On Single-scanline Camera Calibration

Radu Horaud, Roger Mohr, Boguslaw Lorecki

Short Papers

On Single-Scanline Camera Calibration

Radu Horaud, Roger Mohr, and Boguslaw Lorecki

Abstract— A method for calibrating single scanline CCD cameras is described in this paper. We show that the more classical 2-D camera calibration techniques are necessary but not sufficient for solving for the 1-D camera calibration problem. We propose a model for single scanline cameras, and we provide a two-step procedure for estimating its parameters. We also show how the extrinsic camera parameters can be determined geometrically without making explicit the intrinsic camera parameters. The accuracy of the calibration method is analyzed through an application example.

Index Terms—Camera calibration, cross-ratio, intrinsic and extrinsic camera parameters, linear least squares estimation, perspective projection, single scanline CCD cameras.

I. INTRODUCTION

In many applications of computer vision, 1-D (single scanline) cameras may replace 2-D (matrix) cameras. Inspection of parts, for example, is a task that requires, among others, two main constraints: efficiency and accuracy. Linear cameras fit both these constraints for the following reasons. First, the 1-D signal they provide is easier and faster to process than 2-D images. Second, the current size of a linear camera may reach 4096 pixels. Compare this with the size of a matrix camera (at an affordable price) of 512×512 pixels. We conclude that linear cameras are well suited for tasks that do not involve high-level processing and for which accuracy is a crucial issue, such as inspection.

More generally, the calibration of a camera allows one to establish a relationship between image measurements and 3-D measurements. This relationship is useful for a variety of tasks:

- 3-D reconstruction using either passive stereo or active ranging,
- mobile-robot navigation,
- 3-D object recognition by matching a CAD model to 2-D data,
- hand/eye calibration that allows one to relate measurements in the camera-centered frame to the robot-centered frame, and
- 3-D inspection of parts.

The calibration of a matrix camera has received a lot of attention in the past, and robust techniques are now available [3]–[5], [8], [9]. Surprisingly enough, to the best of our knowledge, linear-camera calibration is not described in the technical literature. In [2], a couple of single scanline CCD cameras are used for detecting obstacles in front of a mobile robot. A cylindrical lens that is placed in front of each camera widens the field of view without modifying the geometry of the sensor, e.g., Fig. 1. Motion self-calibrates the camera pair for the purpose of stereo. However, it is assumed that the *viewing planes* (see below) associated with the two cameras are coplanar.
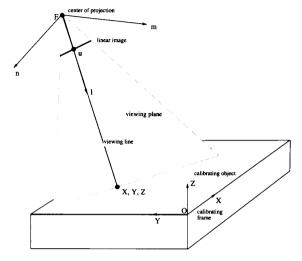
Fig. 1. A general view of a linear camera, a calibrating object, and their associated frames.

The calibration method described here could be used to verify that the two cameras are actually coplanar.

In this paper, we show that currently available camera calibration methods are necessary but not sufficient for solving the linear-camera calibration problem. We propose a two-step camera calibration method that uses the classical approach in conjunction with the *cross-ratio*, a well known projective invariant [6]. We define a camera-centered coordinate frame and determine the extrinsic parameters of the camera, that is, the rigid transformation between the calibrating coordinate frame and the camera frame.

II. TWO-DIMENSIONAL CAMERA CALIBRATION

Central projection or the pin-hole model is a good approximation of the geometry of an imaging system. Using central projection, the 2-D camera calibration problem is described by the following equations [3], [1]:

$$u = \frac{m_{11}X + m_{12}Y + m_{13}Z + m_{14}}{m_{31}X + m_{32}Y + m_{33}Z + m_{34}} \qquad (1)$$

$$v = \frac{m_{21}X + m_{22}Y + m_{23}Z + m_{24}}{m_{31}X + m_{32}Y + m_{33}Z + m_{34}}. \qquad (2)$$

These equations describe a point defined in a calibrating (world) frame projected onto the image. The world coordinates of the 3-D point are X, Y, and Z, and the image coordinates of its projection are u and v. Notice that these equations have another possible interpretation. They describe the line passing through the center of projection of the camera and through an image point, in world coordinates. The 2-D camera calibration problem is the problem of determining the parameters m_{11} through m_{34}. If at least six correspondences are available, and noticing that the m_{ij}'s are defined up to a scale factor, the calibration problem becomes a linear optimization problem [3]. We mean by correspondence i that the set $\{X_i, Y_i, Z_i, u_i, v_i\}$ is known: the coordinates of a 3-D point are

known in world coordinates and its 2-D projection is known in image coordinates as well.

