dg 12/4 NLM

Trans. #:899380 ILL : 111697475

Article **Location : Walter Sci/Eng Library Books (Level F) Quarto TA1632 .D35**

Borrower : SEA **Date :** 12/3/2013 6:10:45 PM

UMC,*MNU,CNS,ILU,GZM,CSL

AUTHOR : Kim, D.Y., Kim, J,J., Meer, P., Minm, D., and Rosenfeld
TITLE : Image Understanding Workshop ; [proceedings].
IMPRINT : San Francisco, CA ; Distributed by Morga

Article : Robust computer vision; A least median of squares based approach
Volume : **Issue :** **Month :** **Year :** 1989 **Pages :** 1117-1134

ODYSSEY
ISxN: 1525-3287 **System ID :** OCLC 39908102 **Copyright Compliance :** CCL
Patron : Hoover, Adam
NOTES :

Default **MaxCost: 30.00IFM**

Loaned To:
SEA - Clemson University Libraries
ILL Unit, Cooper Library
Interlibrary Loan--Box 343001
116 Sigma Drive
Clemson, SC 29634-3001

Loaned From : Documents to U : 15 Andersen Library, Univ. of MN 222 – 21st Ave S. Minneapolis, MN 55455

ROBUST COMPUTER VISION: A LEAST MEDIAN OF SQUARES BASED APPROACH

Dong Yoon Kim[1]
J. John Kim
Peter Meer
Doron Mintz
Azriel Rosenfeld

Center for Automation Research
University of Maryland
College Park, MD 20742-3411

ABSTRACT

Regression analysis (fitting a model to noisy data) is a basic techniques in computer vision. Robust regression methods which remain reliable in the presence of various types of noise are therefore of considerable importance. We present a new paradigm based on the least median of squares (LMS) method proposed by Rousseeuw (1984). The method yield the correct result even when 49.9% of the data is severely corrupted. Its efficiency in the presence of Gaussian noise can be improved by complementing it with a weighted least squares based procedure. The high time complexity of the LMS algorithm can be reduced by a Monte Carlo type speed up technique. The algorithm was successfully applied to mode-based cluster detection, line fitting to noisy data, and designing a local operator performing robust plane fitting in images.

1. INTRODUCTION

Regression analysis (fitting a model to noisy data) is an important statistical tool frequently employed in computer vision for a large variety of tasks. Tradition and ease of computation have made the least squares method the most popular form of regression analysis. The least squares method achieves the optimum results when the underlying error distribution is Gaussian. However, the method becomes unreliable if the noise has non-Gaussian components and/or if outliers (samples with values far from the local trend) are present in the data. The outliers may be the result of clutter, large measurement errors, or impulse noise corrupting the data. At a transition between two homogeneous regions of the image, samples belonging to one region may become outliers for fits to the other region.

Three concepts are usually employed to evaluate a regression method: relative efficiency, breakdown point, and time complexity. The *relative efficiency* of a regression method is defined as the ratio between the lowest achievable variance for the estimated parameters (the Cramer-Rao bound) and the actual variance provided by the given method. The efficiency also depends on the underlying noise distribution. For example, in the presence of Gaussian noise the mean estimator has an asymptotic (large sample) efficiency of 1 (achieving the lower bound) while the median estimator's efficiency is only $\frac{2}{\pi} = 0.637$ (Mosteller and Tukey, 1977).

The *breakdown point* of a regression method is the smallest amount of outlier contamination which may force the value of the estimate outside an arbitrary range. For example, the breakdown point of the mean is 0 since a single large outlier can corrupt the result. The median remains unchanged if less than half of the data are contaminated, yielding asymptotically the maximum breakdown point, 0.5.

The *time complexity* of the least squares method is $O(np^2)$ where n is the number of data points and p is the number of parameters to be estimated. Feasibility of the computation requires a time complexity of at most $O(n^2)$.

A new, improved regression method should provide:

[1]Science Exchange Program Fellow from the Agency for Defense Development, Taejeon, Korea.

- reliability in the presence of various types of noise, i.e., good asymptotic and small sample efficiency;
- protection against a high percentage of outliers, i.e., a high breakdown point;
- a time complexity not much greater than that of the least squares method.

Many statistical techniques have been proposed which satisfy some of the above conditions. These techniques are known as *robust regression* methods. In Section 2 a review of robust regression methods is given. In Section 3 the least median of squares (LMS) method is discussed in detail. This method is then applied to several computer vision problems: in Section 4 to mode-based cluster detection; in Section 5 to line fitting; and in Section 6 to noise cleaning in images through robust local plane fitting.

2. ROBUST REGRESSION METHODS

The early attempts to introduce robust regression methods involved straight line fitting. In one class of methods the data is first partitioned into two or three nearly equal sized parts ($i \leq L; L < i \leq R; R < i$) where i is the index of the data and $L = R$ in the former case. The slope β_1 and the intercept β_0 of the line are found by solving the system of nonlinear equations

$$\begin{aligned} &\underset{i \leq L}{\text{med}} \, (y_i - \beta_0 - \beta_1 x_i) = \underset{i > R}{\text{med}} \, (y_i - \beta_0 - \beta_1 x_i) \\ &\underset{\text{for all } i}{\text{med}} \, (y_i - \beta_0 - \beta_1 x_i) = 0 \end{aligned} \tag{1}$$

where med represents the median operator applied to the set defined below it. The breakdown point of the method is $0.5/k$ where k is the number of partitions (2 or 3) since the median is used for each part separately. Brown and Mood (1951) investigated the method for $k = 2$, and Tukey introduced the resistant line procedure for $k = 3$ (see Johnstone and Velleman, 1985).

Another class of methods uses the slopes between each pair of data points without splitting up the data set. Theil (1950) estimated the slope as the median of all $n(n-1)/2$ slopes which are defined by n data points. The breakdown point of these methods is 0.293 since at least half the slopes should be correct in order to obtain the correct estimate. That is, if ϵ is the fraction of outliers in the data we must have $(1-\epsilon)^2 \geq 0.5$. The intercept can be estimated from the input data by employing the traditional regression formula.

The theory of multidimensional robust estimators was developed in the 70's. The basic robust estimators are classified as M-estimators, R-estimators and L-estimators (Huber, 1981).

The M-estimators are the most popular robust regression methods. These estimators minimize the sum of a symmetric, positive-definite function ρ of the residuals r_i. (A residual is defined as the difference between the data point and the fitted value.) For the least squares method $\rho(r_i) = r_i^2$. The M-estimates of the parameters are obtained by iteratively solving the minimization problem

$$\min \sum_i \rho (r_i) \tag{2}$$

Notice that the sought parameters are represented through the residuals. Several ρ functions have been proposed. Huber (1981) employed the squared error for small residuals and the absolute error for large residuals. Andrews (1974) used a squared sine function for small and a constant for large residuals. Beaton and Tukey's (1974) biweight is another example of these ρ functions. Holland and Welsch (1977) developed algorithms for solving the numerical problems associated with M-estimators.

R-estimators are based on ordering the set of residuals. Jaeckel (1972) proposed obtaining the parameter estimates by solving the minimization problem

$$\min \sum_i a_n(R_i) \, r_i \tag{3}$$

where r_i is the residual; R_i is the location of the residual in the ordered list, i.e., its rank; and a_n is a score function. The score function must be monotonic and $\sum_i a_n(R_i) = 0$. The most frequently used score function is that of Wilcoxon: $a_n(R_i) = R_i - (n+1)/2$. Since $| a_n(R_i) | \leq (n-1)/2$ the largest residuals caused by outliers cannot have a too large weight. Scale invariance (independence from the variance of the noise) is an important advantage of R-estimators over M-estimators. Cheng and Hettmansperger (1983) presented an iteratively reweighted least squares algorithm for solving the minimization problem associated with R-estimators.

The L-estimators employ linear combinations of order statistics. The median and α-trimmed mean based methods belong to this class. It is important to notice, however, that the mean ($\alpha = 0$) is a least squares

estimate, while the median can be regarded also as the M-estimate obtained for $\rho = |r_i|$. Various simulation studies have shown that L-estimators give less satisfactory results than the other two classes (Heiler, 1981).

In spite of their robustness for various distributions the M-, R- and L-estimators have breakdown points that are less than $1/(p+1)$, where p is the number of parameters in the regression (Li, 1985). For example, in planar surface fitting we have $p = 3$, and the breakdown point is less than 0.25, making it sensitive to outliers.

