rule 3.

Rule 3 is included in the set of rules because a laser rangefinder, which measures only depth, will not see an edge between objects which are aligned. In Figure 7, Part C will be grouped into the main body when viewed from the right because it contains part of face 1, and face 1 has convex edges which link it to the rest of the main body. Face 1 contains vertices A, B, C, D, E, and F. If this object is viewed from the left, we want the same grouping to occur so that the description of the scene will be consistent. When viewed from the left Part C can be grouped into the main body by Rule 3. Part A exhibits the same characteristic. Part B is not grouped with the main body since the concave edge is colinear with the outside edge of the main body.

Using the real world as an example, objects on a table are considered separate objects unless they are aligned with an edge of the table.

b. Convexity of Edges

One of the most important features of the object reconstruction scheme is determining whether or not an edge is convex or concave.

To determine if an edge is convex or concave, we define the unit normal vector $\hat{L}_1$ and $\hat{L}_2$ for the edge in both planes intersecting at the edge, Figure 8. These unit normal vectors lie in the planes forming the edge, are perpendicular to the edge, and point to the inside of the plane. Then we form the unit vector pointing from the rangefinder eye to the midpoint for the edge in question and call this vector R. If the dot product of R and the sum of $\hat{L}_1$ and $\hat{L}_2$

$$R \cdot (\hat{L}_1 + \hat{L}_2)$$

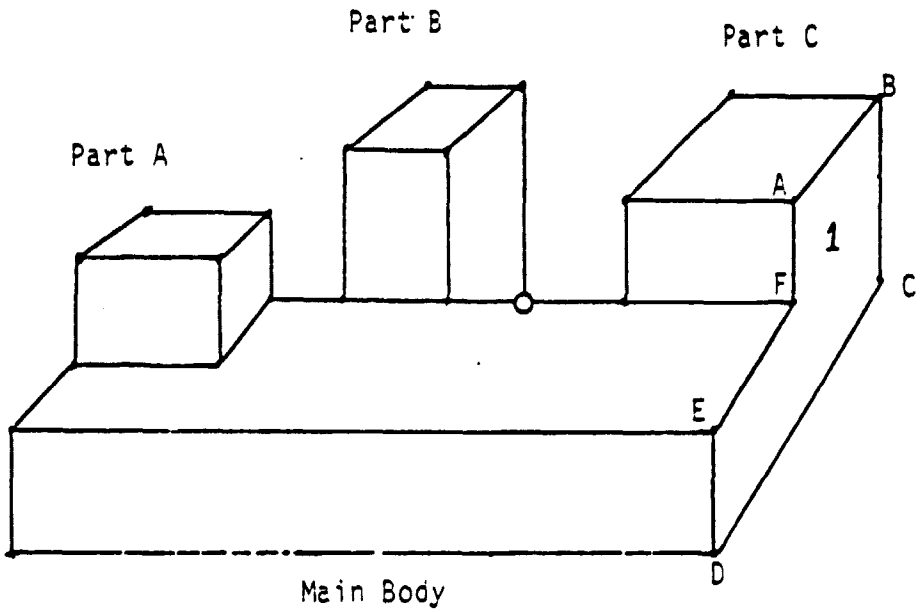


Figure 7

Rule No. 3

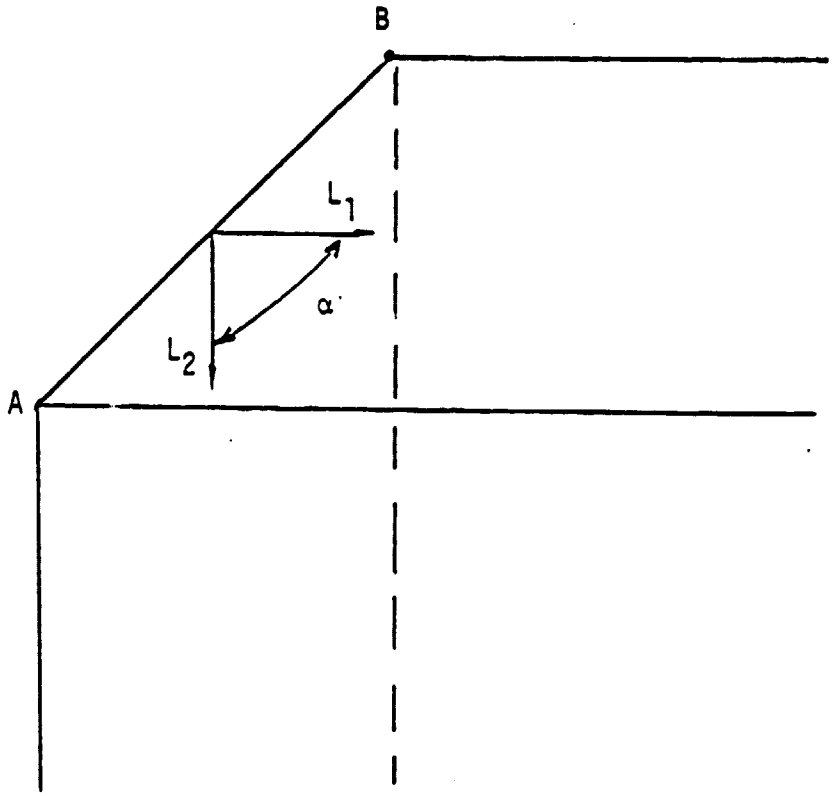


Figure 8

Determining Convexity

If α is less than 180^{o} edge AB is convex.

is greater than zero, a convex edge is indicated. If this dot product is less than zero, a concave edge is indicated. Examples of how convexity is determined are shows in Figures 9, 10, and 11, where R_1 is the projection of R into the plane formed by L_1 and L_2.

c. Colinearity and Coplanarity

All half edges are searched to see if any apirs are colinear. For two half edges to be colinear, three of the possible six unit vectors formed by pairs of the four vertices must be equal (Figure 8). We choose to use the pairs (A,B), (A,C), and (A,D) and say if:

$$2 - \text{ABS}\frac{AB \circ AC}{||AB||\ ||AC||} - \text{ABS}\frac{AB \circ AD}{||AB||\ ||AD||} < K \qquad (16)$$

where K is a small number dependent on the amount of noise, then the half edges are colinear.

Half faces which are coplanar will have equal unit normal vectors associated with the colinear half edges.

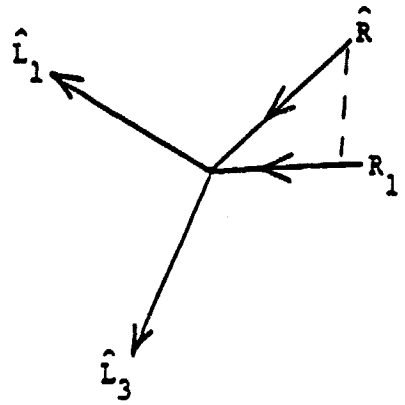
$$\text{If} \qquad ||\hat{L}_1 - \hat{L}_2|| < K_2 \qquad (17)$$

where K_2 is another constant based on measurement noise, then the half faces are coplanar.

If Rule 2 is satisfied, then the half edges are merged and the adjacent coplanar half faces are merged. In Figure 12, $\hat{L}_1 = \hat{L}_2$ indicating that the half faces are coplanar. Also AB is colinear to CE, and both B and C are virtual vertices, satisfying Rule 2. Hence, F2 will be merged with F1 and AB will be combined with CD to form AD.

d. Results

Computer simulation was performed on an IBM 3033 computer in a program written in LISP. A sample workscene is pictured in Figure 13, where 



$\hat{R} \bullet \hat{L}_1 > 0 \quad \hat{R} \bullet \hat{L}_2 > 0$

$R \bullet (\hat{L}_1 + \hat{L}_2) > 0$

edge AB is convex

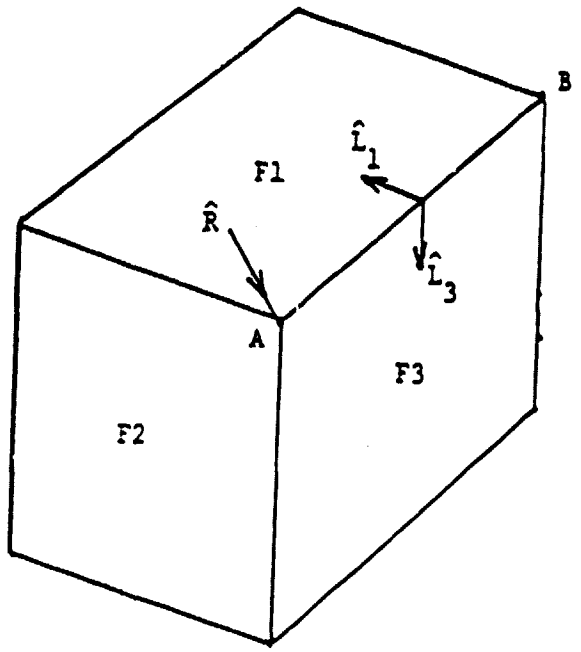
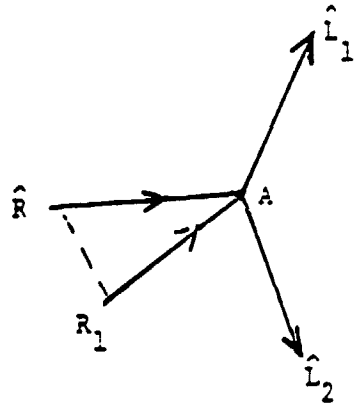