III. ONE-DIMENSIONAL CAMERA CALIBRATION

The best way to describe a single scanline (or linear) camera is to think of a matrix camera for which only one line of pixels is active. A 3-D point projects onto this linear image at pixel u obeying (1). This equation can be written in matrix form:

$$\begin{bmatrix} su \\ s \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}. \quad (3)$$

This projective transformation can be further decomposed into a rigid transformation (rotation and translation) followed by a one dimensional projection:

$$\begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}$$

$$= \begin{bmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where α_u and u_0 describe the projection, $R = [r_{ij}]$ is an orthonormal matrix describing the rotation, and $t = (t_1 t_2 t_3)$ is a translation vector. One can easily notice that $m_{31} = r_{31}$, $m_{32} = r_{32}$, and $m_{33} = r_{33}$, and that we have

$$m_{31}^2 + m_{31}^2 + m_{31}^2 = 1 \quad (4)$$

since R is an orthonormal matrix. This constraint is used in [3] to obtain a better estimation of the projection parameters.

Moreover, in order to be seen by the camera, the 3-D point is constrained to lie in the *viewing plane*, the plane defined by the center of projection and by the active line of pixels, e.g., Fig. 1. Without loss of generality, the equation of the viewing plane can be written as

$$X = pY + qZ + r. \quad (5)$$

By substituting (5) into (1), renaming variables, and noticing that (1) describing a projection is defined up to multiplicative term, we obtain

$$u = \frac{n_1 Y + n_2 Z + n_3}{n_4 Y + n_5 Z + 1}. \quad (6)$$

Equations (5) and (6) define the linear-camera model, that is, they describe the line passing through the center of projection and an image pixel, in world coordinates. This line is also called the *viewing line*.

This model has eight parameters associated with it: n_1, n_2, n_3, n_4, n_5, and p, q, and r. The problem of determining the camera model is the problem of estimating these parameters. In the Appendix, we prove that eight parameters are sufficient for describing a linear-camera model. Hence, the linear-camera calibration problem is twofold:

- Estimate the parameters n_1, n_2, n_3, n_4, and n_5. If at least five correspondences are available (five sets $\{Y_i, Z_i, u_i\}$), this may be mapped into a linear optimization problem similar to the case of 2-D camera calibration.
- Estimate p, q, and r. These parameters can be estimated if and only if world points belonging to the viewing plane are provided.
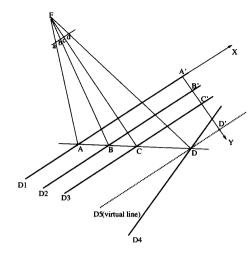

Fig. 2. The structure of the calibrating object being viewed by the linear camera.

Alternatively, one may use (1) and (4) instead of (6), as done in [3]. The drawback is that the camera model is described by 11 parameters instead of eight.

Interesting enough, the linear-camera calibration problem is similar to the problem of calibrating an active range sensor [1]. In the latter case, two calibration steps are also necessary: one for calibrating the camera and one for determining the equation of the plane of light in world coordinates. However, in our case, the two steps are independent while in the case of active range-sensing they are not.

IV. A TWO-STEP CALIBRATION METHOD

We begin by describing the structure of the calibrating object used by our method. The choice for this structure will soon be made clear. The calibrating object is shown on Fig. 2. It consists of four coplanar straight lines, D_1, D_2, D_3, and D_4. The first three are mutually parallel, and D_4 makes an angle with the direction of these three. The equations of these lines are known in the calibrating (world) coordinate frame, defined as follows: the origin of this frame is not constrained to a particular position. The X axis is parallel to D_1, and the Y axis is perpendicular to D_1. The Z axis is perpendicular to the $X - Y$ plane. In this frame and in the plane $Z = 0$, the equations of the four lines can be written as

$$\begin{array}{ll} Y = 0 & (D_1) \\ Y = \alpha & (D_2) \\ Y = \beta & (D_3) \\ Y = \gamma X + \delta & (D_4) \end{array}$$

where the parameters α, β, γ, and δ are fixed and determine the structure of the calibrating object.