Recently several robust estimators having breakdown points close to 0.5 were proposed. Siegel (1982) introduced the repeated median (RM) method of solving multidimensional regression problems. Suppose p parameters are to be estimated from n data samples. A parameter is estimated in the following way: First, for each possible p-tuple of samples the value of the parameter is computed yielding a list of $C(n,p)$ (the binomial coefficient) terms. Then the medians for each of the p indices characterizing a p-tuple are obtained recursively. When the list has collapsed into one term, the result is the RM estimate of the parameter. Once a parameter has been estimated, the amount of computation can be reduced for the remaining $p-1$ parameters.

For example, let $p = 3$ and suppose that we start by estimating the parameter β_2 of the planar fit

$$z = \beta_0 + \beta_1 x + \beta_2 y \tag{4}$$

First the values

$$\beta_2(i,j,k) = \frac{(z_i - z_j)(x_j - x_k) - (z_j - z_k)(x_i - x_j)}{(y_i - y_j)(x_j - x_k) - (y_j - y_k)(x_i - x_j)} \tag{5}$$

are computed for all the triplets defined by $i \neq j \neq k$. The estimate is then

$$\hat{\beta}_2 = \underset{i}{\text{med}} \underset{j(\neq i)}{\text{med}} \underset{k(\neq i,j)}{\text{med}} \beta_2(i,j,k) \tag{6}$$

The parameter β_1 is estimated next, by applying the same algorithm for $p = 2$ to the data $z_i - \hat{\beta}_2 y_i$. Similarly the value of β_0 is obtained by taking the median of the samples $z_i - \hat{\beta}_2 y_i - \hat{\beta}_1 x_i$. The breakdown point of the repeated median method is 0.5 since all the partial median computations are performed over the entire data set. Computation of the median is $O(n)$ and thus the time complexity of the RM method is high, of $O(n^p)$ order. The Gaussian efficiency of the method was found experimentally as being only around 0.6 (Siegel, 1982). The τ-estimate introduced by Yohai and Zamar (1988) achieves high efficiency and high breakdown point simultaneously but its time complexity is very high.

The least median of squares robust regression method proposed by Rousseeuw(1984) also achieves 0.5 breakdown point. The relative efficiency of the method can be improved by combining it with least squares based techniques. The time complexity can be reduced by a Monte Carlo type speed-up technique. In the next section we describe the LMS method in detail.

3. THE LEAST MEDIAN OF SQUARES METHOD

Rousseeuw (1984) proposed the least median of squares (LMS) method in which the parameters are estimated by solving the nonlinear minimization problem

$$\min \underset{i}{\text{med}}\, r_i^2 \tag{7}$$

That is, the estimates must yield the smallest value for the median of residuals computed for the entire data set. The precise meaning of the minimization is clarified below. As in the case of the repeated median method, the LMS minimization problem (7) is also solved by a search in the space of possible estimates generated from the data.

Let a distinct p-tuple of data points be denoted by the indices $i_1, \ldots, i_p$. For every p-tuple the values of $p-1$ parameters (all except the intercept, β_0) are computed. The intercept is chosen as variable because it the easiest to manipulate in computations. There are $C(n;p)$ p-tuples and the minimization is performed in two steps. First, for a given p-tuple the intercept value $\beta_0(i_1, \ldots, i_p)$ that solves the minimization problem

$$\min \underset{i}{\text{med}}\, r_i^2 \qquad \text{given} \quad \beta_j(i_1, \ldots, i_p),\ j = 1, \ldots, (p-1). \tag{8}$$

is obtained. The procedure is repeated for every p-tuple and the one yielding the smallest value for (8) supplies the LMS estimates of *all* the parameters.

Steele and Steiger (1986) proposed to solve (8) by a mode estimation technique. (The mode of a histogram, i.e., probability distribution, is the location of its largest value.) Let $s_1 \leq s_2 \leq \cdots \leq s_n$, be a sorted list of values

obtained from the data. To locate the mode of the underlying continuous distribution the minimum of the differences

$$D_i = s_{i+\lfloor n/2 \rfloor} - s_i \qquad i = 1,2, \ldots, \lceil n/2 \rceil \tag{9}$$

is sought. The functions $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ are the floor and ceiling functions respectively. The minimum difference will appear where most of the values are similar and since the list was sorted this coincides with the peak of the distribution. The resolution of the method is controlled by the distance between the two samples (9) and local maxima are avoided by choosing the largest possible distance $\lfloor n/2 \rfloor$.

Assume that the mode was found to correspond to $i = k$. Its value is then taken as

$$M_k = (s_{k+\lfloor n/2 \rfloor} + s_k)/2 \tag{10}$$

The minimum difference can be written as function of M_k:

$$D_k = M_k - s_k = s_{k+\lfloor n/2 \rfloor} - M_k \tag{11}$$

and the following ordering relations are also valid:

$$|M_k - s_i| \begin{cases} \leq D_k & \text{if } k < i < k+\lfloor n/2 \rfloor \\ \geq D_k & \text{if } 1 \leq i < k \text{ or } k+\lfloor n/2 \rfloor < i \leq n \end{cases} \tag{12}$$

because the s_i are ordered. From (11) and (12) we obtain

$$D_k = \underset{i}{\text{med}} |M_k - s_i| \tag{13}$$

since $\lfloor n/2 \rfloor$ difference values are always less than or equal to D_k while the same number of differences are larger than or equal to it.

When the above procedure is performed for the n intercept values computed for a given p-tuple of data points, the resulting minimum difference is the solution of the minimization problem (8). For n data points $C(n,p)$ minimum differences $D_k(i_1, \ldots, i_p)$ are obtained and the smallest one yields the LMS estimate for the p parameters, i.e, the solution of (7). A detailed example of the LMS algorithm is given in Section 5.

The breakdown point of the least median squares method is 0.5 because all the median computations are over the whole data set. The time complexity of the method, however, is very high. There are $O(n^p)$ p-tuples and for each of them the sorting takes $O(n \log n)$ time. Thus the amount of computation required for the basic LMS algorithm is $O(n^{p+1} \log n)$, prohibitively large. Notice that this complexity is valid only if $p \geq 2$, since for $p = 1$ only sorting is required.

The time complexity is reduced to practical values when a Monte Carlo type speed-up technique is employed in which a $Q \ll 1$ probability of error is tolerated. Let ϵ be the fraction of data contaminated by outliers. Then the probability that all m different p-tuples chosen at random will contain at least one or more outliers is

$$P = [\, 1 - (1-\epsilon)^p \,]^m \tag{14}$$

Note that $1-P$ is the probability that at least one p-tuple from the chosen m has only uncorrupted samples and thus the correct parameter values can be recovered. The smallest acceptable value for m is the solution of the equation

$$P = Q \tag{15}$$

rounded upward to the closest integer, and is independent of n, the size of the data. The amount of computation becomes $O(m\ n \log n)$. This time complexity reduction is very significant. In Table 1 the values of m are given for three values of Q, p between 2 and 8, and ϵ between 0.05 and 0.499. For example, if $p = 3$, $Q = 0.01$ and $\epsilon = 0.3$ then $m = 11$ for any n. Thus, when at most 30 percent of the data is contaminated by outliers, by choosing 11 triplets for the computation of the LMS robust planar fit, the probability of having the whole set of triplets corrupted is 0.01.

Table 1: Number of p-tuples required to achieve the probability of error Q as a function of ϵ, the fraction of outliers.

$Q = 0.01$

$p \setminus \epsilon$	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.499
2	2	3	4	5	6	7	9	11	13	16
3	3	4	5	7	9	11	15	19	26	35
4	3	5	7	9	13	17	24	34	48	71
5	4	6	8	12	17	26	38	57	90	144
6	4	7	10	16	24	37	59	97	165	289
7	4	8	12	20	33	54	92	163	301	579
8	5	9	15	26	44	78	143	272	548	1158

$Q = 0.005$

$p \setminus \epsilon$	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.499
2	3	4	5	6	7	8	10	12	15	19
3	3	5	6	8	10	13	17	22	30	40
4	4	5	8	11	14	20	27	39	56	82
5	4	6	10	14	20	29	43	66	103	166
6	4	7	12	18	28	43	68	111	189	333
7	5	9	14	23	37	62	106	187	346	667
8	5	10	17	29	51	90	164	313	631	1333

$Q = 0.001$

$p \setminus \epsilon$	0.05	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.45	0.499
2	3	5	6	7	9	11	13	16	20	24
3	4	6	8	10	13	17	22	29	38	52
4	5	7	10	14	19	26	36	50	72	107
5	5	8	12	18	26	38	57	86	134	216
6	6	10	15	23	36	56	89	145	247	434
7	6	11	18	30	49	81	138	244	451	869
8	7	13	22	38	66	117	214	408	822	1737

When Gaussian noise is present in addition to outliers the relative efficiency of the LMS method is low. Rousseeuw (1984) has shown that the LMS method converges for large sample sizes as $n^{-1/3}$, much slower than the usual $n^{-1/2}$ for maximum likelihood estimators. To compensate for this deficiency he proposed combining the LMS method with a weighted least squares procedure which has high Gaussian efficiency. Either one-step weighted least squares or an M-estimator with Hampel's redescending function can be employed. For more detail see the book of Rousseeuw and Leroy (1987).

