The Surface Density Criterion and Its Application to Linear/Circular Boundary Detection and Planar/Spherical Surface Approximation

Raghu Krishnapuram, Olfa Nasraoui, and Hichem Frigui
Department of Electrical and Computer Engineering
University of Missouri
Columbia, MO 65211, USA

***Abstract*—In this paper, we introduce a new validity measure called surface density, and develop an unsupervised shell clustering algorithm for the detection of circles in intensity images and for spherical approximation of surfaces in range images. The algorithm can be used when the number of clusters is unknown. Unlike the existing algorithms, the new algorithms can handle the degenerate case of lines and planes, and they are more robust.**

I. INTRODUCTION

Shell clustering techniques [2,4,8-11] have been shown to be effective for boundary detection and surface approximation in computer vision. In this paper, we are mainly concerned with the detection of circular shapes in intensity images and spherical surfaces in range images. One problem with the existing shell clustering algorithms is that they cannot handle the degenerate case of lines/planes, and moreover, they can give poor fits if the circles (spheres) have very large radii [11]. They also do not perform well in noisy situations. We propose a new plano-spherical shell clustering algorithm to overcome these drawbacks. We also show that the existing cluster validity measures are not suitable for the type of shell clusters that one encounters in computer vision problems, since they cannot always distinguish between good clusters and spurious clusters. We introduce a new validity measure called surface density to address this problem.

II. THE FUZZY C PLANO-SPHERICAL SHELLS ALGORITHM

Let an n-dimensional spherical shell β_i be represented by $\boldsymbol{p}_i^{\mathrm{T}}\boldsymbol{q} = 0$; $\boldsymbol{p}_i = [p_{i1}, p_{i2}, \ldots p_{i(n+1)}, p_{i(n+2)}]^{\mathrm{T}}$, and

$$\boldsymbol{q} = [(x_1^2+x_2^2+x_3^2+\ldots+x_n^2), x_1, \ldots, x_n, 1]^{\mathrm{T}}. \quad (1)$$

The approximate (first order) distance [12] of a point $\boldsymbol{x}_j$ from the shell is given by:

$$d_{\mathrm{A}ij}^2 = d_{\mathrm{A}}^2(\boldsymbol{x}_j, b_i) = \frac{d_{ij}^2}{|\nabla d_{ij}|^2} = \frac{d_{ij}^2}{\boldsymbol{p}_i^{\mathrm{T}} D(\boldsymbol{q}_j) D(\boldsymbol{q}_j)^{\mathrm{T}} \boldsymbol{p}_i} \quad (2)$$

where the matrix $D(\boldsymbol{q}_j)$ is the Jacobian of $\boldsymbol{q}$ evaluated at $\boldsymbol{x}_j$. We use the approximate distance rather than the exact (geometric) distance because the exact distance leads to a computationally expensive updating equation for the shell parameters. In order to obtain a C-partition of a given data set $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\}$ with $d^2{}_{\mathrm{A}ij}$ as the distance measure, the objective function to be minimized is:

$$J_{\mathrm{A}}(B,U,X) = \sum_{i=1}^{C}\sum_{j=1}^{N} (u_{ij})^m d_{\mathrm{A}ij}^2$$
$$= \sum_{i=1}^{C}\sum_{j=1}^{N} (u_{ij})^m \frac{\boldsymbol{p}_i^{\mathrm{T}} \boldsymbol{Q}_j \boldsymbol{p}_i}{\boldsymbol{p}_i^{\mathrm{T}} D(\boldsymbol{q}_j) D(\boldsymbol{q}_j)^{\mathrm{T}} \boldsymbol{p}_i} \quad (3)$$

In (3), $\boldsymbol{B} = (\beta_1, \ldots \beta_C)$, $m \in [1,\infty)$ is a weighting exponent called the fuzzifier, u_{ij} is the membership of $\boldsymbol{x}_j$ in cluster β_i and $\boldsymbol{U} = [u_{ij}]$ is the $C \times N$ fuzzy C-partition matrix [1] such that:

$$u_{ij} \in [0,1] \text{ for all } i \text{ and } j, \quad 0 < \sum_{j=1}^{N} u_{ij} < N \text{ for all } i. \text{ and}$$
$$\sum_{i=1}^{C} u_{ij} = 1 \text{ for all } j. \quad (4)$$