A. Estimating n_1, n_2, n_3, n_4, and n_5

When a linear camera "looks" at this object, it "sees" four image points a, b, c, d, which are the projections of A, B, C, D, the intersections of the viewing plane with D_1, D_2, D_3, D_4, e.g., Fig. 2. Notice that the X coordinates of A, B, C, D need not be known. Due to the structure of the calibrating object and the particular choice of the calibrating frame, the Y and Z coordinates of A, B, C are known independently of the position and orientation of the viewing plane (with respect to the calibration frame).

Therefore, one can establish the following correspondences: $\{Y_A, Z_A, u_a\}$, $\{Y_B, Z_B, u_b\}$, and $\{Y_C, Z_C, u_c\}$. By moving the

calibration object in the Y direction and/or in the Z direction with known increments, one can establish three correspondences at each new position of the calibrating object. Each correspondence i verifies (6), which can be written as

$$Y_i n_1 + Z_i n_2 + n_3 - u_i Y_i n_4 - u_i Z_i n_5 = u_i. \qquad (7)$$

For n such correspondences, $n \geq 5$, we obtain

$$\begin{bmatrix} Y_1 & Z_1 & 1 & -u_1 Y_1 & -u_1 Z_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ Y_n & Z_n & 1 & -u_n Y_n & -u_n Z_n \end{bmatrix} \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \end{bmatrix} = \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix}. \qquad (8)$$

This is an overconstrained system of n linear equations in five unknowns of the form $AX = B$. By premultiplying with the transpose of A, we obtain a linear system with five equations and five unknowns: $A^t AX = A^t B$. Notice that $X = (A^t A)^{-1} A^t B$ is the optimal solution in the least squared sense and that the covariance matrix associated with the estimated solution may be computed as well [7, pp. 528–532]. Since we wish to compute not only the parameters n_1 through n_5 but also the covariance matrix, it is most convenient to use Gauss–Jordan elimination to perform the linear algebra.

B. Estimating p, q, r

In order to estimate p, q, and r, sets of 3-D coordinates of points belonging to the viewing plane must be available. Obviously a linear camera cannot provide point-to-point correspondences in the general case. The structure of the calibrating object has been purposively designed to be able to provide, in conjunction with the camera, 3-D point coordinates that belong to the viewing plane.

We recall the definition of the cross-ratio of four collinear points:

$$\text{cross-ratio}_{A,B,C,D} = (A, B, C, D)$$
$$= (CA/CB)/(DA/DB)$$

CA being the algebraic distance from C to A. The fundamental property of the cross-ratio is that it is invariant by central or parallel projection. Therefore, we have

$$(A, B, C, D) = (a, b, c, d).$$

We consider a *virtual* line D_5 parallel and coplanar with D_1, D_2, D_3 and passing through the point D. These four parallel lines D_1, D_2, D_3, and D_5 intersect the Y axis in A', B', C', and D'. From the projective invariance of the cross-ratio, we have

$$(A, B, C, D) = (A', B', C', D')$$

and hence

$$(C'A'/C'B')/(D'A'/D'B') = (ca/cb)/(da/db).$$

We conclude that if the linear camera sees simultaneously D_1, D_2, D_3, and D_4 (in fact, the camera sees the points A, B, C, and D), then one is able to compute the cross-ratio r, $r = (a, b, c, d)$ from the image measurements and therefore determine the position of D' along the Y axis, say $Y = \lambda$. We have

$$r = (C'A'/C'B')/(D'A'/D'B')$$
$$= \left(\frac{\beta}{\beta - \alpha}\right) / \left(\frac{\lambda}{\lambda - \alpha}\right)$$

and

$$\lambda = \frac{\alpha \beta}{r\alpha + (1 - r)\beta}.$$

The coordinates of D are therefore given by the intersection of the object line D_4 with the virtual line D_5

$$Y = \gamma X + \delta$$
$$Y = \lambda$$
$$Z = 0.$$

Moreover, the point D is guaranteed to belong to the viewing plane and its position is computed via image measurements independent of the camera parameters. By moving the calibration object in the Y and/or Z directions with known increments, one may determine the coordinates of one viewing-plane point at each new position of the calibrating object. Thus, we obtain k points, $k \geq 3$, each point j verifying the equation

$$Y_j p + Z_j q + r = X_j.$$

By combining these equations, we obtain

$$\begin{bmatrix} Y_1 & Z_1 & 1 \\ \vdots & \vdots & \vdots \\ Y_k & Z_k & 1 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} X_1 \\ \vdots \\ X_k \end{bmatrix}. \qquad (9)$$

This system of equations has the same structure as (8), and hence the same method can be used to find a solution.