The breakdown point of the combined method is still 0.5 since the standard deviation of the noise, σ, is estimated from the LMS part and thus the weights in the least squares procedure can be correctly determined. The standard deviation estimate

$$\hat{\sigma} = 1.4826 \left(1 + \frac{5}{n-p}\right) \operatorname*{med}_i |r_i| \tag{16}$$

can be immediately obtained since the median of the residual is the value returned by the LMS procedure for the parameter estimates. Note that the usual robust standard deviation estimate does not contain the term $5/(n-p)$. This term is recommended by Rousseeuw and Leroy (1987) as a finite sample correction factor.

In the following sections we describe different applications of the LMS technique to computer vision problems. The number of estimated parameters p increases from one in mode-based clustering, to three in the computation of local planar fits.

4. MODE-BASED CLUSTER DETECTION

Given n data points in a plane, finding the centers of their clusters is a classical problem in pattern recognition. The number of clusters K is usually known a priori. Most clustering algorithms have four steps (see Jain and Dubes (1988) for a monograph on the subject):

Step 1. The n points are initially partitioned arbitrarily into K groups.

Step 2. The center of each group is estimated.

Step 3. The points are re-partitioned based on their distances to the current cluster centers.

Step 4. Steps 2 and 3 are repeated until the changes no longer exceed a convergence threshold.

The weight center of the points belonging to a cluster is often taken as its center, i.e., the mean of the x and y coordinates of the points. This method is known as the K-means clustering technique. The K-means approach, however, introduces artifacts whenever the data points are corrupted by non-uniformly distributed noise. The time complexity of the K-means technique is $O(n)$.

Another basic method of cluster detection involves searching for regions of high density—that is, mode seeking. In the mode seeking approach uniformly distributed noise points do not offset the mode of the original data unless the noise destroys the data entirely. Two classes of mode seeking methods are described in the literature. In the first class the points are grouped into bins and local maxima of the resulting multi-dimensional histogram are sought. The method is very sensitive to bin size; too small a size yields false alarms, while too large a size may smooth out significant maxima. The method requires large storage space, but its time complexity is still $O(n)$. In the other class of mode seeking methods the distances between all the possible point pairs are taken into consideration for clustering. These methods have the time complexity $O(n^2)$.

In Section 3 we have shown that mode estimation is part of the least median of squares algorithm and therefore application of the algorithm to mode-based cluster detection is immediate. As the examples will show, cluster detection using the LMS method accurately locates the cluster centers even in non-uniform background noise.

Assume that after the $(l-1)$st iteration the n data points were partitioned into K clusters having centers $[x_{c_i}(l-1), y_{c_i}(l-1)]$, where $i = 1,2, \ldots, K$. The ith cluster contains $n_i(l)$ data points.

The lth iteration of the clustering algorithm starts by independently computing the updated x and y coordinates for the new cluster centers. The same LMS based procedure is employed. Let $x_1, \ldots, x_{n_i(l)}$ be the list of abscissas for the points currently belonging to the ith cluster. After the application of the LMS algorithm ($p = 1$ in this case), the mode of the underlying distribution (9) is taken as a first approximation to the abscissa of the ith cluster center, $X_{c_i(l)}$. The robust standard deviation $\hat{\sigma}_{i,x}(l)$ (16) of this one-dimensional distribution is also estimated.

One-step weighted least square fitting is performed next for each two-dimensional cluster to increase the relative efficiency of the clustering algorithm to Gaussian noise. The distance between the jth data point (x_j, y_j) belonging to the ith cluster and the current center of the cluster is defined as:

$$d_{i,j}^2(l) = \left(\frac{x_j - X_{c_i(l)}}{\hat{\sigma}_{i,x}(l)} \right)^2 + \left(\frac{y_j - Y_{c_i(l)}}{\hat{\sigma}_{i,y}(l)} \right)^2 \tag{17}$$

At the initial partition, the directional standard deviations are set to one. Hence, (17) gives the Euclidean distance initially, and gives normalized distances at subsequent partitions. Notice that we assumed that each cluster has an elliptical shape with the major axes parallel to one of the sides of the image. For rotated clusters a cross-term including the correlation coefficient estimated from the data should be added to (17). Based on its distance the data point can be allocated with the weight $w_j(l)$:

$$w_j(l) = \begin{cases} 1 & d_{i,j} \leq 2.96 \\ 0 & d_{i,j} > 2.96 \end{cases} \tag{18}$$

where the threshold 2.96 corresponds to 98.76 percent of the two-dimensional normal distribution being taken as

inliers. The coordinates $[\, x_{c_i}(l),\, y_{c_i}(l)\,]$ of the ith cluster center after the lth iteration are then the solutions of the least squares problems

$$\min_{x_{c_j(l)}} \sum_{j=1}^{n_i(l)} w_j(l)\,[\, x_j - x_{c_j(l)}\,]^2 \qquad \min_{y_{c_j(l)}} \sum_{j=1}^{n_i(l)} w_j(l)\,[\, y_j - y_{c_j(l)}\,]^2 \tag{19}$$

To further increase the Gaussian efficiency of the clustering algorithm the directional standard deviations are reestimated from the two-dimensional data taking into account only the inliers of the cluster:

$$\hat{\sigma}_{i,x}(l) = \frac{\sum_{j=1}^{n_i(l)} w_j(l)\,[\, x_j - \overline{x}_i\,]^2}{\sum_{j=1}^{n_i(l)} w_j(l) - 1} \qquad \hat{\sigma}_{i,y}(l) = \frac{\sum_{j=1}^{n_i(l)} w_j(l)\,[\, y_j - \overline{y}_i\,]^2}{\sum_{j=1}^{n_i(l)} w_j(l) - 1} \tag{20}$$

where $\overline{x}_i$ and $\overline{y}_i$ are the mean coordinates of the ith cluster's inlier points.

Once the updated coordinates of all the K cluster centers are found the new, $(l+1)$st partitioning of the data points can be performed. The partitioning is done by finding the closest cluster center for each point. The distance (17) is computed from a given point to every cluster center, employing the new directional standard deviations of the given cluster. The point is allocated to the cluster yielding the smallest distance. The $(l+1)$st iteration can now start. The iterations are repeated until no change occurs between two consecutive partitions.

Since only one parameter (the abscissa or the ordinate) is estimated by the LMS algorithm the time complexity is given by the sorting of the data points and is $O(n \log n)$ per iteration. Putting the restriction on the data that every point lie on an integer grid of a fixed size, which is the most common case in computer vision, the sorting can be accomplished in linear time. The complexity of the LMS algorithm is then only $O(n)$, the same as the K-means algorithm mentioned at the beginning of the section.

To compare the LMS based clustering method with other techniques three typical test data sets were used: circularly symmetric clusters (CSC), CSC's in uniform background noise, and CSC's in non-uniform background noise. Three cluster detection methods were applied to each data set. Besides the K-means and K-LMS methods the K-weighted-means technique proposed by Jolion and Rosenfeld (1988) was also investigated. In this method each data point is given a weight according to the density of points in its vicinity and thus the time complexity is $O(n^2)$.

Figure 1a shows the first data set, two CSC's having 100 and 150 points, distributed as Gaussians with means (10, 10) and (35, 35) and standard deviations 5. The experimental results are given in Table 2.

Table 2: Detected Cluster Centers. Noiseless case.

Method	Center 1	Center 2
K-means	(10.01, 10.06)	(35.06, 34.74)
K-weighted-means	(10.77, 10.93)	(35.14, 34.92)
K-LMS	(9.86, 9.98)	(35.01, 34.81)

As expected, Gaussian clusters are best estimated by the maximum likelihood K-means method but the other two algorithms also give correct answers.

In Figure 1b 75 uniformly distributed points were added as background noise to the data set.

Table 3: Detected Cluster Centers. Uniform Background Noise.