While minimizing (3) with respect to $\boldsymbol{p}_i$, a suitable constraint needs to be chosen to avoid the trivial solution. The previous algorithms use the implicit constraint that $p_{i1} = 1$, which precludes linear solutions. To overcome this problem, one could generalize Pratt and Taubin's constraint [11,12] to the fuzzy case and to fit C curves simultaneously. This is achieved by choosing the constraint to be

$$\boldsymbol{p}_i^{\mathrm{T}}\Big[\sum_{j=1}^{N} (u_{ij})^m [D(\boldsymbol{q}_j) D(\boldsymbol{q}_j)^{\mathrm{T}}]\Big]\boldsymbol{p}_i = \sum_{j=1}^{N} (u_{ij})^m, \text{ or}$$
$$\boldsymbol{p}_i^{\mathrm{T}} \boldsymbol{G}_i \boldsymbol{p}_i = N_i, \quad \text{for } i = 1, \ldots, C, \quad (5)$$

where

$$\boldsymbol{G}_i = \sum_{j=1}^{N} (u_{ij})^m [D(\boldsymbol{q}_j) D(\boldsymbol{q}_j)^{\mathrm{T}}], \text{ and } N_i = \sum_{j=1}^{N} (u_{ij})^m. \quad (6)$$

Minimizing (3) subject to (5) yields a cumbersome equation which cannot be solved for $\boldsymbol{p}_i$ explicitly. In order to avoid an iterative solution, we assume: i) the magnitude of the gradient of each curve β_i at each point $\boldsymbol{x}_j$ is approximately constant, i.e., $\boldsymbol{p}_i^{\mathrm{T}} D(\boldsymbol{q}_j) D(\boldsymbol{q}_j)^{\mathrm{T}} \boldsymbol{p}_i \approx$ constant for all $\boldsymbol{x}_j \in \beta_i$, and ii) all data points are reasonably close to some cluster, i.e., the u_{ij} are close to being hard. It can be shown that assumption (i) is always valid for the case of lines (planes) and it is valid for circles (spheres and cylinders) when $\boldsymbol{q}_j$ corresponds to a point on the curve (surface). Assumption (ii) is reasonable when the data is not highly scattered and when there is little noise. It is also reasonable for the noisy case if we use possibilistic memberships, which will be discussed in Section III. From these assumptions and (6), it follows that $\boldsymbol{p}_i^{\mathrm{T}} D(\boldsymbol{q}_j) D(\boldsymbol{q}_j)^{\mathrm{T}} \boldsymbol{p}_i \approx 1$. Hence, we will obtain approximately

0-7803-0614-7/93$03.00 ©1993IEEE

the same solution if we ignore the denominator in (3). Using Lagrange multipliers, it is easy to show that the parameter vectors p_i can be obtained by the following generalized eigenvector problem:

$$F_i p_i = \lambda_i G_i p_i, \text{ where } F_i = \sum_{j=1}^{N} (u_{ij})^m Q_j. \quad (7)$$

Minimization of the objective function with respect to U subject (4) gives

$$u_{ij} = \begin{cases} \dfrac{1}{\sum_{k=1}^{c} \left(\dfrac{d_{Aij}}{d_{Aik}} \right)^{\frac{2}{m-1}}} & \text{if } I_j = \emptyset \\ 0 \quad i \notin I_j & \text{if } I_j \neq \emptyset \\ \sum_{i \in I_j} u_{ij} & \text{if } I_j \neq \emptyset \end{cases} \quad (8)$$

where $I_j = \{ i \mid d_{Aij}^2 = 0 \}$.