An alternative for fitting a plane to a set of 3-D points is to minimize the sum of the squared distances from each point to the sought plane. Let the plane be described by

$$aX + bY + cZ = d$$

with the constraint $a^2 + b^2 + c^2 = 1$. The least squared criterion to be minimized is

$$Q = \sum_{j=1}^{k} (aX_j + bY_j + cZ_j - d)^2 + \mu(1 - a^2 - b^2 - c^2)$$

where μ is the Lagrange multiplier. In practice, the plane computed by (9) is exaclty the same as the plane computed by minimizing the above criterion.

V. THE EXTRINSIC PARAMETERS

The linear camera model is described by eight parameters that we just defined and estimated. We recall that this model describes a line passing through the center of projection and an image pixel, in world frame. This frame may well be viewed as the sensor frame, and for a number of applications it is sufficient to have such a model. Nevertheless, in some applications of camera calibration, such as hand/eye calibration, it is necessary to calibrate the camera several times from several viewpoints with respect to the calibration object and to determine the rigid transformation between a camera frame and the calibrating frame [10]. The parameters of this rigid transformation are also called the extrinsic camera parameters, and they will be made explicit in this section.

First we define a camera frame; see Fig. 1. The origin of the camera frame is the center of projection F. Therefore, the translation vector associated with the transformation is the vector from O to F having as coordinates the coordinates of F in the calibrating frame

$$\vec{t} = \begin{bmatrix} X_F \\ Y_F \\ Z_F \end{bmatrix}.$$

The axes of the camera Cartesian coordinate frame are defined by the following vectors:

- the direction of an arbitrarily chosen viewing line $\vec{l}$;
- the normal to the viewing plane $\vec{m}$, and
- the cross-product of these vectors $\vec{n}$.

Without loss of generality, we suppose that the camera has 1024 pixels, u_1 through u_{1024}. The viewing line passing through the central pixel u_{512} is given by the following equations (see (5) and (6)):

$$(n_1 - n_4 u_{512})Y + (n_2 - n_5 u_{512})Z + n_3 - u_{512} = 0$$
$$-X + pY + qZ + r = 0.$$

The direction of this line $\vec{l}$ is given by the cross-product of the normals to these two planes. These two normals are

$$\vec{N}_1 = (0 \quad n_1 - n_4 u_{512} \quad n_2 - n_5 u_{512})^t$$
$$\vec{N}_2 = (-1 \quad p \quad q)^t.$$

The axes of the camera frame have the following unit direction vectors:

$$\vec{l} = \frac{\vec{N}_1 \times \vec{N}_2}{\|\vec{N}_1 \times \vec{N}_2\|}$$
$$\vec{m} = \frac{\vec{N}_2}{\|\vec{N}_2\|}$$
$$\vec{n} = \vec{l} \times \vec{m}.$$

These three vectors determine the rotation matrix associated with the rigid transformation

$$R = (\vec{l} \, \vec{m} \, \vec{n}).$$

To determine the translation one may easily notice that all the viewing lines pass through the center of projection, in particular, the viewing lines associated with the first and last pixels (or any other two pixels). The coordinates of F are given by combining two viewing-line equations

$$(n_1 - n_4 u_1)Y_F + (n_2 - n_5 u_1)Z_F + n_3 - u_1 = 0$$
$$(n_1 - n_4 u_{1024})Y_F + (n_2 - n_5 u_{1024})Z_F + n_3 - u_{1024} = 0$$
$$-X_F + pY_F + qZ_F + r = 0.$$

It is worthwhile to notice that the extrinsic parameters embedded by the rotation matrix R and the translation vector $\vec{t}$ have been determined geometrically without making explicit the intrinsic camera model.

VI. RESULTS AND CONCLUSION

Using the method described here and for one calibration, we used 150 measurements to estimate the n_i's and 50 measurements to estimate p, q, and r. Indeed, for each position of the calibration object with respect to an initial position, points A, B, and C are used for estimating n_1 through n_5 while the point D is used for estimating p, q, and r; see Section IV.