Figure 9

Acute Convexity



$\hat{R} \bullet \hat{L}_1 > 0 \quad \hat{R} \bullet \hat{L}_2 > 0$

$|\hat{R} \bullet \hat{L}_1| > |\hat{R} \bullet \hat{L}_2|$

$\hat{R} \bullet (\hat{L}_1 + \hat{L}_2) > 0$

edge AB is convex

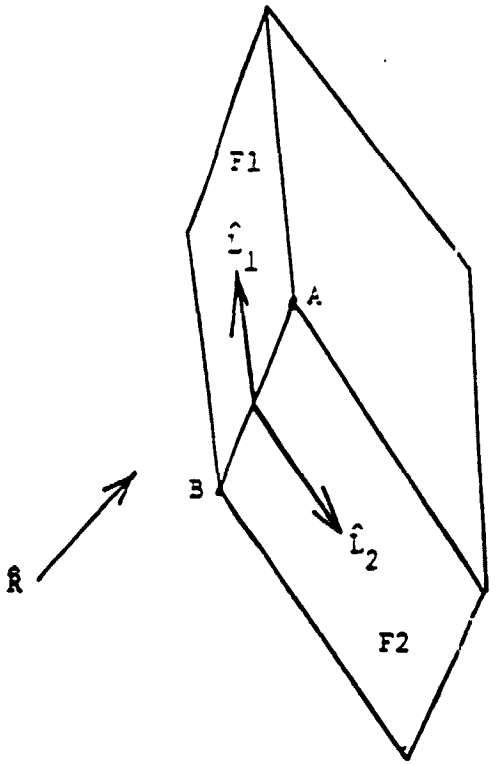
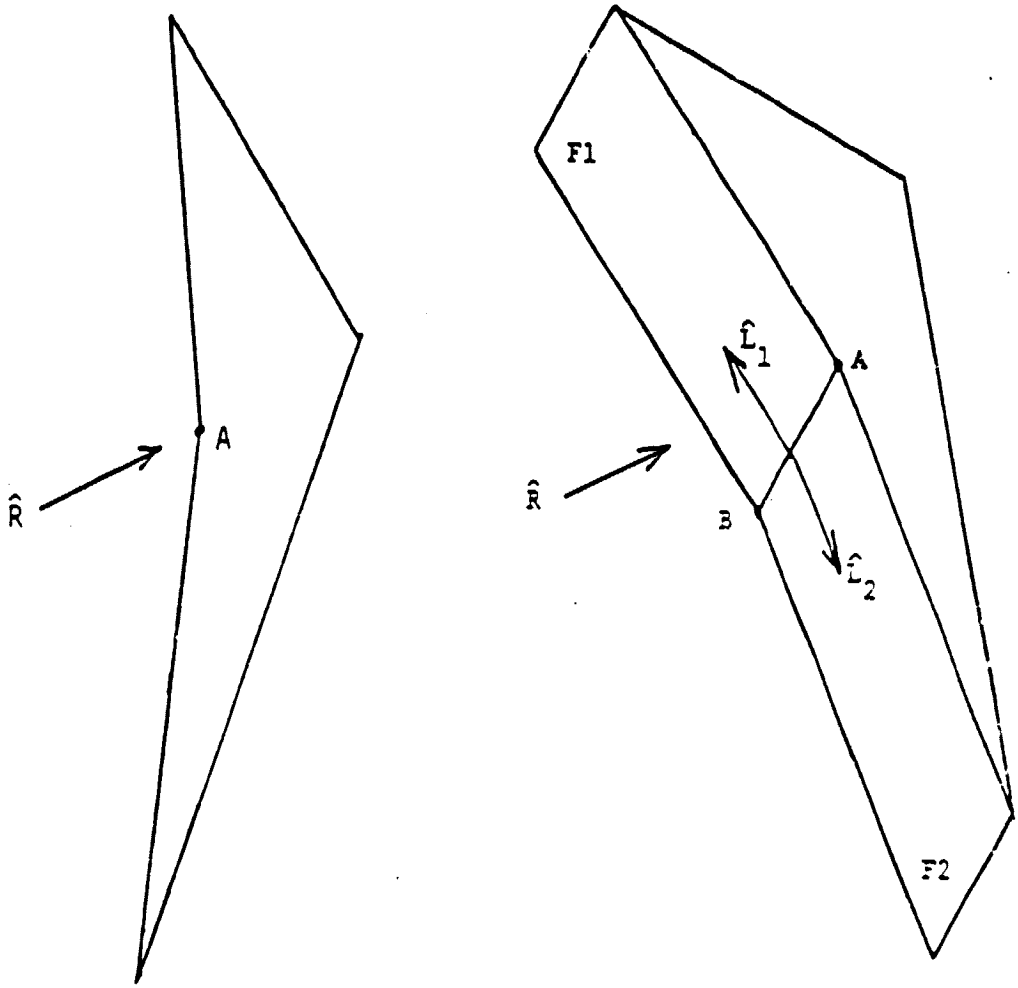
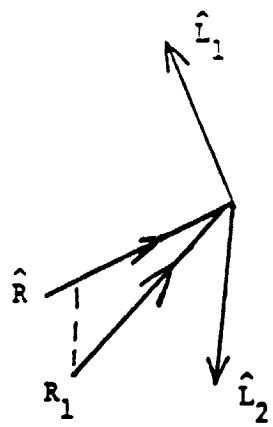