THE FUZZY C-PLANO-SPHERICAL SHELLS (FCPSS) ALGORITHM

Fix the number of clusters C; fix m, $1 < m < \infty$;
Set iteration counter $l = 1$, and initialize the fuzzy C-partition $U^{(0)}$;
Repeat
  Compute the matrices $F_i^{(l)}$, and $G_i^{(l)}$ for each cluster β_i using (6) and (7);
  Compute $p_i^{(l)}$ for each cluster β_i by solving (7);
  Update $U^{(l)}$ using (8);
  Increment l;
Until ($\| U^{(l-1)} - U^{(l)} \| < \varepsilon$).

III. THE POSSIBILISTIC C PLANO-SPHERICAL SHELLS ALGORITHM

Most C-means type clustering algorithms are adversely affected by noise, because they are based on the minimization of the sum squared errors. This problem can be overcome if one could assign very low memberships in all clusters to noise points, since they are unrepresentative of any cluster. However, this is not possible with the constraint in (4). Moreover, the relative memberships generated by the constraint introduce a bias in the estimation of the prototype parameters. This situation can be improved by the possibilistic formulation of the clustering problem that has been proposed recently [7]. This gives the following equation for updating the memberships.

$$u_{ij} = \left[1 + \left(\frac{d_{Aij}^2}{\eta_i} \right)^{\frac{1}{m-1}} \right]^{-1}, \quad (9)$$

where the "bandwidths" η_i are suitable positive numbers. The prototype parameter update equations remain the same. In the case of shell clustering algorithms, the values for η_i may be set to the square of the expected thickness of the shells. This is desirable if one needs to compute cluster validity, as will be discussed in Section V. We call the resulting algorithm the Possibilistic C Plano-Spherical Shells (PCPSS) algorithm.

IV. PARTITION AND CLUSTER VALIDITY

Clustering algorithms have one major drawback in common, which is the difficulty in determining the number of subgroups C present in the data set. Traditionally, the optimum number of clusters is determined by evaluating a certain global validity measure of the C-partition for a range of C values, and picking the value of C that optimizes the validity measure [1,3,5]. However, this is tedious and is not always guaranteed to work [3,6,9], since the performance measure may not be optimized at the "correct" value of C. This may happen when the resulting partition for the correct value of C happens to correspond to a local minimum rather than a global one. It may also happen when one of the clusters is split into two or more clusters, and when spurious clusters are formed. These situations occur frequently with data sets that are complex or data sets that contain a large number of clusters. Another approach to determine the number of clusters C is to perform progressive clustering [6,9]. In this approach, after the clustering algorithm converges with an overspecified number of clusters, spurious

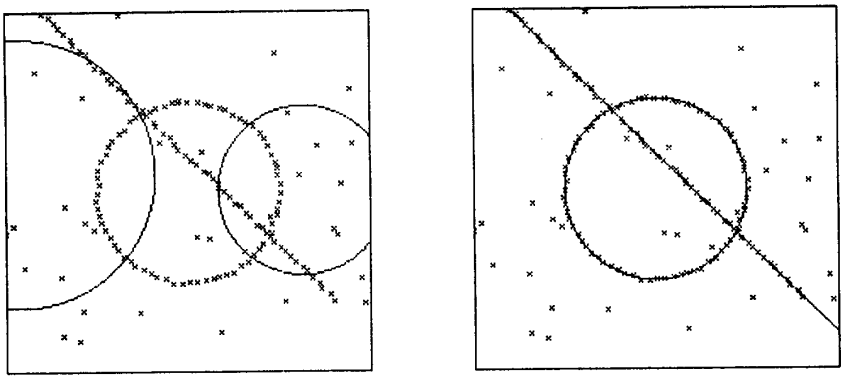

Fig. 1 (Left) Results of FCPSS on a data set consisting of a scattered line and a circle contaminated with noise. The prototypes are superimposed on the data set. (Right) Result of PCPSS algorithm. The fixed value of η_i was 2.

clusters are eliminated, compatible clusters are merged, "good" clusters are identified, and points belonging to these clusters are removed from the data set. Then clustering is performed again with the reduced number of clusters and data points. This process is repeated until no more good clusters can be removed. However, for this approach to work, we need a validity criterion to evaluate the goodness of an individual cluster.