Method	Center 1	Center 2
K-means	(11.11, 11.49)	(34.86, 33.46)
K-weighted-means	(9.85, 9.90)	(35.14, 34.70)
K-LMS	(10.55, 10.33)	(35.17, 34.76)

As shown in Table 3, only the result of K-means is significantly affected by the noise.

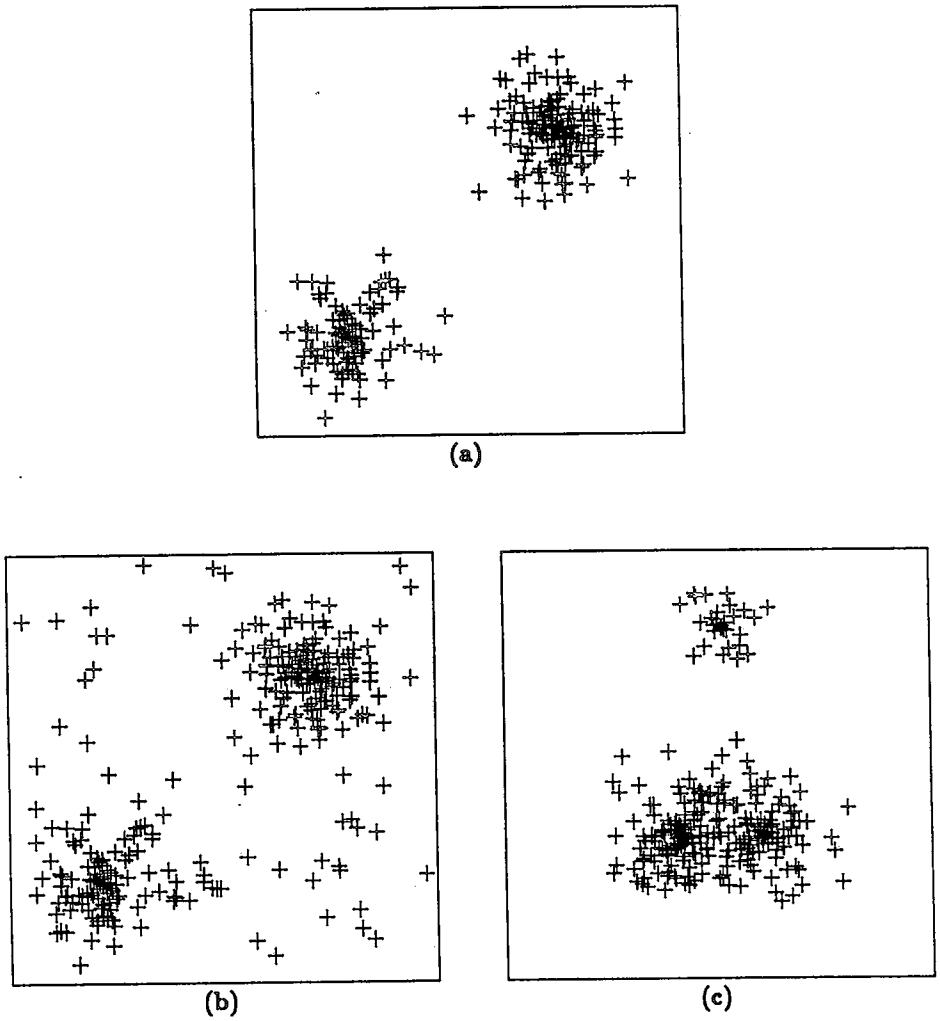

Figure 1. Test data for the clustering experiments. a) Two circularly symmetric clusters. b) The clusters from a) embedded in uniform background noise. c) Two data clusters with a biased noise cluster (top).

Figure 1c shows two CSC's having 100 and 150 points, distributed as Gaussians with means $(20, 15)$ and $(30, 15)$ and standard deviations 5, together with 30 biased noise points also normally distributed around $(25, 40)$ with standard deviation 3.

Table 4: Detected Cluster Centers. Non-uniform Background Noise.

Method	Center 1	Center 2
K-means	(23.95, 15.26)	(25.22, 40.18)
K-weighted-means	(20.00, 14.91)	(28.84, 18.21)
K-LMS	(20.59, 15.06)	(30.25, 15.28)

The results given in Table 4 show that only the K-LMS method gives the correct result. The noise can also be regarded as a cluster not taken into account, i.e., the a priori information about the number of clusters was wrong. In this case the output of the K-LMS method can be employed for detecting the overlooked cluster. Subtracting the result from the original data and performing a clustering on the difference the third cluster is detected.

In our experiments with the K-LMS method the data points were restricted to a lattice with unit step size, i.e., the coordinates of a point were rounded to the nearest integers. While this coarse quantization was not present in the other two methods, the LMS based algorithm's performance was always equal or superior to them. The number of storage bins required by the K-LMS algorithm increases only linearly with the dimension of the feature vectors (two in our examples) since the modes are determined separately along each coordinate axis. This is another advantage of the K-LMS algorithm relative to the histogram based methods, in which the storage increases exponentially with the dimension of the feature vector.

5. LINE FITTING TO NOISY DATA

Detection of straight lines in noisy data containing fragmented segments is an important task in computer vision. The Hough transform, one of the most often employed methods, can already be classified as a robust technique since it has the ability to detect the longest line segment even if it comprises less than 50% of the data points. For purposes of comparison with the LMS based line fitting method to be described below we consider the *pairwise* variant of the Hough transform in which for each pair of data points (x_i,y_i) and (x_j,y_j) the values of the parameter pair (ρ, θ) are calculated from the equations

$$\begin{aligned} \rho &= x_i \cos\theta + y_i \sin\theta \\ \rho &= x_j \cos\theta + y_j \sin\theta \end{aligned} \tag{21}$$

A histogram in (ρ, θ) space is built for all combinations of pairs of points. The line segments are detected by finding the peaks of the histogram, i.e., its modes.

The disadvantage of any type of Hough transform method is due to the histogram usage. A histogram using a bin size that is too small may have a wrong mode, while a histogram using a bin size that is too large may yield estimates that are too coarse. The discretization of image space also makes the distribution of parameters in Hough space non-homogeneous and non-equiprobable. The time complexity of the pairwise Hough transform is $O(n^2)$.

In the previous sections we have shown how the least median of squares algorithm finds the mode of a distribution, and how this algorithm can be employed for clustering problems. The histogram bin size does not create artifacts for the K–LMS clustering method, since the resolution of its mode seeking procedure is controlled only by the distance employed when computing the differences (11) between the sorted samples. For n data points, however, the K–LMS algorithm requires an additional $O(n^2 \log n)$ processing time to detect the clusters in the Hough space.

Before proceeding to give the details of the LMS based line fitting method we will prove that a speed-up technique similar to the one described in Section 3 cannot be successfully applied to the pairwise Hough transform. For example, the number of sample pairs which can guarantee a 0.01 error probability for a speeded-up LMS line fitting method yields a much larger error probability for the Hough transform. In the data given in Figure 4 five out of the eleven data points are outliers, that is, ϵ is approximately 0.45. To achieve 0.01 probability of error in line fitting ($p = 2$), from Table 1 we see that an LMS based line fitting technique should employ only $m = 13$ pairs of points instead of the total 55. Let a be the number of pairs chosen (from the total 15) containing only inlier points. Similarly, let b be the number of chosen pairs (from the total 10) containing only outliers and let c be the number of pairs containing one inlier and one outlier point (a maximum of 30 such pairs). Several triplets of the a, b and c values yield Hough spaces from which the correct line fit cannot be recovered. The probability of such a triplet is

$$\text{Prob}\,(\,a=3,\,b=3,\,c=7 \mid m=13\,) = \frac{C(30,7)\,C(15,3)\,C(10,3)}{C(55,13)} = 0.0766. \tag{22}$$

Other triplets have similar probabilities and the sum of all the unfavorable cases is much higher than 0.01.