Figure 10 Obtuse Convexity





$$\hat{R} \bullet \hat{L}_1 > 0 \quad \hat{R} \bullet \hat{L}_2 < 0$$

$$|\hat{R} \bullet \hat{L}_1| < |\hat{R} \bullet \hat{L}_2|$$

$$R \bullet (\hat{L}_1 + \hat{L}_2) < 0$$

edge AB is concave

Figure 11

Obtuse concavity

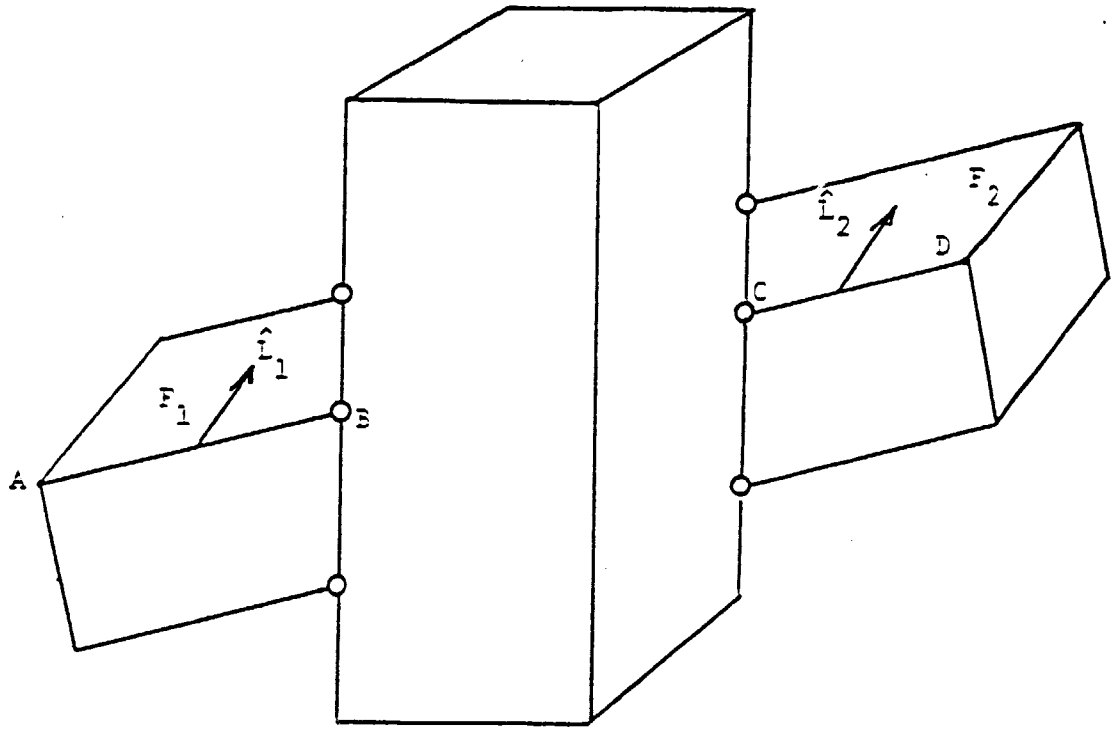


Figure 12

Determining Colinearity

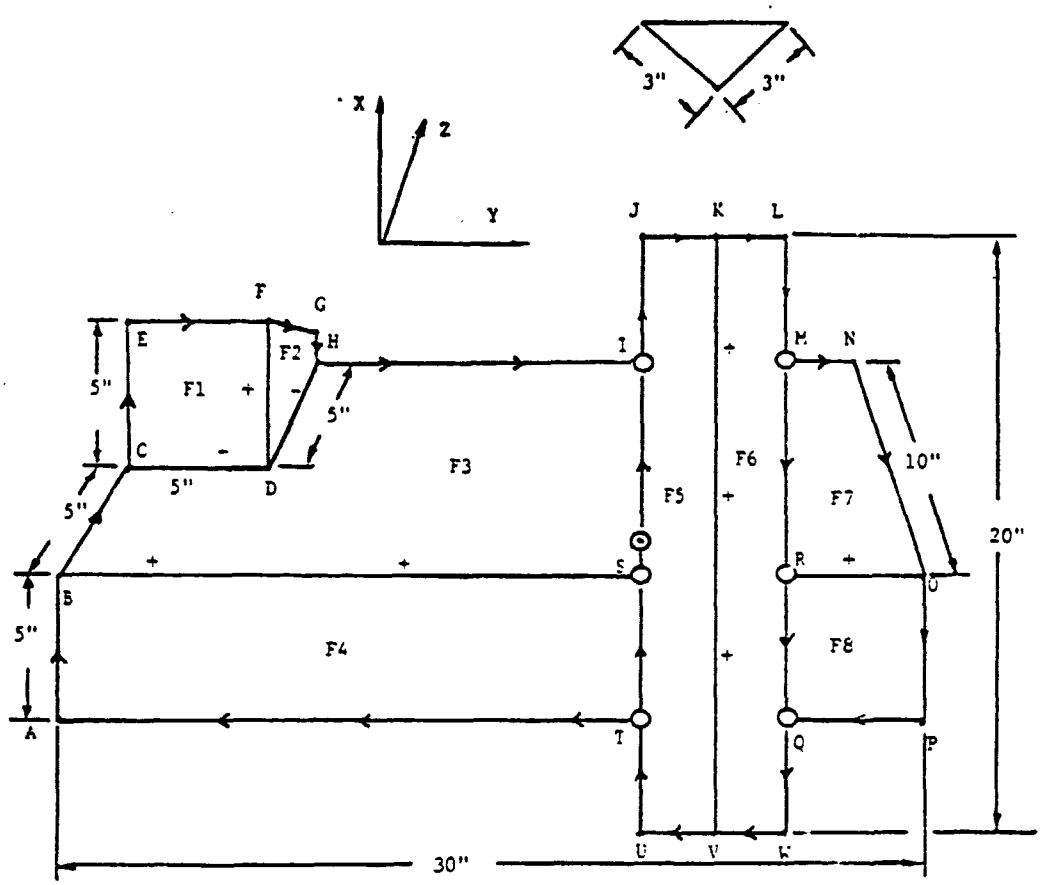


Figure 13

Sample Workscene

○ virtual vertex

⊙ Rangefinder eye position
1 foot in front of workscene

signifies the position of the laser rangefinder eye, which is the origin of the cartesian coordinate system in which the input data is recorded.

The input is in the form of lists of features. A face is defined as the list of edges which bound it; an edge is defined as a pair of vertices and contains the information on the unit vectors, and a vertex is defined by this list of its coordinates. For example, face F1 is defined as (EF DF CD CE) in the sample workscene. Also, edge DF is defined as ((F1 0 -1 0) (F2 0 0 1)) where (1 -1 0) is the direction of the unit normal vector associated with F1 at edge DF. Vertex A is defined as (-6 -20 12), its coordinates relative to the rangefinder eye location. Which vertices are virtual is determined using the fact that virtual vertices are the endpoints of only one edge while other vertices are formed by the intersection of more than one edge.

The output contains the description of the scene upon completion of the object reconstruction scheme.

The output was:

VERTUAL VERTICES:	(L S T M R Q)
NEWEDGES:	(HN BO AP)
CONCAVE:	(CD DH)
CONVEX:	(DE KV BO)
OBJECT1:	(F1 F2 F3 F4)
OBJECT2:	(F5 F6)

The scheme actually worked as follows:

- Virtual vertices were found
- Colinear half edges were fixed creating NEWEDGES HN BO AP and merging faces F7 and F8 into faces F3 and F4 respectively.
- F1 and F2 were linked across convex edge FD
- F3 and F4 were linked across convex edge BO
- F5 and F6 were linked across convex edge KV
- F1 and F3 were both linked with F3 based on Rule 3 at edges CD and DH

No other links were possible so that the groups of faces were then considered objects. Note that F7 and F8 do not appear in any object list since they have been merged away.

e. Conclusions

A heuristic scheme for three-dimensional object reconstruction has been presented above which used heuristic rules based solely on three-dimensional geometric considerations. We found that given three-dimensional input describing a work scene, the objects contained in the scene can be reconstructed using these heuristic rules. In cases where enough information is available, parts of the scene which are masked by occlusions can also be rebuilt. The reconstruction is consistent and allows for error due to measurement noise.

TASK B. POLYNOMIAL SPLINE APPRAOCH TO TWO-DIMENSIONAL SIGNAL PROCESSING

A state space appraoch to 2-D data processing using spline function is reported in the following pages. The state space notation of splines not only keeps the notation simple, compact, and uniform, but also it provides the crucial point in the recursive data processing. The fundamental principles and limitations on recursive smoothing splines are explored, which are neglected in the previous research. One special case which results in forward on-line recursive algorithm is presented.

1. The State Space Approach to 2-D Vector Processing Using Splines

The 2-D spline smoothing problem is defined as follows. A set of noise corrupted 2-D data $Z = \{z_{ij};\ i = 1,2,---M,\ j = 1,2,---,N\}$ is given, we wish to estimate the original function values and its derivatives from these discrete noisy measurements.

The measurement process is described by

$$z_{ij} = f(\xi_i, \eta_j) + V_{ij}, \quad \begin{matrix} i = 1,---,M \\ j = 1,---,N \end{matrix} \tag{18}$$

where $f(\xi_i, \eta_j)$ is the sample value of original function which is assumed of (mi ed partial) derivatives up to certain order. V_{ij} is random noise with zero mean and finite variance R_{ij}.

The approximating function adopted here is the 2-D spline function which is defined in wide sense, a piecewise function of continuous derivatives up to certain order and each piece is defined only in one grid, for example, the regular square grid as shown in Figure 14.

The reason for using spline function is that the approximating error analysis has been well developed in the numerical mathematics [9, 10]. Another advantage of this approach is that under certain conditions the recursive computation can be realized, which makes the real time operation possible.

The ent.re spline function in domain R as dictated in Figure 14 would be

$$S(\xi,\eta) = \begin{cases} S_{00}(\xi,\eta) & \xi_0 \le \xi \le \xi_1 \ , \quad \eta_0 \le \eta \le \eta_1 \\ S_{10}(\xi,\eta) & \xi_1 \le \xi \le \xi_2 \ , \quad \eta_0 \le \eta \le \eta_1 \\ S_{ij}(\xi,\eta) & \xi_i \le \xi \le \xi_{i+1} \ , \quad \eta_j \le \eta \le \eta_{j+1} \\ S_{M-1,N-1}(\xi,\eta) & \xi_{M-1} \le \xi \le \xi_M \ , \quad \eta_{N-1} \le \eta \le \eta_N \end{cases} \tag{19}$$

The objective function which we wish to minimize is

$$J = \sum_{i=1}^{M} \sum_{j=1}^{N} \rho_{ij} [z_{ij} - S(\xi_i,\eta_j)]^T R_{ij}^{-1} [z_{ij} - S(\xi_i,\eta_j)] + \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (1-\rho_{ij}) \int_{\xi_i}^{\xi_{i+1}} \int_{\eta_j}^{\eta_{j+1}} C(s) d\xi d\eta \tag{20}$$

where ρ_{ij}'s are weighting parameters and

$$0 \le \rho_{ij} \le 1,$$

$C(s)$ is the smoothness measure function usually defined as function of the high order derivatives of $S(\xi,\eta)$.

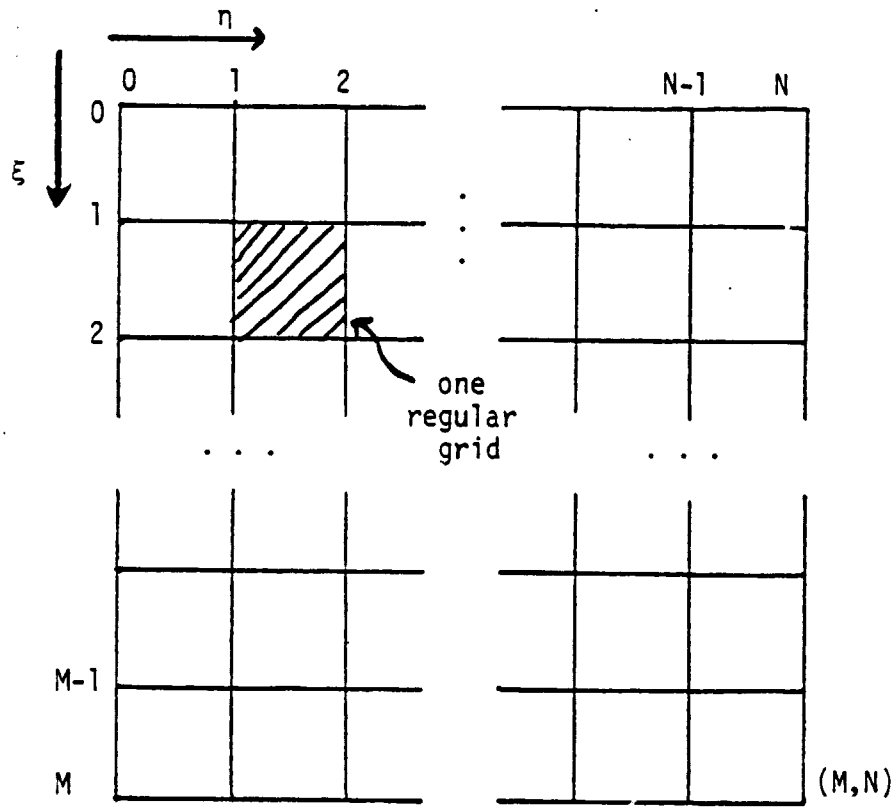


Figure 14 The region of 2-D splines, R

The first term in Equation (3) represents the noise effect at the grid points and the second term is a measure of smoothness of the approximating function. The selection of $C(s)$ and $S(\xi,\eta)$ could be independent, but the optimum $S(\xi,\eta)$ which minimizes a special $\int\int C(s)d\mu d\lambda$ is often unique.