An important merit of any calibration procedure is its accuracy. Tables I and II summarize the results of one calibration. In addition, the covariance matrices are provided with the values obtained for the sought parameters [7].

The only parameter not accurately estimated is r, the intersection of the viewing plane with the X axis. The reason for this problem is that the set of 3-D points used for estimating the viewing plane equation were not evenly distributed in this plane; they were concentrated. One way to overcome this problem is to consider a calibrating object that has more than one oblique line.

TABLE I
NOMINAL VALUES AND COVARIANCE MATRIX FOR
THE FIRST STEP OF THE CALIBRATION METHOD

n_1	n_2	n_3	n_4	n_5
46.76	7.47	130.62	0.0008	0.0122

Covariance matrix				
0.0083	0.0006	0.039	0.0	0.0
0.0006	0.0136	0.001	0.0	0.0
0.039	0.001	0.197	0.0	0.0
0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0

TABLE II
NOMINAL VALUES AND COVARIANCE MATRIX FOT
THE SECOND STEP OF THE CALIBRATION METHOD

p	q	r
−0.434	−0.023	18.836

Covariance matrix		
0.011	0.0008	−0.216
0.0008	0.0187	−0.0115
−0.216	−0.0115	4.269

APPENDIX
WHY EIGHT PARAMETERS?

The single-scanline camera model describes the central projection of a 3-D point onto a straight line (the linear set of pixels). The 3-D point generally being described in a world frame that is different from the camera frame, a frame change must be performed prior to the projection. Six parameters are necessary to describe the frame change transformation: three for rotation and three for translation.

The camera frame is chosen such that the linear image is perpendicular to one of the axes (the optical axis) and the origin is at the center of projection. Two parameters are necessary to describe this central projection (four in the case of a 2-D image): the pixel coordinate of the intersection of the image with the optical axis and the ratio of the focal length with the pixel size.

It is straightforward to see that the variation of the value of one of these eight parameters modifies the camera model and that this modification cannot be compensated by the variation of any of the remaining parameters. Hence, the model is described by eight independent parameters.

REFERENCES

[1] R. C. Bolles, J. H. Kremers, and R. A. Cain, "A simple sensor to gather three-dimensional data." SRI International, Tech. Rep. 249, July 1981.
[2] R. A. Brooks, A. M. Flynn, and T. Marill, "Self calibration of motion and stereo vision for mobile robots," in *4th Int. Symp. Robotics Res.* (Santa Cruz, CA, Aug. 1987). Cambridge, MA: MIT Press, 1987, pp. 277–286.
[3] O. D. Faugeras and G. Toscani, "The calibration problem for stereo," in *Proc. Comput. Vision Pattern Recognit.* (Miami Beach, FL), June 1986, pp. 15–20, .
[4] W. I. Grosky and L. A. Tamburino, "A unified approach to the linear camera calibration problem," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, no. 7, pp. 663–671, July 1990.
[5] R. K. Lenz and R. Y. Tsai, "Techniques for calibration of the scale factor and image center for high accuracy 3-d machine vision metrology," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 10, no. 5, pp. 713–720, Sept. 1988.

[6] R. Mohr and L. Morin, "Relative positioning from geometric invariants," in *Proc. Comput. Vision Pattern Recognit. Conf.* (Lahaina, Maui, HI), June 1991, pp. 139–144.

[7] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Wetterling, *Numerical Recipes in C: The Art of Scientific Computing*. New York: Cambridge, 1988.

[8] P. Puget and T. Skordas. "An optimal solution for mobile camera calibration," in O. Faugeras, Ed., *Computer Vision-ECCV 90, Proc. 1st European Conf. Comput. Vision* (Antibes, France, Apr. 1990). New York: Springer Verlag, Apr. 1990, pp. 187–198.

[9] R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE J. Robotics Automat.*, vol. RA-3, no. 4, pp. 323–344, Aug. 1987.

[10] R. Y. Tsai and R. K. Lenz, "Real time versatile robotics hand/eye calibration using 3D machine vision," in *Proc. IEEE Int. Conf. Robotics Automat.* (Philadelphia, PA), Apr. 1988, pp. 554–561.