The least median of squares based line fitting method is an application (for $p = 2$) of the general procedure described in Section 3. To obtain the LMS estimates for the parameters β_0 and β_1 first the slopes $\beta_1(j_1,j_2)$ are computed for each pair of points (x_{j_1},y_{j_1}) and (x_{j_2},y_{j_2}):

$$\beta_1(j_1,j_2) = \frac{y_{j_1} - y_{j_2}}{x_{j_1} - x_{j_2}} \tag{23}$$

Then for every $\beta_1(j_1,j_2)$ the values

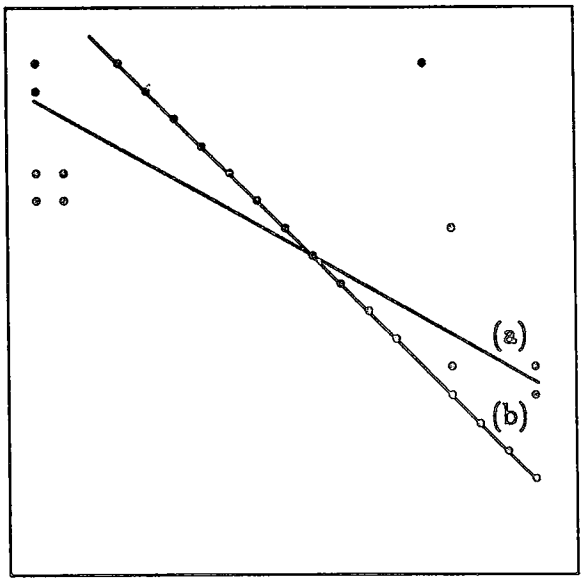

Figure 2. Line fits to data corrupted by symmetric noise. (a) Least squares. (b) MI and LMS algorithms.

$$\alpha_{j_1,j_2}(i) = y_i - \beta_1(j_1,j_2)\, x_i \qquad i = 1, \ldots, n \tag{24}$$

are sorted and the mode of the distribution (10) is computed by the procedure described in Section 3. The mode is taken as the intercept $\beta_0(j_1,j_2)$ and it was shown that the difference (13) is

$$\underset{i}{\text{med}}\,|\,\beta_0(j_1,j_2) - \alpha_{j_1,j_2}(i)\,| = \underset{i}{\text{med}}\,|\,\beta_0(j_1,j_2) - y_i + \beta_1(i,j)\, x_i\,| = \underset{i}{\text{med}}\,|\,r_i\,| \tag{25}$$

The initial values for the parameters of the fitted line are the ones yielding the minimum of (25) for all the m pairs of points considered in the speeded up algorithm.

One-step weighted least squares fit is performed next to increase the relative efficiency of the LMS based procedure to Gaussian noise. The ith data point is given the weight w_i depending on the value of the residual q_i obtained from the initial fit values:

$$w_i = \begin{cases} 1 & \dfrac{|\,q_i\,|}{\hat{\sigma}} \le 2 \\ \dfrac{3 - |\,q_i\,|}{\hat{\sigma}} & 2 < \dfrac{|\,q_i\,|}{\hat{\sigma}} \le 3 \\ 0 & 3 < \dfrac{|\,q_i\,|}{\hat{\sigma}} \end{cases} \tag{26}$$

where $\hat{\sigma}$ is the estimated standard deviation of the noise (16). The final estimates of the slope and intercept are the solutions of the classic weighted least squares minimization problem:

$$\min_{i} \sum_{i=1}^{n} w_i\, [y_i - \beta_1\, x_i - \beta_0]^2 \tag{27}$$

The algorithm has the LMS part's high breakdown point, but is made more efficient when the underlying residual distribution (i.e. the noise) is Gaussian. The time complexity of the LMS method is significantly reduced by employing the Monte-Carlo type of speed-up technique described in Section 3.

Three different line fitting methods were compared in our experiments: The traditional least squares approach, the median of intercepts (MI) method of Kamgar-Parsi *et al.* (1989), and the LMS based algorithm. In the MI method the intercept and slope is computed for every pair of points the medians of the obtained lists are the MI estimates of the two parameters. As was mentioned at the beginning of Section 2 algorithms in this class have a theoretical breakdown point of 0.293.

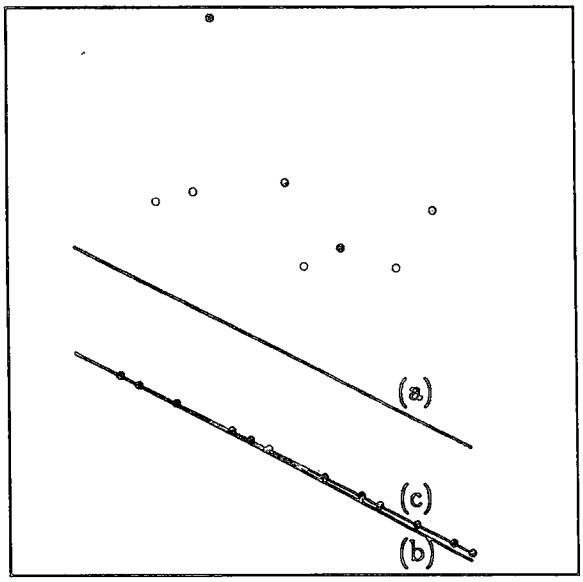

Figure 3. Line fits to data corrupted by asymmetric noise. (a) Least squares. (b) MI algorithm. (c) LMS algorithm.

The data shown in Figure 2 contains fraction $\epsilon = 0.42$ of outliers. While the least squares method fails to find the correct fit (a), the results of the MI and LMS algorithms are identical (b). Although the breakdown point of the median of intercepts method is 0.293, it was able to find the correct line due to the symmetry of the noise distribution.

In the second example (Figure 3) $\epsilon = 0.4$ and the data is corrupted by Weibull distributed asymmetric noise. The Weibull random process that was used had amplitude $A = 2$ and cumulative probability distribution $F(u)=0$ for $u<0$, $F(u)=1-e^{-u^2}$ for $u \geq 0$. The least square method (a) fails, the MI algorithm (b) produces a close approximation, and only the LMS algorithm (c) succeeds in finding the original line.

Two different line segments are combined in the third example (Figure 4). Six points belong to one line segment and five to the other, and thus $\epsilon = 0.45$ when we want to fit a line to this discontinuity. Only the LMS method (c) fits the line to the majority of the points.

For the above three examples, the Hough transform correctly finds the line segment corresponding to the majority of the points. However, the Hough transform may fail when systematic errors are present in the data. In Figure 5 a coarsely digitized line segment gave rise to the data points. In this case the three methods recover the correct line, but the Hough transform fails since several false modes are generated by the aligned data points.

We conclude that the robust LMS line fitting algorithm provides the best results for the types of data degradation that were investigated.

6. ROBUST LOCAL OPERATORS

Median and trimmed mean based local operators (L-estimators) have been employed in computer vision for a long time (see for example Bovik *et al.* (1987) for recent results). Recently M-estimators have also become popular. Kashyap and Eom (1988) treated an image as a causal autoregressive model driven by a noise process assumed to be Gaussian with a small percent of the samples (at most 8%) contaminated by impulse noise, i.e. outliers. By employing M-estimators the parameters of the autoregressive process were iteratively refined simultaneously with cleaning the outliers in the noisy image. Besl *et al.* (1988) proposed a hierarchical scheme in which local fits of increasing degrees were obtained by M-estimators. The different fits were compared through a robust fit quality measure to determine the optimal parameters. The authors' claim of a 0.5 breakdown point

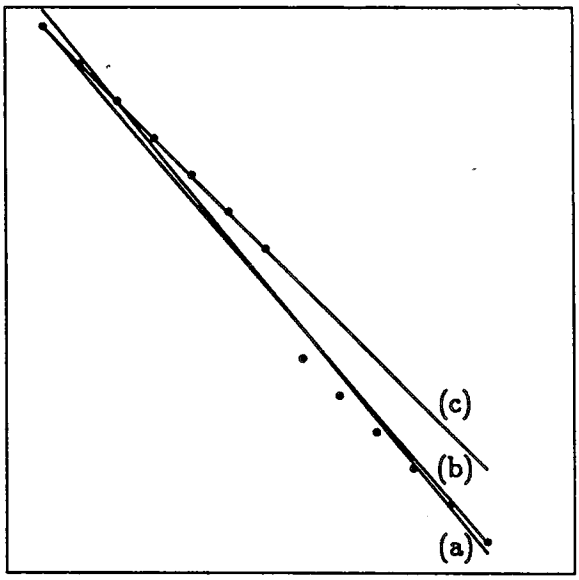

Figure 4. Line fits to a discontinuity. (a) Least squares. (b) MI algorithm. (c) LMS algorithm.