Notice that $S(\xi,\eta)$ would act as an interpolation function when ρ_{ij} approaches to zero, and on the other extreme, when ρ_{ij} approaches to one $S(\xi,\eta)$ would become a plane. Thus, the value of weighting parameter ρ_{ij} controls the trade-off between the measurement error and the smoothness of the approximation function.

a. The State Space Model of Bicubic Spline

The most often used splines are polynominal splines because of its simplicity in computation and the closed form solution. In general, a 2-D spline defined in one grid is described by a partial differential equation. However, only few of the solutions satisfying some restrictions could be represented by linear discrete state space model defined on the grid points. The bicubic polynominal spline is the one among them of particular importance.

The bicubic polynominal spline is the polynominal which minimizes the smoothness measure $C(s) = \left[\frac{\partial^4 s}{\partial\xi^2\partial\eta^2}\right]^2$. For convenience, new notations of variables are introduced

$$\mu \triangleq \xi - \xi_i \quad , \qquad \xi_i \leq \xi \leq \xi_{i+1}$$

$$\lambda \triangleq \eta - \eta_j \quad , \qquad \eta_j \leq \eta \leq \eta_{j+1}$$

and thus,

$$0 \leq \mu \leq \Delta\xi \quad , \qquad \Delta\xi \triangleq \xi_{i+1} - \xi_i$$

$$0 \leq \lambda \leq \Delta\eta \quad , \qquad \Delta\eta \triangleq \eta_{j+1} - \eta_j$$

From the principle of calculus of variation, the optimum solution of

$$J' = \int_0^{\Delta\xi}\int_0^{\Delta\eta} \left[\frac{\partial^4 S_{ij}(\mu,\lambda)}{\partial\mu^2\partial\lambda^2}\right]^2 d\mu d\lambda \qquad (21)$$

is the one satisfies the Euler equation,

$$\frac{\partial^8 S_{ij}(\mu,\lambda)}{\partial\mu^4\partial\lambda^4} = 0 \tag{22}$$

with proper boundary conditions [15].

If only polynominal solutions are of concern, then the solution satisfies Equation (22) is equivalent to the one which satisfies the following equation,

$$\frac{\partial^4 S_{ij}(\mu,\lambda)}{\partial\mu^2\partial\lambda^2} = [\mu\lambda \;\; \lambda \;\; \mu \;\; 1] \begin{bmatrix} U_{ij}^3 \\ U_{ij}^2 \\ U_{ij}^1 \\ U_{ij}^0 \end{bmatrix} \tag{23}$$

where $U_{ij} \triangleq [U_{ij}^3 \;\; U_{ij}^2 \;\; U_{ij}^1 \;\; U_{ij}^0]^T$ are under-determined parameters.

The general solution of Equation (23) is

$$\begin{aligned} S_{ij}(\mu,\lambda) &= \phi_1(\lambda) + \mu\phi_2(\lambda) + \psi_1(\mu) + \lambda\psi_2(\mu) \\ &+ \begin{bmatrix} \frac{\mu^3\lambda^3}{36} & \frac{\mu^2\lambda^3}{12} & \frac{\mu^3\lambda^2}{12} & \frac{\mu^2\lambda^2}{4} \end{bmatrix} U_{ij} \end{aligned} \tag{24}$$

where ϕ_1, ϕ_2, ψ_1, and ψ_2 are aribitrary polynominal functions. If we define the "state" as the column vector $[S, \frac{\partial s}{\partial\mu}, \frac{\partial s}{\partial\lambda}, \frac{\partial^2 s}{\partial\mu\partial\lambda}]^T$, then the exact linear discrete model of Equation (24) defined on the end points of one grid can be obtained.

$$X_{i+1,\,j+1} = F^{01}X_{i,\,j+1} + F^{01}X_{i+1,\,j} + F^{00}X_{i,j} + \Gamma U_{ij} \tag{25}$$

where

$$X_{i+\ell,j+m} \triangleq [S(\mu,\lambda) \;\; \frac{\partial s}{\partial\mu}(\mu,\lambda) \;\; \frac{\partial s}{\partial\lambda}(\mu,\lambda) \;\; \frac{\partial^2 s}{\partial\mu\partial\lambda}(\mu,\lambda)]^T \Big|_{\lambda=m\Delta\eta}^{\mu=\ell\Delta\xi} \tag{25a}$$

$$\ell = 0, 1;\; m = 0,1$$

$$F^{01} = \begin{bmatrix} 1 & \Delta_\xi & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta_\xi \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad F^{10} = \begin{bmatrix} 1 & 0 & \Delta_\eta & 0 \\ 0 & 1 & 0 & \Delta_\eta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{25b}$$

$$F^{00} = - \begin{bmatrix} 1 & \Delta_\xi & \Delta_\eta & \Delta_\xi\Delta_\eta \\ 0 & 1 & 0 & \Delta_\eta \\ 0 & 0 & 1 & \Delta_\xi \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{25c}$$

and

$$\Gamma = \begin{bmatrix} \frac{\Delta_\xi^3\Delta_\eta^3}{36} & \frac{\Delta_\xi^2\Delta_\eta^3}{12} & \frac{\Delta_\xi^3\Delta_\eta^2}{12} & \frac{\Delta_\xi^2\Delta_\eta^2}{4} \\ \frac{\Delta_\xi^2\Delta_\eta^3}{12} & \frac{\Delta_\xi\Delta_\eta^3}{6} & \frac{\Delta_\xi^2\Delta_\eta^2}{4} & \frac{\Delta_\xi\Delta_\eta^2}{2} \\ \frac{\Delta_\xi^3\Delta_\eta^2}{12} & \frac{\Delta_\xi^2\Delta_\eta^2}{4} & \frac{\Delta_\xi^3\Delta_\eta}{6} & \frac{\Delta_\xi^2\Delta_\eta}{2} \\ \frac{\Delta_\xi^2\Delta_\eta^2}{4} & \frac{\Delta_\xi\Delta_\eta^2}{2} & \frac{\Delta_\xi^2\Sigma_\eta}{2} & \Delta_\xi\Delta_\eta \end{bmatrix} \tag{25d}$$

The linear discrete state model for a general 2-D spline is often not available. Because of the restriction of the positions fo grid points, it only allows the function $S(\mu,\lambda)$ propogates the variation along the lines parallel to the ξ or η axis. Consequently, the arbitrary functions could only be functions of only one variable, μ or λ, so that they can be cancelled. Linearization or approximation may be used to derive the linear discrete model for those do not meet above requirements.

Since the finite state model is one of the essential condition for recursive algorithm, the bicubic spline is the one, if not the only one, of the members which can be implemented recursively.

b. Vector Processing Formulation

1-D estimation theory is well developed, nevertheless, the corresponding theorems in 2-D is not always possible. This complication occurs partially due to the choice of the support defined for the 2-D filters [16-18]. An estimator with support defined by Equation (25) usually does not have the optimum solution either in the sense of least square or quadratic performance index.

To avoid this fundamental restriction, a vector processing model is constructed as follows. Define the glcbal state quantity to be the vector along the column of region R,

$$\underline{X}_j = \begin{bmatrix} X_{0j} \\ X_{1j} \\ X_{2j} \\ \cdot \\ \cdot \\ \cdot \\ X_{Mj} \end{bmatrix} \qquad j=0,1,2....N \tag{26}$$

The state equations along the column strip are

$$\begin{cases} X_{0,j+1} = F^{10}X_{0,j} + \Gamma U_{-1,j} \\ X_{1,j+1} = F^{01}X_{0,j+1} + F^{10}X_{1,j} + F^{00}X_{0,j} + \Gamma U_{0,j} \\ \cdot \\ \cdot \\ \cdot \\ X_{M,j+1} = F^{01}X_{M-1,j+1} + F^{10}X_{M,j} + F^{00}X_{M-1,j} + \Gamma U_{M-1,j} \end{cases} \tag{27}$$

or

$$\underline{X}_{j+1} = \underline{\underline{F}}^{01}\underline{X}_{j+1} + \underline{\underline{F}}^{10}\underline{X}_j + \underline{\underline{F}}^{00}\underline{X}_j + \underline{\underline{\Gamma}}^{0}\underline{U}_j \tag{27a}$$

which could be rewritten as

$$\underline{X}_{j+1} = \underline{F}\,\underline{X}_j + \underline{\Gamma}\,U_j \tag{28}$$

where

$$\underline{F} = [I - \underline{F}^{01}]^{-1}\ [\underline{F}^{10} + \underline{F}^{00}] \quad ,$$

$$\underline{\Gamma} = [I - \underline{F}^{01}]^{-1}\ \underline{\Gamma}^{0} \quad ,$$

and

$$F^{01} = \begin{bmatrix} 0 & 0 & & \\ F^{01} & 0 & & \\ 0 & F^{01} & & \\ & \ddots & \ddots & 0 \\ & & F^{01} & 0 \end{bmatrix}, \qquad F = \begin{bmatrix} F^{10} & 0 & & \\ 0 & F^{10} & & \\ & \ddots & \ddots & 0 \\ & & 0 & F^{10} \end{bmatrix},$$

$$F^{00} = \begin{bmatrix} 0 & 0 & & \\ F^{00} & 0 & & \\ 0 & F^{00} & & 0 \\ & \ddots & \ddots & 0 \\ & & F^{00} & 0 \end{bmatrix}, \qquad \Gamma^{0} = \begin{bmatrix} \Gamma & 0 & & \\ 0 & \Gamma & & \\ & \ddots & & 0 \\ & & 0 & \Gamma \end{bmatrix}.$$