Several validity measures can be defined for shell clusters [3,9]. This is done by first defining the distance vector from a feature point to a shell prototype as $\Delta_{ij} = x_j - z^i_j$, where z^i_j is the closest point on the prototype curve (surface) β_i to x_j. In the case of spherical clusters,

$\Delta_{ij} = (x_j - c_i) - r_i (x_j - c_i) \| x_j - c_i \|^{-1}$.

The fuzzy average shell thickness may be defined to be

$$T_i = (1/N_i) \sum_{j=1}^{N} (u_{ij})^m \| \Delta_{ij} \|^2, \quad (10)$$

where N_i is as in (7). The fuzzy shell covariance matrix is defined by

$$C_{Si} = (1/N_i) \sum_{j=1}^{N} (u_{ij})^m \Delta_{ij} \Delta_{ij}^T. \quad (11)$$

Based on (11), the shell hypervolume and the shell density of a cluster may be defined as

$$V_{Si} = \sqrt{\det(C_{Si})} \text{ and } D_{Si} = S_i / V_i, \quad (12)$$

where S_i is the sum of close members of shell β_i given by

$S_i = \sum_j u_{ij}$ sum over j such that $\Delta_{ij}^T C_{Si}^{-1} \Delta_{ij} < 1$.

Fig. 2(a) shows an example to illustrate the problems with the validity measures mentioned above. The data set consists of three clusters: a large circle, and a half-circle having the same radius, and a small circle. The individual cluster validities obtained after correctly clustering this data set using the PCPSS algorithm with $C = 3$ are listed in Table I. As seen, the shell volume and density vary very much depending on the partiality (incompleteness) and size of the cluster. Fig. 2(c) shows another data set consisting of two elliptical clusters. When this data set was clustered with the Possibilistic Modified C Quadric Shells (PMCQS) algorithm [10] with an overspecified number of clusters $C = 3$, the two good elliptical clusters were found, but a spurious cluster was also formed. The validity measures for this data set are listed in Table II. In this case the shell volume is low for the spurious cluster because it contains very few points, giving the wrong indication that the spurious cluster is better than the good ones. Also the shell density and thickness are almost the same for the spurious cluster and for the good clusters.