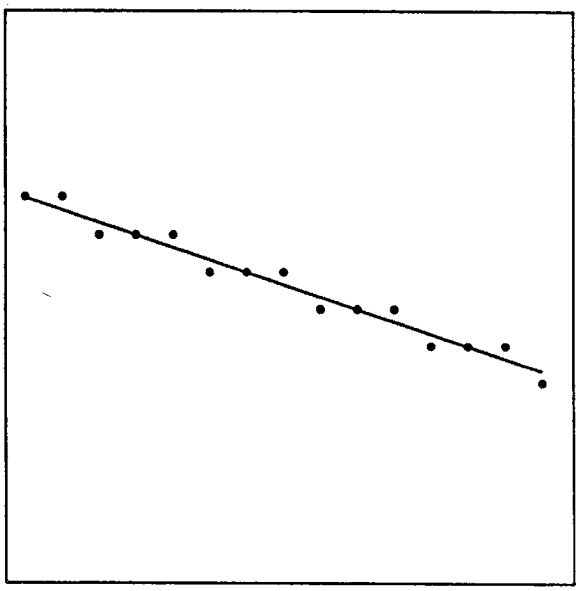

Figure 5. Line fit to quantization effects.

must be regarded with caution since for planar surfaces ($p = 3$) it would exceed the theoretical limit of 0.25 mentioned in Section 2. Haralick and Joo (1988) applied M-estimators to solve the correspondence problem between two sets of 2D perspective projections of model points in 3D. The correct pose solution was then obtained with up to 30% of the pairs mismatched.

The theoretical value of the breakdown point may not be achieved when local operators are applied to image discontinuities. Consider a noiseless, ideal step edge to which a 5×5 robust local operator was applied. Assume that 10 pixels in the window belong to the edge (high amplitude) and 15 to the background (low amplitude) and that the center of the window falls on a background pixel. Let the operator have the largest possible

breakdown point, 0.5. The operator returns the value of the majority of pixels, that is, the low amplitude of the background.

The image is then corrupted with fraction $\epsilon = 0.2$ of asymmetric noise driving the corrupted samples into saturation at the upper bound. Without loss of generality we can assume that only 3 of the pixels belonging to the background were corrupted in the processing window. There are now 13 pixels with high amplitudes and the operator returns, incorrectly, a high value similar to the amplitude of the edge. Thus, even when the fraction of outliers is much smaller than the theoretical breakdown point of a robust estimator, the operator may systematically fail near transitions between homogeneous regions in images. At transitions, samples of one region are outliers (noise) when fitting a model to the other region and a small fraction of additional noise may reverse the class having the majority.

The size of the local operator also limits ϵ, the maximum amount of tolerated contamination. For a one-dimensional window $2n+1$ pixels long at most n pixels should be corrupted, yielding $\epsilon \leq n/(2n+1)$, which only in the limit is 0.5. For example, if $n = 4$ (window length 9) an operator with breakdown point 0.5 tolerates only $\epsilon = 0.44$ contamination.

Least median of squares based local operators perform the algorithm described in Section 5. In every window, the parameters minimizing the median of squares are obtained. Their values can then be employed in various ways. We describe here the smoothing of images corrupted by asymmetric (impulse) noise. This application is of special interest since most smoothing methods fail to achieve good results for this type of noise.

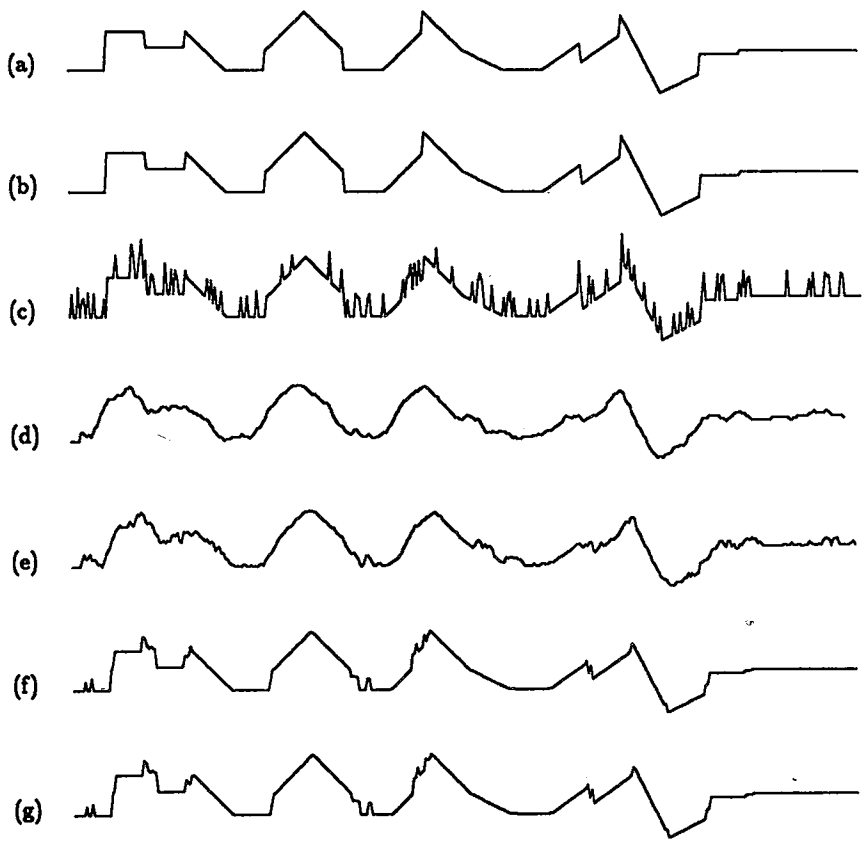

Figure 6. Local line fitting operators applied to one-dimensional data. a) Noiseless data. b) Result of LMS algorithm with speed-up applied to a). c) Noisy data, $\epsilon = 0.22$ fraction of the samples corrupted with Weibull noise. d) Least squares method. e) Result of an M-estimator (Hubel's ρ function). f) LMS algorithm with speed-up. g) LMS algorithm without speed-up.

In a smoothing algorithm the value of the fit in the center of the window becomes the new pixel amplitude. The window coordinates of the center can be taken as (0,0) and thus only the value of the intercept $\hat{\beta}_0$ must be returned by the local operator. Recall, however, that the LMS minimization procedure supplies the values of the other parameters as well.

The experiments were performed with linear (one dimensional) and planar (two dimensional) models. While the algorithm is unchanged for higher order models we have observed that the additional degree of freedom introduced by a second order fit strongly reduces the smoothing achieved in images corrupted by asymmetric impulse noise. If desired, quadratic fits can be applied to the image presmoothed with linear models. It must be mentioned that for images corrupted only with impulse noise, one-step weighted least squares post-processing is not necessary. The LMS algorithm has already eliminated all the noise with possible exceptions around transitions. The post-processing is of importance, however, when Gaussian noise is also present. This case is discussed later in this section.

In Figure 6a a noiseless, piecewise linear waveform containing 400 samples is shown. When an odd sized processing window is applied to the noiseless signal, the majority of the pixels always belong to the region on which the center of window falls. Therefore we have the following important property:
Any noiseless, piecewise polynomial signal built from segments of degree r or less will remain unchanged after being processed by a smoothing operator with 0.5 breakdown point taking into account models of up to order r; in other words, such a signal is a *root* signal of that operator.
Being built from line segments, the waveform remains unchanged when it is processed by the robust LMS smoothing operator of length 9 employing first order models (Figure 6b).

In Figure 6c fraction $\epsilon = 0.22$ of the samples in the waveform were corrupted with Weibull noise of amplitude $A = 8$. The cumulative distribution of this asymmetric noise process was given in Section 5. Note the severe perceptual distortions of the signal around the transition regions. Several smoothing algorithms were then compared using the same processing window size of 9 samples.

When the least squares procedure is applied to obtain the intercept values, the output is oversmoothed simultaneously with the removal of the impulse noise (Figure 6d). A robust M-estimator employing Hubel's ρ function (Hubel, 1981) gave similar results (Figure 6e). To measure the convergence of the estimation process, the Euclidean distance between the points defined by two consecutive estimate pairs $[\hat{\beta}_0(l), \hat{\beta}_1(l)]$ and $[\hat{\beta}_0(l+1), \hat{\beta}_1(l+1)]$ was employed. The iterations were stopped once this distance became less than 0.05. The poor performance of the M-estimator is caused by the asymmetric nature of the noise and the relative high value of ϵ. While the theoretical breakdown point of this M-estimator is 0.25, the contamination produces regions with much higher ϵ around transitions.