To simplify the calculation and deduce the recursive solution, only the quadratic smoothness measure is considered, that is from Equation (6) assume

$$\int_0^{\Delta_\xi}\int_0^{\Delta_\eta} C(s)\,d\mu d\lambda = \int_0^{\Delta_\xi}\int_0^{\Delta_\eta} \left[\frac{\partial^4 s_{ij}}{\partial\mu^2\partial\lambda^2}\right]^2 d\mu d\lambda \tag{29}$$

$$= U_{ij}^{T}\, Q_{ij}^{-1}\, U_{ij} \tag{29a}$$

where $$Q_{ij}^{-1} \triangleq \int_0^{\Delta_\xi}\int_0^{\Delta_\eta} [\mu\lambda\ \ \lambda\ \ \mu\ \ 1]^{T}\ \ [\mu\lambda\ \ \lambda\ \ \mu\ \ 1]\,d\mu d\lambda \tag{29b}$$

for bicubic splines. However, the result of above problem can be easily extended to the general quadratic form,

$$\int_0^{\Delta_\xi}\int_0^{\Delta_\eta} C(s)\,d\mu d\lambda = [U_{ij}\ X_{ij}\ X_{i+1,j}\ X_{i,j+1}]\ \underline{C}^{-1} \begin{bmatrix} U_{ij} \\ X_{ij} \\ X_{i+1,j} \\ Xi,j+1 \end{bmatrix} \tag{30}$$

Let $\underline{z}_j \triangleq [o\ z_{1j}\ z_{2j}\ \ldots\ldots z_{Nj}]^T$

then the objective function (Equation 20) can be expressed in the global state form,

$$J = (\hat{X}_o - \underline{X}_o)\ \underline{P}_o^{-1}\ (\hat{\underline{X}}_o - \underline{X}_o) + \sum_{j=1}^{N} [z_j - \underline{\underline{H}}\underline{X}_j]^T\ \underline{\underline{R}}_j^{-1}\ [z_j - \underline{\underline{H}}\underline{X}_j]$$

$$+ \sum_{j=0}^{N-1} \underline{U}_j^T\ \underline{\underline{Q}}_j^{-1}\ \underline{U}_j \qquad (31)$$

where $\hat{\underline{X}}_o$ is the initial guess.

$$\underline{\underline{H}} = \begin{bmatrix} H & 0 & \bigcirc \\ 0 & H & 0 \\ \bigcirc & 0 & H \end{bmatrix} \qquad (31a)$$

$$\underline{\underline{R}}_j^{-1} = \begin{bmatrix} \rho_{0j} R_{0j}^{-1} & 0 & \bigcirc \\ 0 & \rho_{ij} R_{ij}^{-1} & 0 \\ 0 & \bigcirc & 0 \quad \rho_{Mj} R_{Mj}^{-1} \end{bmatrix} \qquad (31b)$$

$$Q_j^{-1} = \begin{bmatrix} (1-\rho_{0j})Q_{0j}^{-1} & 0 & \bigcirc \\ 0 & (1-\rho_{1j})Q_{1j}^{-1} & 0 \\ 0 & \bigcirc & 0 \quad (1-\rho_{Mj})Q_{Mj}^{-1} \end{bmatrix} \qquad (31c)$$

The first term in Equation (14) is added to reduce the effect of initial guess.

c. Smoothing Algorithms

We now turn to the development of the estimation algorithms. The closed form recursive formula for the problem described in the previous sections does not always exist. There are some significant special cases that provide simple recursive computations as discussed below.

For the sake of forward real time recursive computation, the unknow coefficients $\underline{U}_j$ can only be the function of previous states $\underline{X}_i$'s and parameters $\underline{U}_i$'s, i<j. Therefore, the minimization procedure of Equation (31) can be expressed as

$$\underset{\underline{X}_0,\ \underline{U}_0 \cdots \underline{U}_{j-1}}{\text{Min}} \quad J_0^j = \underset{\underline{X}_0}{\text{Min}} \left[J_0 + \underset{\underline{U}_0}{\text{Min}} \left[J_1 + \ldots . \quad \underset{\underline{U}_{j-1}}{\text{Min}} \left[J_j . \ \ldots . \right] \right] \right] | Z_1^j \tag{32}$$

where $$Z_1^j \triangleq \{z_\ell : 1 \leq \ell \leq j\} \tag{32a}$$

$$J_\ell \triangleq (z_\ell - \underline{H}\, \underline{X}_\ell)^T\, \underline{R}_\ell^{-1}\, (z_\ell - \underline{H}\, \underline{X}_\ell) + \underline{U}_{\ell-1}^T\, \underline{Q}_{\ell-1}^{-1}\, \underline{U}_{\ell-1} \qquad \ell = 1,2,\ldots . j \tag{32b}$$

$$J_0 \triangleq (\hat{\underline{X}}_0 - \hat{\underline{X}}_0)^T\, P^{-1}\, (\hat{\underline{X}}_0 - \hat{X}_0) \tag{32c}$$

and $$J_i^j = \sum_{\ell=i}^{j} J_\ell$$

The notation $|Z_1^j$ emphasizes that the measurements are up to j^{th} stage.

Let us denote the optimum estimate of $\underline{X}_i$ which minimize J_0^j is $\hat{\underline{X}}_{i|j}$, that is, $\hat{\underline{X}}_{i|j}$ is the optimum estiamte of $\underline{X}_i$ using the data $z_1, z_2, \ldots . , z_j$. The relation between $\hat{\underline{X}}_{i|j}$ and $\hat{\underline{X}}_{i+1|j}$ comes from Equation (25)

$$\hat{\underline{X}}_{i+1|j} = F\, \hat{\underline{X}}_{i|j} + \Gamma \underline{U}_{i|j} \tag{33}$$

where the index i/j indicates the quantities are the optimum solution using Z_1^j.

In order to obtain $\hat{X}_{j|j}$, we must compute all $U_{i|j}$'s and $X_{o|j}$ via Equation (32). The whole procedure of computation is dictated in Figure 15. First, we compute $U_{i|j}$'s from j^{th} stage backward to o^{th} stage, and then the smoothed estimate $S_{i|j}$'s are calculated forward by using Equation (33).

At the j^{th} state, $U_{j-1|j}$ is obtained by minimizing J_j with respect to U_{j-1}.

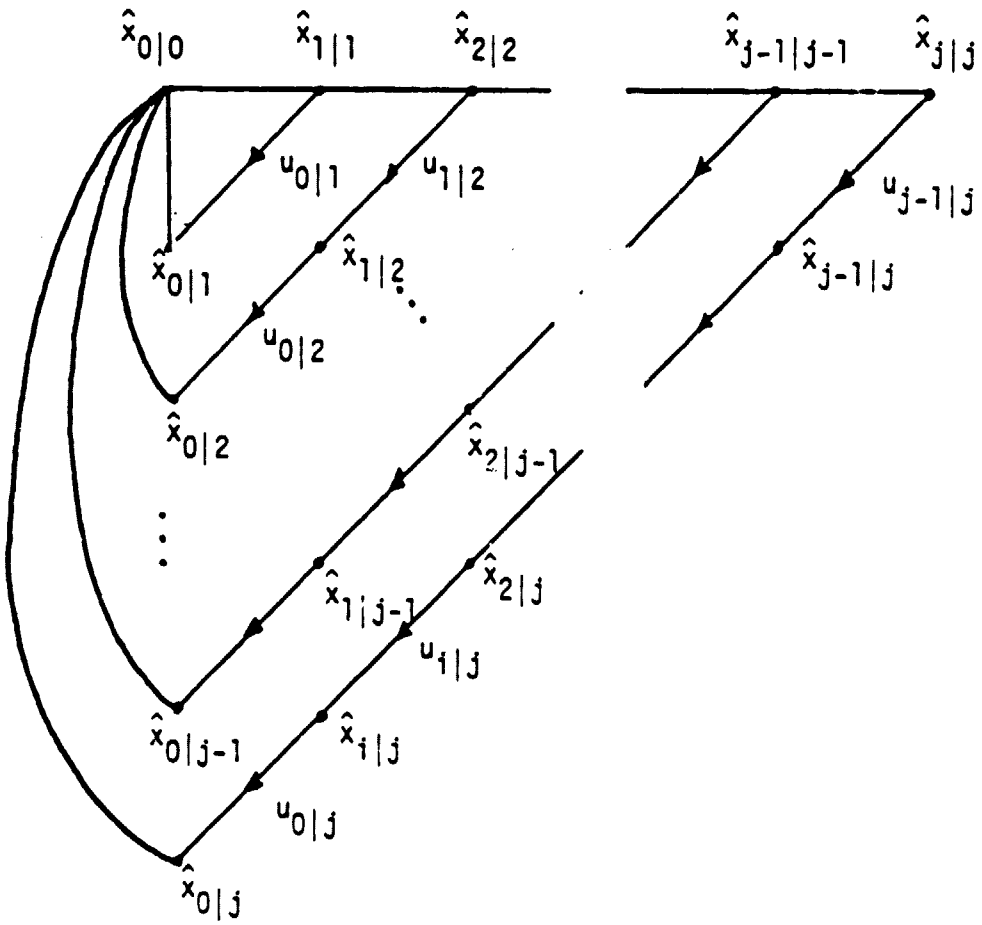


Figure 15 The Computation Scheme

$$J_j = (\underline{z}_j - \underline{\underline{H}}\,\underline{X}_j)^T\,\underline{\underline{R}}_j^{-1}\,(\underline{z}_j - \underline{\underline{H}}\,\underline{X}_j) + \underline{U}_{j-1}^T\,\underline{\underline{Q}}_{j-1}^{-1}\,\underline{U}_{j-1}$$