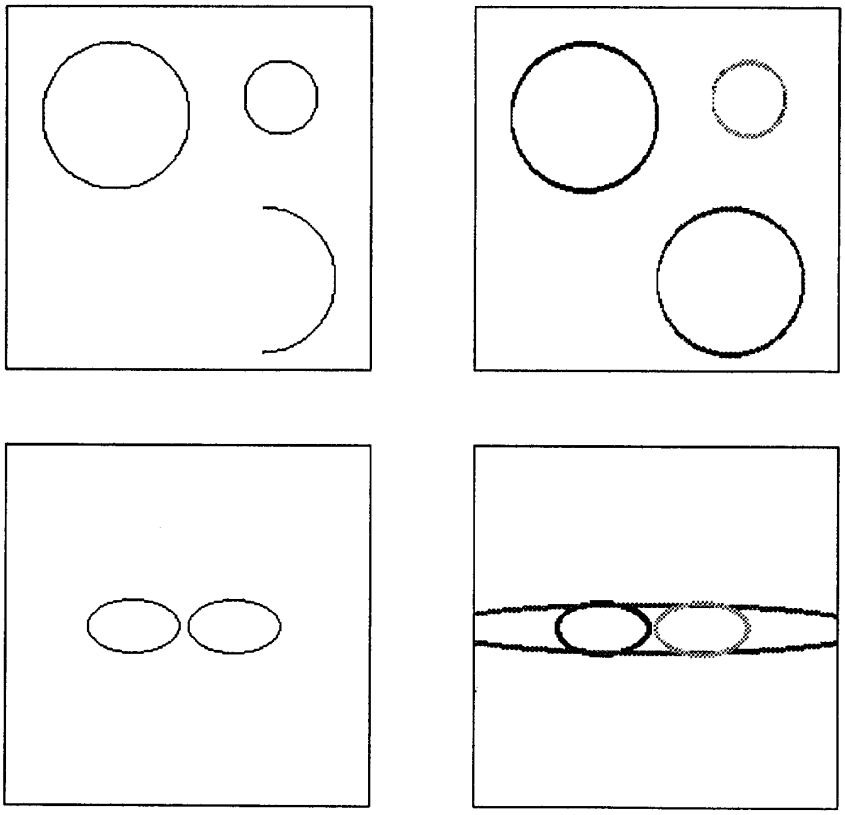

Fig. 2 (Clockwise from top left) (a) A data set with circles of various sizes and partiality, (b) prototypes found by the PCPSS superimposed on the data set, (c) a data set with two ellipses, and (d) Prototypes found by the PMCQS.

TABLE I
VALIDITIES FOR THE CLUSTERS IN FIG. 2(b)

Cluster	Shell Vol.	Density	Thickness
large circle	0.83	260.0	0.006
small circle	3.05	35.8	0.012
half circle	2.4	45.4	0.006

TABLE II
VALIDITIES FOR THE CLUSTERS IN FIG. 2(d)

Cluster	Shell Vol.	Density	Thickness
left ell.	0.66	112.3	2.08
right ell.	0.66	112.3	2.08
spur. ell.	0.17	92.0	2.5

Thus, from a boundary and surface fitting point of view, the above cluster validity measures suffer from many drawbacks. They also lack a standard value against which one can compare the validity of a particular cluster. Spurious clusters often contain only a few points and hence the fits can be very good. For partial shells and small shells, since the sum of central members is relatively small, the shell density goes down. All this can happen even though the image contains no noise and the edges have uniform thickness. A more appropriate measure of validity for such subspace clusters needs to relate to the density of the cluster as viewed in the subspace. We now introduce a new validity measure that overcomes these disadvantages.

V. THE SURFACE DENSITY CRITERION

Our validity measure is related to the shell density measure introduced in the previous section, except that instead of measuring the number of points per unit shell volume, we measure the number of points per unit curve length or surface area. To do this, we need to be able to estimate the curve length or the surface area without assuming connectivity etc. of points belonging to a shell. The curve length can be estimated from the cluster parameters provided the cluster is complete. For example, the curve length of a circular cluster is equal to $2\pi r$. However, this method will not work for partial curves. As an estimate of the arc length spanned by an incomplete circular shell cluster, one may use $2\pi r_{eff}$, where r_{eff} is the radius of a circle which is in some sense equivalent to the partial circle. We refer to the radius of the equivalent circle as the effective radius r_{eff}.

Since the covariance matrix C (not the shell covariance matrix) of a cluster measures the "span" of the cluster, C is an obvious choice to be used in the definition of equivalence. For a complete hypersphere with radius r, it can be shown that the covariance matrix is given by $(r^2/n)I_n$, where I_n is the identity matrix of size $n \times n$. This suggests that the effective radius should be defined as

$$r_{eff} = \sqrt{\mathrm{Tr}\, C}. \tag{13}$$

If the original cluster of radius r is partial, then r_{eff} and r will not be the same. It is easy to show that the equivalent hypersphere with radius r_{eff} has the same second moment as the shell cluster under consideration. Thus, a geometric interpretation of the definition of equivalence is established.

In the 2-D case, the surface density δ is defined as the number of points per unit arc length of cluster i.