The LMS algorithm with speed-up (Figure 6f) is clearly superior to the previous methods. The original waveform is accurately recovered except at a few transitions where the above discussed artifact appears, and in regions where the contamination exceeds the 0.44 upper bound. Note the recovery of the small step at the right of the waveform. The speed-up is of lesser importance in the one-dimensional case. From the total of 36 possible pairs in the processing window only 19 were considered. Since $\epsilon \leq 0.44$ from Table 1 we obtain the probability of error $Q \leq 0.001$. The probabilistic nature of the speed-up procedure does not degrade the results. The result of the complete LMS algorithm in which all the pairs were considered for the minimization (Figure 6g) does not produce a significantly different result.

In the experiments with two-dimensional data, we have applied the least median square algorithm with speed-up to both synthetic images and natural scenes. The size of the processing window was 5×5. Instead of the 2300 possible triplets only 19 were chosen at random, yielding 120-fold speed-up. Since $p = 3$ the assumed contamination is $\epsilon = 0.4$ for a probability of error $Q = 0.01$ (Table 1).

In Figure 7a the perspective plot of a noiseless 64×64 synthetic image is shown. The image contains several polyhedral objects and a hollow cylinder. When the LMS smoothing algorithm is applied to the noiseless image (Figure 7b) the only degradation is the removal of the pixels at the corners. This effect is present whenever medians are computed over a rectangular processing window. It can be eliminated by selectively computing the medians along principal directions (0, 45, 90 degrees).

In the noisy image fraction $\epsilon = 0.15$ of the samples were corrupted with a Weibull random process having amplitude $A = 75$ (Figure 7c). The output of the LMS smoothing algorithm is given in Figure 7d. Note that while the noise is completely cleaned in the uniform regions distortions may remain around transitions. The processing took 385 seconds of CPU time on a VAX 11/785 computer. Using parallel hardware instead of a serial machine much faster processing times could be achieved.

A 128×128 aerial scene (upper left, Figure 8) was also corrupted with Weibull noise having $A = 255$ (upper right). The employed noise process was equivalent to removing $\epsilon = 0.18$ of the pixels and replacing them

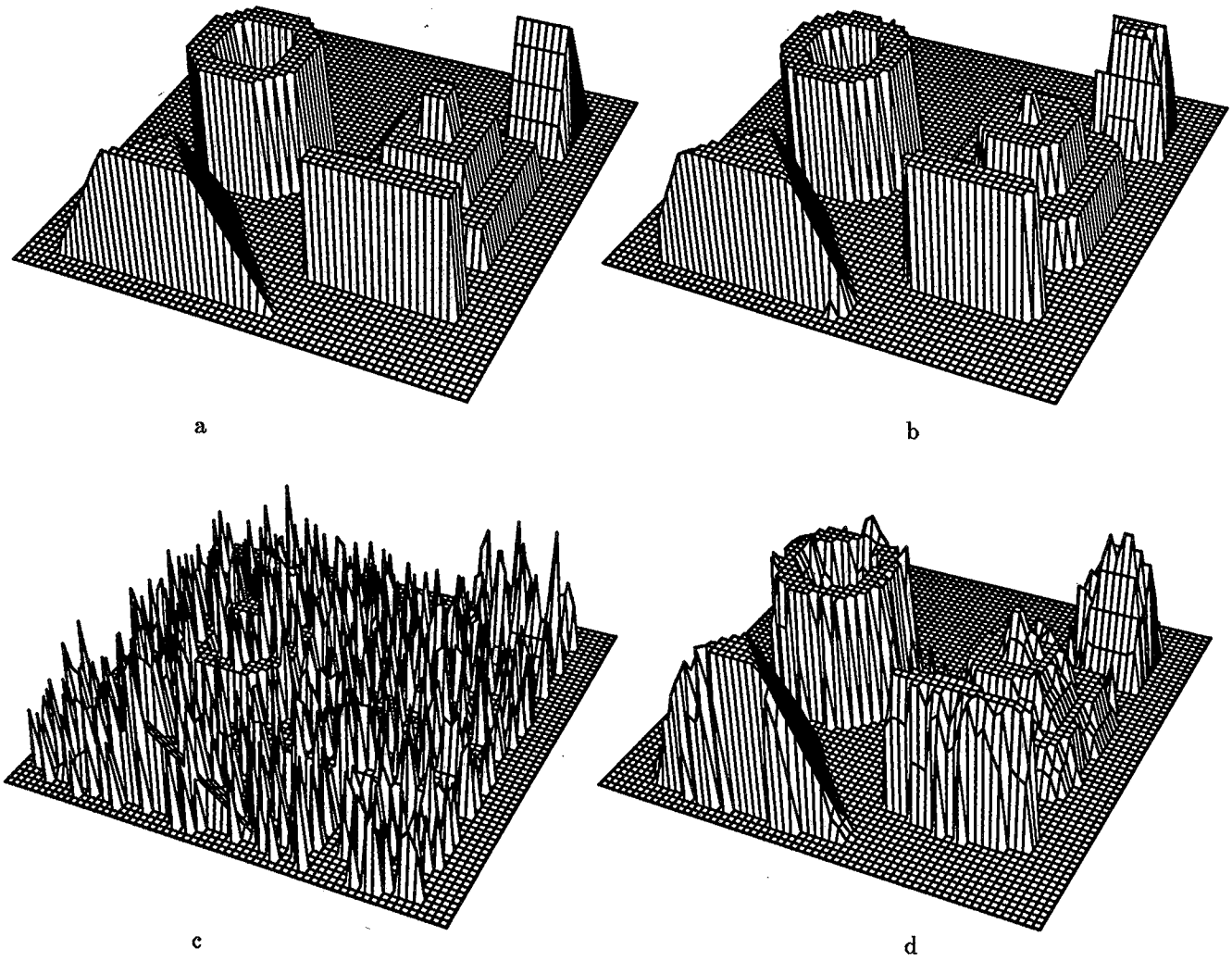

Figure 7. Perspective plots of a synthetic image. a) Noiseless. a) After application of the LMS algorithm. b) Corrupted with Weibull noise. c) After application of the LMS algorithm.

with the maximum gray level value. The image smoothed by the LMS algorithm is shown at the lower left. Most of the details are recovered and thus the algorithm produced a nonlinear interpolation of the missing samples. The processing took 1623 seconds. The result obtained by employing a 5×5 Gaussian weighted smoothing window is given for comparison at the lower right of Figure 8. A serious amount of blurring is introduced, showing the superiority of the robust method for impulse noise removal.

The importance of the one-step weighted least squares post-processing can be observed from Figure 9. In Figure 9a the waveform of Figure 6a is shown corrupted with zero mean Gaussian white noise, standard deviation $\sigma = 2$. Again all the smoothing operators used window size 9. The result of the least squares algorithm is given in Figure 9b. The least squares method is optimum for homogeneous regions and in those regions the results can be regarded as the bound on the achievable performance for the given window size. (Recall that the least squares method fails at transitions, Figure 4.) The result of Hubel's M-estimator (Figure 9c) is similar to the least squares output.

When the LMS algorithm followed by a one-step weighted least squares procedure is employed (Figure 9d) an improvement in the recovery of transitions (edges) and an increase in the noise related fluctuations in the homogeneous regions can be observed. These fluctuations, however, were already attenuated by the post-processing, as a comparison with the output of the LMS algorithm applied alone (Figure 9e) shows.

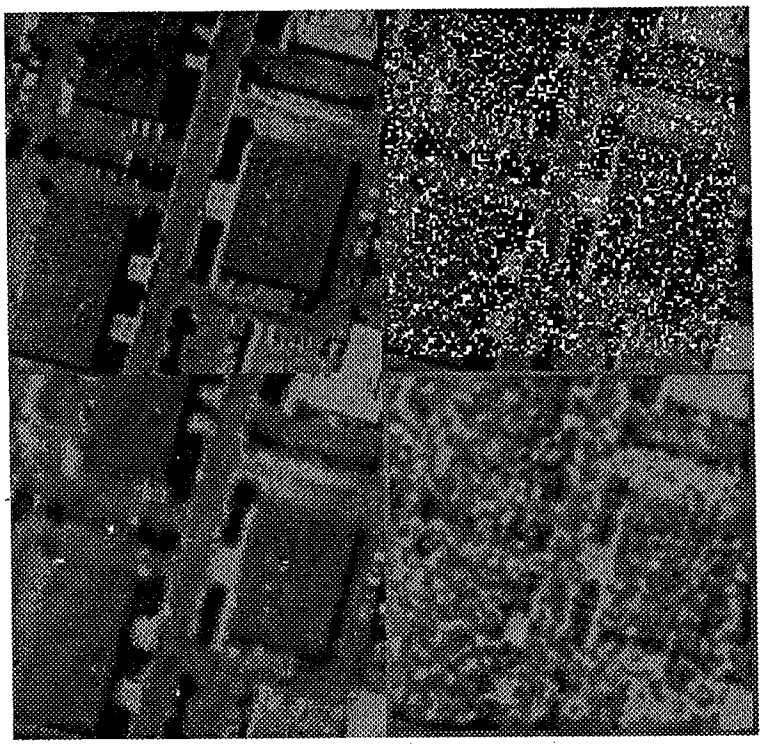

Figure 8. Aerial image. Upper left: Noiseless. Upper right: Noisy with 15% of the the samples removed. Lower left: After application of the LMS algorithm. Lower right: Gaussian smoothing.