$$= (\underline{z}_j - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{X}_{j-1} - \underline{\underline{H}}\,\underline{\underline{\Gamma}}\,\underline{U}_{j-1})^T\,\underline{\underline{R}}_j^{-1}\,(\underline{z}_j - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{X}_{j-1} - \underline{\underline{H}}\,\underline{\underline{\Gamma}}\,\underline{U}_{j-1})$$

$$+ \underline{U}_{i-1}^T\,\underline{\underline{Q}}_{j-1}^{-1}\,\underline{U}_{j-1}, \qquad (34)$$

since $X_i = F\,X_{j-1} + \quad U_{j-1}$. Then taking the gradient of Equation (17), we obtain the expression,

$$-\underline{\underline{\Gamma}}^T\underline{\underline{H}}\,\underline{\underline{R}}_j^{-1}(\underline{z}_j - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{X}_{j-1} - \underline{\underline{H}}\,\underline{\underline{\Gamma}}\,\underline{U}_{j-1}) + \underline{\underline{Q}}_{j-1}^{-1}\,\underline{U}_{j-1} = 0 \qquad (35)$$

or $$\underline{U}_{j-1} = \underline{\underline{T}}_j^{-1}\underline{\underline{\Gamma}}\,\underline{\underline{H}}^T\,\underline{\underline{R}}_j^{-1}\,(\underline{z}_j - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{X}_{j-1}) \qquad (36)$$

where $\underline{\underline{T}}_j \triangleq \underline{\underline{Q}}_{j-1}^{-1} + \underline{\underline{\Gamma}}^T\underline{\underline{H}}^T\underline{\underline{R}}_j^{-1}\underline{\underline{H}}\,\underline{\underline{\Gamma}}$ (36a)

Because the preceeding solution is solved by using data Z_1^j , $\underline{U}_{j-1}$, and $\underline{X}_{j-1}$ should be denoted as $\underline{U}_{j-1|j}$ and $\underline{X}_{j-1|j}$ respectively.

To compute the $\underline{U}_{j-2}$ Equation (36) is substituted into Equation (34), and J_j is simplified as function of $\underline{X}_{j-1}$ only.

$$J_j = (\underline{z}_j - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{X}_{j-1})^T\,\underline{\underline{W}}_j^{-1}\,(\underline{z}_j - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{X}_{j-1}) \qquad (37)$$

where $$W_j = R_j + \underline{\underline{H}}\,\underline{\underline{\Gamma}}\,\underline{\underline{Q}}_{j-1}\,\underline{\underline{\Gamma}}^T\,\underline{\underline{H}}^T \qquad (37a)$$

At $j\text{-}1^{th}$ stage, the problem to be solved is:

$$\min_{\underline{U}_{j-2}} J_{j-1}^j = \min_{\underline{U}_{j-2}} (J_{j-1} + J_j)$$

$$= \min_{\underline{U}_{j-2}} [z_{j-1} - \underline{\underline{H}}\,\underline{X}_{j-1})^T\,\underline{\underline{R}}_{j-1}^{-1}\,(z_{j-1} - \underline{\underline{H}}\,\underline{X}_{j-1})$$

$$+ \underline{U}_{j-2}^T\,\underline{\underline{Q}}_{j-2}^{-1}\,\underline{U}_{j-2} + (\underline{z}_j - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{X}_{j-1})^T\,\underline{\underline{W}}_j^{-1}\,(\underline{z}_j - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{X}_{j-1})] \qquad (38)$$

The resultant $U_{j-2|j}$ is

$$\underline{U}_{j-2|j} = \underline{\underline{T}}_{j-1}^{-1}\,\{\underline{\underline{\Gamma}}^T\,\underline{\underline{H}}^T\,\underline{\underline{R}}_{j-1}^{-1}\,(\underline{z}_{j-1} - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{X}_{j-2}) + \underline{\underline{\Gamma}}^T\,\underline{\underline{F}}^T\,\underline{\underline{H}}^T\,\underline{\underline{W}}_j^{-1}(z_j - \underline{\underline{H}}\,\underline{\underline{F}}\,\underline{\underline{F}}X_{j-2})\} \qquad (39)$$

where $\underline{\underline{T}}_{j-1} = \{\underline{\underline{\Gamma}}^T \underline{\underline{H}}^T \underline{\underline{R}}_{j-2}^{-1} \underline{\underline{H}}\, \underline{\underline{\Gamma}} + \underline{\underline{Q}}_{j-2}^{-1} + \underline{\underline{\Gamma}}^T \underline{\underline{F}}^T \underline{\underline{H}}^T \underline{\underline{W}}_j^{-1} \underline{\underline{H}}\, \underline{\underline{F}}\, \underline{\underline{\Gamma}}\}$ (39a)

Observing Equation (36) and (39), it seems that there is no simple closed recursive formula to compute $\underline{U}_{i|j}$'s.

However, if the state Equation (33) satisfies certain conditions, the closed recursive formulas for $\underline{U}_{i|j}$'s or $\hat{\underline{X}}_{i|j}$'s are possible. Substituting Equation (22) to Equation (21), J_{j-1}^j becomes function of $\hat{\underline{X}}_{j-2}$ only. Repeat the above procedure backward down to 0^{th} stage, we will have J_0^j which is a function $\hat{\underline{X}}_0$ only. Then $\underline{X}_{0|j}$ can be obtained by minimizing J_0^j with respect to $\hat{\underline{X}}_0$. After that, $\hat{X}_{1|j}$, $\hat{X}_{2|j}$, ... are then computed forward via Equation (33). This nonrecursive computing procedure can be applied to the general state model on which no restrictions are posed.

There are two special cases which save the computation and/or memory storage significantly. The first case is that the calculation of $\underline{U}_{i|j}$ could be recursive, if $\underline{\underline{H}}$ is invertable. However, in many practical applications, this may not be true.

The second case is that we can compute $X_{i|j}$ directly from $X_{j-1|j-1}$ without computing all the $U_{i|j}$'s, i= 0, 1,..., j-2, if $\underline{\underline{\Gamma}}$ is invertable. The second case is much more important not only because it is true for the splines defined in Section a , but also it results in a true forward recursive on-line algorithm.

Assume $\underline{\underline{\Gamma}}$ is invertable, $\underline{U}_i = \underline{\underline{\Gamma}}^{-1} (\underline{X}_{i+1} - F\underline{X}_i)$. Because all $\underline{U}_i$'s are independent variables, the $\underline{X}_i$'s can be considered as independent variables. Therefore, J_i can be rewritten of the form

$$J_i = (\underline{z}_i - \underline{\underline{H}}\, \underline{X}_i)^T \underline{\underline{R}}_i^{-1} (\underline{z}_i - \underline{\underline{H}}\, \underline{X}_i) + (\underline{X}_i - \underline{X}_{i-1})^T \underline{\underline{\Gamma}}^{-T} \underline{\underline{Q}}_{i-1}^{-1} \underline{\underline{\Gamma}}^{-1} (\underline{X}_i - \underline{X}_{i-1}) \quad (40)$$

Thus,

$$\min_{X_0,U_0,\ldots,U_{j-1}} J_0^j = \min_{X_0,\,X_1,\ldots,X_j} J_0^j$$

$$= \min_{\underline{X}_j,\,\underline{X}_{j-1}} J_j + \min_{\underline{X}_{j-1},\,\underline{X}_{j-2}} J_{j-1} + \ldots + \min_{X_0} J_0 \tag{41}$$

Noted that every term in Equation (41) is a function of two variables only, and every indpendent variables $\underline{X}_i$'s are involved in two terms only. In comparison with Equation (32), Equation (41) has a significant decoupling structure which enables the direct forward recursive algorithm.

To illustrate this feature, let us see the following example. Assume the optimum filter estimate $\hat{X}_{j|j}$ is already available, which is the solution of

$$\min_{X_0,\,X_1,\ldots.X_j} J_0^j = \min_{X_0,X_{j-1}} J_j + \min_{X_{j-1},X_{j-2}} J_{j-1} + \ldots. + \min_{X_0} J_0 \tag{42}$$

Now, we wish to estimate $\hat{X}_{j+1\ j+1}$, that is, we want to find the X_{j+1} such that

$$\min_{X_0,\,X_1,\ldots.X_j,X_{j+1}} J_0^{j+1} = \min_{X_{j+1},X_j} J_{j+1} + \min_{X_j,X_{j-1}} J_j + \ldots. + \min_{X_0} J_0 \tag{43}$$

Compare Equations (42) and (43), it is evident that

$$\min_{X_0,X_1\ldots.,X_{j+1}} J_0^{j+1} = \min_{X_{j+1},X_j} J_{j+1} + \min_{X_0,X_1,\ldots.,X_j} J_0^j \tag{44}$$

Consequently, the present estimate $\hat{X}_{j+1|j+1}$ would equal to the previous estimate $\hat{X}_{j|j}$ plus a correction term which depends on z_{j+1} only.