Performance Evaluation of a Class of M-Estimators for Surface Parameter Estimation in Noisy Range Data

Muhammad J. Mirza and Kim L. Boyer

Abstract— Depth maps are frequently analyzed as if the errors are normally, identically, and independently distributed. This noise model does not consider at least two types of anomalies encountered in sampling: a few large deviations in the data, often thought of as outliers, and a uniformly distributed error component arising from rounding and quantization. Estimates based on the least squares (LS) philosophy, appropriate under a Gaussian noise assumption, can be excessively influenced by such rogue observations. The theory of robust statistics formally addresses these problems and is efficiently used in a robust sequential estimator (RSE) of our own design. The RSE assigns different weights to each observation based on the maximum likelihood analysis when it is supposed that the errors follow a t distribution which, being heavy tailed, represents the outliers more realistically. This work extends this concept to several well known maximum-likelihood estimators (M-estimators). Since most M-estimators do not have a target distribution, the weights are obtained by a simple linearization and then embedded in the same RSE algorithm. We include experimental results over a variety of real and synthetic range imagery acquired from independent sources. We evaluate the performance of these estimators under different noise conditions. We highlight the effects of tuning constants and the necessity of simultaneous scale and parameter estimation. We emphasize the choice of the t distribution model because of its relative performance and simple mechanism for simultaneous scale estimation. We emphasize the potential application of this approach in simultaneous parametrization and surface-based range image segmentation.

I. INTRODUCTION

Surface perception plays a key role in range image understanding and 3-D object recognition. Typically, a series of processes over several levels of abstraction are required to extract useful information from depth maps. The goal of much of this processing is to organize the data in terms of common characteristics or features. The function approximation approach has generally been used to compute the local

or global properties of the underlying surface [3]–[6]. However, the samples are often obtained from an unknown statistical population, and attempts to fit models to this type of data must account for systematic grouping errors as well as random measurement errors.

After selecting an appropriate model for the underlying surface, the most popular approach is to use least squares analysis to estimate the model parameters. This presupposes that errors in the data are normally, identically, and independently distributed. However, the first two of these assumptions are frequently inappropriate. In fact, the error distributions are more likely to be leptokurtic and/or contaminated by occasional anomalous values, or outliers.

The outliers play havoc with estimators using ordinary least squares. Even without outliers problems occur because range images consist of noisy quantized samples of object surfaces, and quantization errors are not Gaussian distributed. To remedy the problem of nonnormal errors, new statistical techniques known as robust methods have been developed that are insensitive to such departures in the data [7], [8].

Although many informal robust concepts did exist in the computer-vision literature, strong interest in developing formal robust algorithms has emerged recently. Early on, Haralick [9] stressed the necessity of developing robust algorithms. Forstner [10] used formal robust statistics in his work on reliability analysis in mensuration problems. Besl *et al.* [11] developed a robust window operator for image smoothing and derivative estimation. Haralick *et al.* [12] employed robust statistics in pose estimation. Chen [13] used recursive QR updating and outlier rejection for robust curve growing. Least median of squares, although not suitable for sequential implementation, has been used in various vision tasks to remove salt and pepper noise [14] and for the first stage of a surface reconstruction algorithm [15].

Mirza and Boyer [1], [2] have developed a robust sequential estimator (RSE) based on maximum likelihood analysis for a wide class of nonnormal distributions. The distinguishing feature of this estimator is that it uses W-estimation for initial estimates while using w-estimation for sequential updates (these terms will be explained in the following discussion). The specific implementation in [1] and [2] was based on a t distribution error model. This is a novel robust approach since it models the errors by a heavy tailed distribution that is more prone to produce outliers, and it achieves computational efficiency through sequential updating. Our weighted sequential scheme is different from other sequential approaches, e.g., Kalman filtering, which is optimal only for Gaussian noise. Recursive QR updating [13] rejects the possible outliers and then equally weights the accepted observations. Our algorithm assigns weights to each observation based on a t distribution error model and, hence, rejects the outliers by down weighting. The other outstanding feature of the RSE is that it considers the outliers as *unusual* rather than as *unimportant* observations. This fact has been effectively exploited in our work on simultaneous organization and parametrization of surfaces in range data.

The selection of the t distribution error model for the RSE was based primarily on intuition, and no thorough experimental investigation of other error distribution models and maximum-likelihood estimators (M-estimators) was undertaken. This work, then, follows up by investigating a variety of well known M-estimators (a class of robust estimator defined below) when embedded in the same RSE. We compare them on the basis of efficiency and performance for different known and unknown noise models.