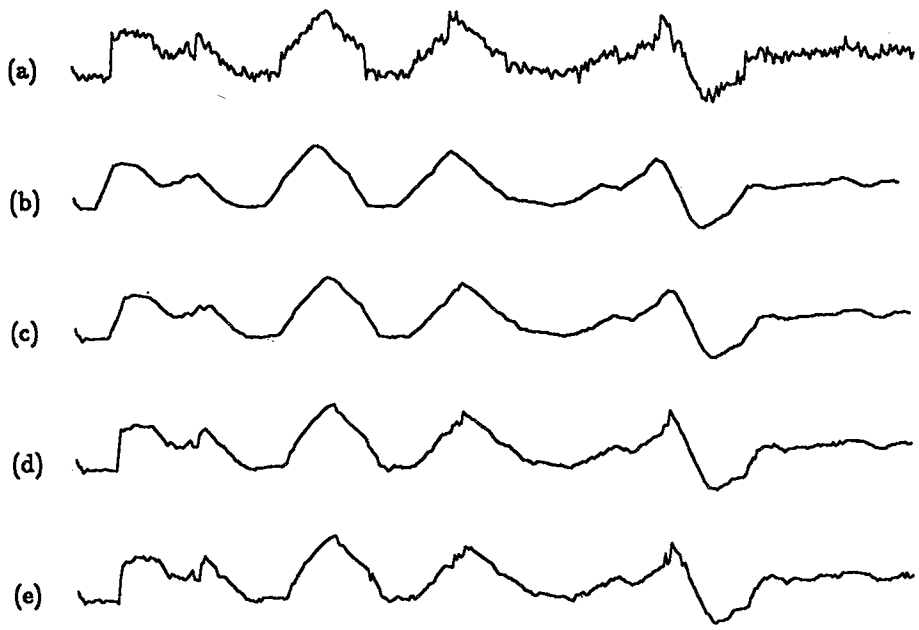

Figure 9. Smoothing of waveform corrupted by Gaussian noise. a) Noisy data, $\sigma = 2$. b) Least squares method. c) Result of Hubel's M-estimator. d) LMS algorithm and one-step weighted least squares. e) LMS algorithm alone.

7. FURTHER DIRECTIONS OF RESEARCH

We have presented a novel robust paradigm based on the least median of squares method for solving computer vision problems. Several directions for further investigation are suggested:

- Design of robust clustering algorithms in which the number of clusters is not known a priori.
- Development of a Hough transform variant in which the analysis of the accumulator is done by the clustering technique described in Section 4. The dependence of the Hough transform's accuracy on the chosen bin size and the effect of digitization of the data will be eliminated.
- Employing heuristics to improve the performance of LMS based algorithms around transitions between homogeneous regions in images. If local connectivity can be established *before* processing the impulse noise can be separated from the samples belonging to the adjacent regions. Successive application of differently sized windows (a multiresolution approach) may also be employed to dichotomize a local region into data and impulse noise.
- To improve the performance of the LMS algorithm's output in the presence of zero mean, symmetric noise processes (e.g. Gaussian), the following approach may be helpful. The input is first smoothed by an LMS procedure which better preserves the transitions. The resulting signal is presegmented into homogeneous regions which then are processed with a robust M-estimator.
- The capacity of the LMS smoothing algorithm to act as a nonlinear interpolation scheme, which preserves transitions better than do linear methods, can be employed in solving computer vision problems where irregularly sampled data is frequent (stereo, optical flow etc.).
- Development of LMS based algorithms for other computer vision problems in which regression analysis is involved.

References

[1] D.F. Andrews (1974): A robust method for multiple linear regression. *Technometrics,* **16,** 523–531.

[2] A.E. Beaton and J.W. Tukey (1974): The fitting of power series, meaning polynomials, illustrated on band-spectroscopic data. *Technometrics,* **16,** 147–185.

[3] P.J. Besl, J.B. Birch and L.T. Watson (1988): Robust window operators. *Proceedings of the Second International Conference on Computer Vision,* December 1988, Tampa, Florida, 591–600.

[4] A.C. Bovik, T.S. Huang and D.C. Munson, Jr. (1987): The effect of median filtering on edge estimation and detection. *IEEE Trans. Pattern Analysis Machine Intelligence* **PAMI-9,** 181–194.

[5] G.W. Brown and A.M. Mood (1951): On median tests for linear hypotheses. *Proceedings of the 2nd Berkeley Symposium on Mathematical Statistics and Probability,* J. Neyman (Ed.), University of California Press, Berkeley and Los Angeles, 159–166.

[6] K.S. Cheng and T.P. Hettmansperger (1983): Weighted least-squares rank estimates. *Commun. Stat.* **A12,** 1069–1086.

[7] R.M. Haralick and H. Joo (1988): 2D-3D pose estimation. *Proceedings of the 9th International Conference on Pattern Recognition,* November 1988, Rome, Italy, 385–391.

[8] S. Heiler (1981): Robust estimates in linear regression—A simulation approach. In *Computational Statistics* H. Büning and P. Naeve (Eds.), De Gruyter, Berlin, 115–136.

[9] P.W. Holland and R.E. Welsch (1977): Robust regression using iteratively reweighted least squares. *Commun. Stat.* **A6,** 813–828.

[10] P.J. Huber (1981): *Robust Statistics.* John Wiley & Sons, N.Y.

[11] L.A. Jaeckel (1972): Estimating regression coefficients by minimizing the dispersion of residuals. *Ann. Math. Stat.* **5,** 1449–1458.

[12] A.K. Jain and R.C. Dubes (1988): *Algorithms for Clustering Data,* Prentice-Hall, Englewood Cliffs, NJ.

[13] J. Jolion and A. Rosenfeld (1988): Cluster Detection in Background Noise, CAR-TR-363, Center for Automation Research, University of Maryland, College Park, June 1988. To appear in *Pattern Recognition Letters.*

[14] I.M. Johnstone and P.F. Velleman (1985): The resistant line and related regression methods. *J. Am. Stat. Assoc.* **80,** 1041–1059.

[15] B. Kamgar-Parsi, B. Kamgar-Parsi and N.S. Netanyahu (1989): A nonparametric method for fitting a straight line to a noisy image. To appear in *IEEE Trans. Pattern Analysis Mach. Intell.* See also CAR-TR-315, Center for Automation Research, University of Maryland, College Park.

[16] R.L. Kashyap and Kie-Bum Eom (1988): Robust image models and their applications. In *Advances in Electronics and Electron Physics, Vol. 70,* Academic Press, San Diego, CA, 79–157.

[17] G. Li (1985): Robust regression. In D.C. Hoaglin, F. Mosteller and J.W. Tukey (Eds.), *Exploring Data Tables, Trends and Shapes.* John Wiley & Sons, 281–343.

[18] F. Mosteller and J.W. Tukey (1977): *Data Analysis and Regression.* Addison-Wesley, Reading, MA.

[19] P.J. Rousseeuw (1984): Least median of squares regression. *J. Amer. Stat. Assoc.* **79,** 871–880.

[20] P.J. Rousseeuw and A.M. Leroy (1987): *Robust Regression & Outlier Detection.* John Wiley & Sons,

[21] A.F. Siegel (1982): Robust regression using repeated medians. *Biometrika* **69,** 242–244.

[22] J.M. Steele and W.L. Steiger (1986): Algorithms and complexity for least median of squares regression. *Discrete Applied Math.* **14,** 93–100.

[23] H. Theil (1950): A rank-invariant method of linear and polynomial regression analysis (parts 1–3), *Ned. Akad. Wetensch. Proc. Series A* **53,** 386–392, 521–525, 1397–1412.

[24] V.J. Yohai and R.H. Zamar (1988): High breakdown point estimates of regression by means of the minimization of an efficient scale. *J. Amer. Stat. Assoc.* **83,** 406–413.