After some derivation, filtered estimate is given by

$$\hat{\underline{X}}_{j+1|j+1} = \underline{\underline{F}}\,\hat{\underline{X}}_{j|j} + \underline{\underline{P}}_{j+1}^{-1}\,\underline{\underline{H}}^T\underline{\underline{R}}_{j+1}^{-1}\,(\underline{z}_{j+1} - \underline{\underline{H}}\,\underline{\underline{F}}\,\hat{\underline{X}}_{j|j}) \tag{45}$$

here $$\underline{\underline{P}}_{j+1} = (\underline{\underline{H}}^T \underline{\underline{R}}_{j+1}^{-1} \underline{\underline{H}} + \underline{\underline{M}}_j^{-1})^{-1} \quad (45a)$$

$$\underline{\underline{M}}_{j+1} = \underline{\underline{\Gamma}}\, \underline{\underline{Q}}_j\, \underline{\underline{\Gamma}}^T + \underline{\underline{F}}\, \underline{\underline{P}}_j\, \underline{\underline{F}}^T \quad (45b)$$

Similarly, the 1-lag smoothing equation can be obtained,

$$\hat{\underline{X}}_{j|j+1} = \hat{\underline{X}}_{j|j} + \underline{\underline{P}}_j\, \underline{\underline{F}}^T\, \underline{\underline{H}}^T\, \underline{\underline{R}}_{j+1}\, (\underline{z}_{j+1} - \underline{\underline{H}}\, \underline{X}_{j+1|j+1}) \quad (46)$$

which can be extended to m-lag smoothing equation,

$$\hat{\underline{X}}_{j-m|j} = \hat{\underline{X}}_{j-m\ j-m} + \underline{\underline{P}}_{j-m}\, \underline{\underline{F}}^T\, \underline{\underline{H}}^T\, \underline{\underline{R}}_{j-m+1}^{-1}\, (\underline{z}_{j-m+1} - \underline{\underline{H}}\, \underline{X}_{j-m+1|j}) \quad (47)$$

d. Conclusions

The general properties of 2-D smoothing splines problem has been discussed in this paper. The principles and restrictions on forming a recursive algorithm of 2-D vector processor using splines are explored. It is found that the recursive on-line structure for general splines is not always possible. One special case which result in forward on-line recursive algorithm was developed.

The adoptation of state space model clarifies the crucial point of recursive spline smoothing algorithm, which were neglected before. However, the adjustment of weighting parameter ρ_{ij} is not included in this report. The development of adaptive algorithms and the extension to nonregular grid splines will be presented in the future.

2. Basis Functions For Piecewise Hermite Polynomials (p.H.p)

A systematic method for deriving the bases of piecewise Hermite polynomials is reported here which can be extended for the general cases of 2-D and higher dimensional p.H.p.'s. Among the polynomial splines, piecewise Hermite polynomials are of particular importance because they have the following attractive properties: (i) The p.H.p. requires data in only a finite local area around the point of approximation, hence it can be implemented recursively (finite local

support). (ii) The continuity of p.H.p. can be prespecified which results in a smooth function for curve fitting. (iii) p.H.p. is of finite error bounds and the error behavior can be analyzed accurately. We denote ${}_nC^m_{n+m}$ the collection of all piecewise 2-D functions that have continuity up to nth order partial derivatives on the first variable (ξ), mth order partial derivatives on the second variable (η), and the n+mth mixed partial derivatives on these two variables. Because of the requirements of continuities on the boundaries of every piece (grid), the order for the polynomial which meets these boundary conditions are 2(n+1) for the first variable and 2(m+1) for the second variable, which is the so-called "piecewise Hermite polynomial (p.H.p.)."

a. Cardinal Method

Consequently, the p.H.p. of order 2(n+1) of polynomials of the first variable and order 2(m+1) of the second variable would form a basis of the linear space ${}_nC^m_{n+m}$. The total number of the p.H.p. bases is 4(n+1)(m+1) in each grid.

If a 2-D partition $\Pi(\xi,\eta)$ called observation region is given as shown in Figure 16, with (N-1) intervals in the variable ξ, and (M-1) intervals in the variable η, the p.H.p. of each grid defined by

$$[\xi_i \le \xi \le \xi_{i+1},\ \eta_j \le \eta \le \eta_{j+1}]\ ,\quad i=1,2,\ \ldots\ (N-1),\quad j=1,2,\ \ldots\ (M-1)$$

can be constructed by the following procedure.

(i) Because of the independence of the first and the second variables, the p.H.p. basis can be formed by the tensor produce [6] of the 2(n+1) p.H.p. basis factors (polynomials of degree, 2n+1) in the first variable ξ, $H^{2(n+1)}(\xi)$, and the 2(m+1) p.H.p. basis factors in the second variable η, $H^{2(m+1)}(\eta)$. Hence, the general p.H.p. basis in one grid can be expressed by the combination of $H^{2(n+1)}(\xi)$ and $H^{2(m+1)}(\eta)$.

(ii) According to the continuity requirements along the boundaries (4 end points of the grid), a set of 4(n+1) (m+1) equations are posed on these p.H.p. basis factors.

(iii) Each of the p.H.p. basis factors is obtained by using the boundary and normalization conditions.

b. Bases for Twice Differentiable Continuous Functions in Both Directions

To illustrate the above procedure, an example for consturcting the p.H.p. bases of ${}_2C_4^2$ is given in the sequel.

If a function $f(\xi,\eta)$ is constructed by using the p.H.p. bases from the given data vector

$$\left[f, \frac{\partial}{\partial\xi}, \frac{\partial}{\partial\eta}, \frac{\partial^2 f}{\partial\xi\,\partial\eta}, \frac{\partial^2 f}{\partial\xi^2}, \frac{\partial^2 f}{\partial\eta^2}, \frac{\partial^3 f}{\partial\xi^2\,\partial\eta}, \frac{\partial^3 f}{\partial\xi\,\partial\eta^2}, \frac{\partial^4 f}{\partial\xi^2\,\partial\eta^2}\right]^T = \underline{X}^T$$

at every grid end point, the resultant function $f(\xi,\eta)$ will satisfy the continuity requirement of ${}_2D_4^2$ and thus belong to ${}_2C_4^2$.

(i) The whole function can be expressed as the summation of piecewise functions in every grid,

$$f(\xi,\eta) = \sum_{i-1}^{N-1} \sum_{j=1}^{M-1} f_{i,j}(\xi,\eta) \tag{48}$$

where

$$f_{i,j}(\xi,\eta) = \begin{cases} f_{i,j}(\xi,\eta) & \xi_i \le \xi \le \xi_{i+1},\ \eta \le \eta_{j+1} \\ 0 & \text{otherwise} \end{cases}$$

And in turn every piecewise function $f_{i,j}(\xi,\eta) \in {}_2C_4^2$ can be represented in terms of the p.H.p. bases,

$$f_{i,j}(\xi,\eta) = \sum_{k=0}^{1} \sum_{\ell=0}^{1} [\underline{W}_{k,\ell}]^T \underline{X}_{i+k,\, j+\ell} \tag{49}$$

where

$$W_{k,\ell} = [\phi_{0k}(\xi)\cdot\psi_{0\ell}(\eta),\ \phi_{1k}(\xi)\cdot\psi_{0\ell}(\eta),\ \phi_{0K}(\xi)\cdot\psi_{1\ell}(\eta),\phi_{1k}(\xi)\cdot\psi_{1\ell}(\eta),\ \phi_{2k}(\xi)\cdot\psi_{0\ell}(\eta),$$

$$\phi_{0k}(\xi)\cdot\psi_{2\ell}(\xi),\ \phi_{2k}(\xi)\cdot\psi_{1\ell}(\eta),\ \phi_{1k}(\xi)\cdot\psi_{2\ell}(\eta),\ \phi_{2k}(\xi)\cdot\psi_{2\ell}(\eta)] \qquad (50)$$

$$\underline{X}_{i+k,\ j+\ell} = \underline{X}\Big|_{\xi=\xi_{i+k},\ \eta=\eta_{j+\ell}}$$

$\phi_{0k}(\xi)$, $\phi_{1k}(\xi)$, $\phi_{2k}(\xi)$ and $\psi_{0\ell}(\eta)$, $\psi_{1\ell}(\eta)$, $\psi_{2\ell}(\eta)$ are the p.H.p. basis factors of variables ξ and η, respectively. Each of these is a polynomial of degree 5. The complete p.H.p. bases are shown in Equation (50).

The next step is to determine every p.H.p. basis factor mentioned above. Each p.H.p. basis factor is a fifth degree polynomial, hence, there are six undetermined coefficients for each factor and a total of 72 parameters to be calculated for these 12 basis factors.

(ii) From the given boundary conditions, these coefficients are determined. This results in 12 sets of simultaneous equations corresponding to 12 factors

$$D_\xi^p\ \phi_{qk}\ (\xi_{i+d}) = \delta_{pq}\cdot\delta_{kd} \qquad (51) \quad \text{and} \quad D_\eta^p\ \psi_{qk}(\eta_{j+d}) = \delta_{pq}\cdot\delta_{kd} \qquad (52)$$

where $D_\xi^p = d^p/d\xi^p$ and $D_\eta^p = d^p/d\eta^p$, $\delta_{a,b}$ is Kronecker δ function

$p = 0,1,2$; $q = 0,1,2$; and $k = 0,1$; $d = 0,1$.

(iii) By solving the simultaneous Equations (51) and (52), every undetermined coefficient can be obtained. Because only one term is nonzero for each basis factor (for a fixed q and k in Equations (51) and (52)), it is easy to obtain all the six coefficients of this particular fifth degree polynomial.

It can be written without difficulty that

$$\phi_{00}(\xi) = (\xi-\xi_{i+1})^3\ [a_0(\xi-\xi_i)^2 + b_0(\xi-\xi_i) + c_0]$$

The unknown parameters a_0, b_0, and c_0 can be calculated by the equalities $\phi_{00}(\xi_i) = 1$, $\phi'_{00}(\xi_i) = 0$ and $\phi''_{00}(\xi_i) = 0$. For simplicity, let $\Delta\xi \triangleq \xi_{i+1} - \xi_i = 1$ and $\lambda \overset{\Delta}{=} \xi - \xi_i$, then the foregoing equation becomes

$$\phi_{00}(\lambda) = (\lambda-1)^3 (-6\lambda^2-3\lambda-1) = 1-10\lambda^3+15\lambda^4-6\lambda^5$$

By the same token, all the other basis factors can be derived similarly.

$$\phi_{10}(\lambda) = \lambda - 6\lambda^3 + 8\lambda^4 - 3\lambda^5 \qquad \psi_{00}(\nu) = 1 - 10\nu^3 + 15\nu^4 - 6\nu^5$$

$$\phi_{20}(\lambda) = \tfrac{1}{2}\lambda^2 - \tfrac{3}{2}\lambda^3 - \tfrac{3}{2}\lambda^4 - \tfrac{1}{2}\lambda^5 \qquad \psi_{10}(\nu) = \nu - 6\nu^3 + 8\nu^4 - 3\nu^5$$

$$\phi_{01}(\lambda) = 10\lambda^3 - 15\lambda^4 + 6\lambda^5 \qquad \psi_{20}(\nu) = \tfrac{1}{2}\nu^2 - \tfrac{3}{2}\nu^3 + \tfrac{3}{2}\nu^4 - \tfrac{1}{2}\nu^5$$

$$\phi_{11}(\lambda) = -4\lambda^3 + 7\lambda^4 - 3\lambda^5 \qquad \psi_{01}(\nu) = 10\nu^3 - 15\nu^4 + 6\nu^5$$

$$\phi_{21}(\lambda) = \tfrac{1}{2}\lambda^3 - \lambda^4 + \tfrac{1}{2}\lambda^5 \qquad \psi_{11}(\nu) = -4\nu^3 + 7\nu^4 - 3\nu^5$$

$$\psi_{21}(\nu) = \tfrac{1}{2}\nu^3 - \nu^4 + \tfrac{1}{2}\nu^5$$

where $\Delta\eta \overset{\Delta}{=} \eta_{i+1} - \eta_i = 1$ and $\nu \overset{\Delta}{=} \eta - \eta_i$.

After obtaining all the 12 basis factors, the p.H.p. bases are readily obtained by combining these factors as indicated in Equation (50).

c. Conclusions

A systematic method for deriving the bases of p.H.p.'s has been described. This approach can be used either for high order polynomials or high dimensional functions. As can be observed from the example, the derivation procedure and the calculations are simple. Consequently, this procedure is suitable for practical applications. For example, the piecewise Hermite polynomials of ${}_2C_4^2$ can be used to estimate the second derivations of 2-D image. Because of the smoothness property of piecewise Hermite polynomials, they can also be used to filter out the noise of a stochastic signal.

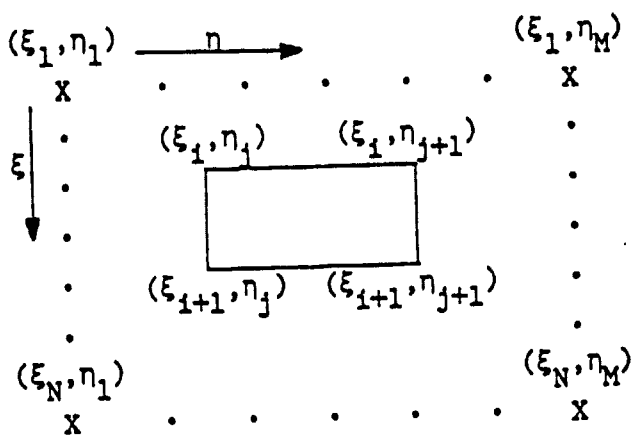


Figure 16 Observation Region

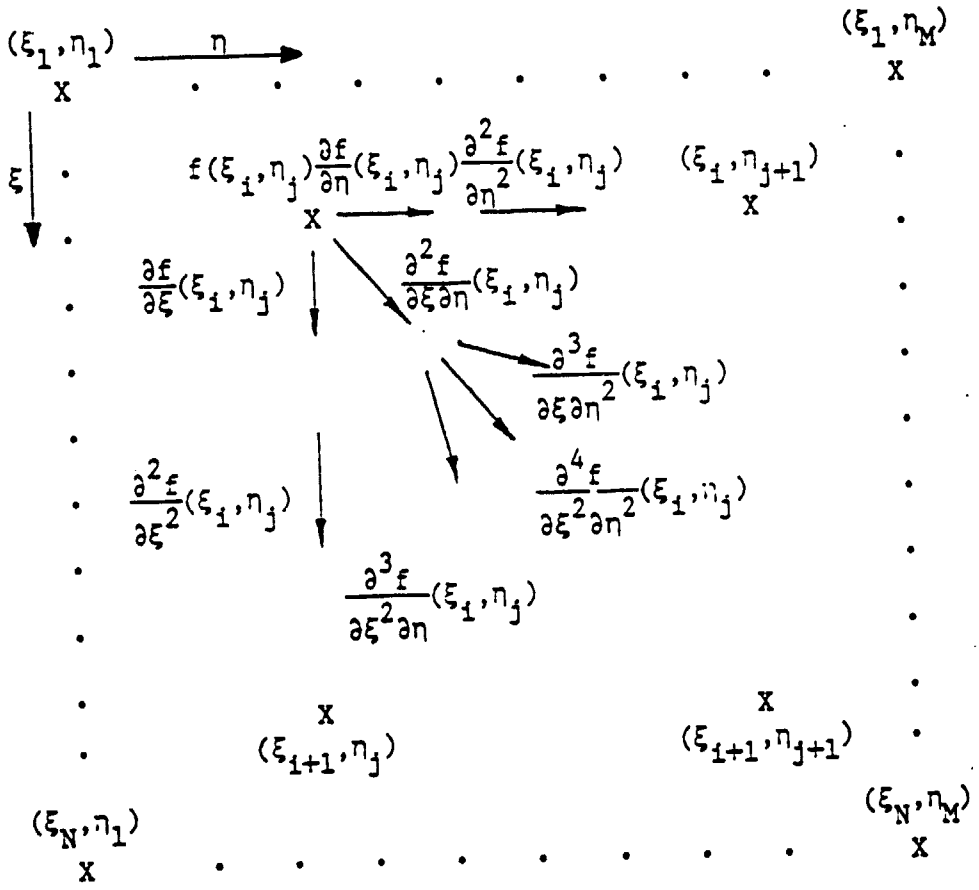


Figure 17 Given Data for $_2C_4^2$

IV. LITERATURE CITED

[1] Kim, C. S., Marynowki, R. C., and Shen, C.N., "Obstacle Detection Using Stabilized Rapid Estimation Scheme With Modified Decision Tree," Proceedings of JACC, Philadelphia, PA, October 1978.

[2] Kim, C.S., and C. N. Shen, "Design of a Recursive Vector Processor Using Polynomial Splines," Proceedings of the 19th IEEE, CDC, 1980.

[3] Duda, R.O., and Hart, P.E., "Pattern Classification and Scene Analysis," John Wiley and Sons, New York, 1973.

[4] Fukanaga, K.J., "Statistical Methods of Pattern Recognition," Academic Press, New York, 1972.

[5] C.N. Shen, "Data Acquisition and Analysis of Range-Finding Systems for Space Construction," RPI Technical Report MP-809, Rensselaer Polytechnic Institute, October 1980, Troy, New York,12181.

[6] Jain, A.K.; "A Semicansal Model for Recursive Filtering of Two-Dimensional Images," IEEE Trans. Comput., Vol. C-26, pp. 343-350, April 1977.

[7] Habibi, A.; "Two-Dimensional Bayesian Estimate of Images," Proc. IEEE, Vol. 60, No. 7, pp. 878-883, July 1972.

[8] Woods, J.W., and C. H. Radewan, "Kalman Filtering in Two-Dimensions," IEEE Trans. Inform. Theory, Vol. II-23, pp. 473-482, July 1977.

[9] Ahlberg, J.H., E. N. Nilson, and J.L.Walsh; The Theory of Splines and Their Applications, NY: Academic Press, 1967.

[10] Schultz, M.H.; Spline Analysis, NJ: Prentice-Hall, 1973.

[11] Hou, H.S. and H.C. Andrews; "Least Squares Image Restoration Using Spline Basis Functions," IEEE Trans. Comput., Vol. 26, pp. 856-873, Sept. 1977.

[12] Hou, H.S., and H. C. Andrews; "Cubic Splines for Image Interpolation and Digital Fitlering," IEEE Trans. Acoust., Speech, Signal Processing, Vol. ASSP-26, pp. 508-517, Dec. 1978.

[13] Prenter, P.M., Splines and Variational Methods, New York: John Wiley and Sons, Inc., 1975.

[14] Guzman, A., "Decomposition of a Visual Scene Into Three Dimensional Bodies," AFIPS Proc. Fall Joint Computer Conf., 33, pp. 291-304 (1968).

[15] Gelfand, I.M., and S. V.Fomin; Calculus of Variations, NJ: Prentice-Hall, 1963.

[16] Marzetta, T.L.; "Two-Dimensional Linear Prediction: Autocorrelation Arrays, Minimum-Phase Prediction Error Filters, and Reflection Coefficient Arrays," IEEE Trans. Acoust., Speech and Signal Processing, Vol. ASSP-28, pp. 725-733, Dec. 1980.

LITERATURE CITED- Continued

[17] Strintzis, M.G.; "Comments on 'Two-Dimensional Bayesian Estimate of Images," Proc. IEEE, Vol. 64, No. 8. pp. 1255-1257, August 1976.

[18] Barry, P.E., R. Gran, and C. R. Waters; "Two Dimensional Filtering--A State Estimator Approach," Proc. of 15th CDC, pp. 613-618, 1976.