$$\delta_i = \frac{S_i}{2\pi r_{eff}};\ S_i = \sum_j m_{ij} \tag{14}$$

In (14), the sum is over j such that $\| \Delta_{ij} \| < \tau_{\max}$.where$\tau_{\max}$ is the expected thickness of the shell. A good value for $\tau_{\max}$ in our applications was found to be $\tau_{\max} = \sqrt{\eta_i} = \sqrt{2}$. In the three-dimensional case, we define surface density as follows:

$$\delta_i = \frac{S_i}{4\pi r_{eff}^2}. \tag{15}$$

A. Surface Density For Circles

Consider a circular arc parametrized by: $x = [x, y]^T = [r\cos\theta, r\sin\theta]^T$. Let the arc extend from $\theta = \phi_1$ to $\theta = \phi_2$. In this case,

$$C = \frac{1}{L_\phi}\int_{\phi_1}^{\phi_2} xx^T\, dl - mm^T, \text{ where } m = [m_x, m_y]^T, \text{ with}$$

$$m_x = \frac{1}{L_\phi}\int_{\phi_1}^{\phi_2} x\, r d\theta \text{ and } m_y = \frac{1}{L_f}\int_{\phi_1}^{\phi_2} y\, r d\theta,$$

$dl = r\, d\theta$, $r = \sqrt{x^2 + y^2}$, L_ϕ is the arc length $r(\phi_1 - \phi_2) = r\phi$. Using these substitutions, we obtain the theoretical value of δ (assuming the arc is perfect and continuous) to be

$$d = \frac{L_\phi}{2\pi r_{\text{eff}}} = \frac{L_\phi}{2\pi\sqrt{\mathrm{Tr}\, C}} = \frac{\phi}{2\pi\sqrt{1 - 4f^2\sin^2(\phi/2)}}. \tag{16}$$

The values of r_{eff}/r and δ are tabulated in Table III for various circular cluster configurations. Lines are considered as circles with an infinite radius. Hence $r_{eff}/r = 0$ for lines. The last column of this table will be explained in Section V.C. From the table, we see that for a complete circle the surface density criterion is independent of the size (radius), and reaches its upper bound of 1. This sets an absolute value of δ for circles (=1) against which to compare the validities. However, δ drops as the circle becomes more partial. This problem will be addressed and corrected in Section V.C, when δ is compensated for partiality.

TABLE III
VALUES OF SURFACE DENSITY FOR SOME TYPICAL CIRCULAR CLUSTERS AND FOR A LINE.

Cluster type	$\frac{r_{eff}}{r}$	$d_{\text{theoretical}}$	$f = \frac{1}{d_{\text{theoretical}}}$
complete circle	1.00	1.00	1.00
semicircle	0.77	0.65	1.54
quarter circle	0.44	0.57	1.74
line	0.00	0.55	1.81

B. Surface Density For Ellipses and Ellipsoids

Results similar to those derived in Section V.A can be derived for ellipses. However, in this case, the expression for Tr C can be integrated only numerically. In the 3-D case, the expression for Tr C is integrable. The integration is rather tedious, and we present only certain special cases.

TABLE IV
VALUES OF SURFACE DENSITY FOR SOME TYPICAL ELLIPTICAL AND ELLIPSOIDAL SHELL CLUSTERS.

Ellipse Type	$\frac{a}{b}$	Surface Density	Shell Type	$\frac{R_3}{R_1}$	$\frac{R_2}{R_1}$	Surface Density
complete	2	0.93	complete sphere	1	1	1.00
"	10	0.80	complete ellipsoid	1	2	0.83
half	2	0.60	half sphere	1	1	0.66
"	10	0.57	half ellipsoid	1	2	1.19
quarter	2	0.59	quarter sphere	1	1	0.50
"	10	0.69	quarter ellipsoid	1	2	0.78

Table IV summarizes some of these results. In this table, a and b represent the major and minor semiaxes of the ellipse, and R_1, R_2, and R_3 represent the three radii of the ellipsoid. It can be seen that the variation in the surface density is surprisingly small for large variations in the partiality and elongation of the ellipses and ellipsoids.

C. Compensation factors for Surface Density

Although the relatively small variation of surface density does not pose problems in practice when one wants to discriminate between good and bad clusters, it is still possible to make it invariant to partiality, and to give it a standard value of 1. Here we discuss how this is achieved for the case of circles and lines. It can be seen from Table III that r_{eff}/r can be used as a measure of partiality for circular clusters. Our goal is to find a scaling factor f which when multiplied by δ, will normalize it to the standard value (=1). Thus, for the three common circular arcs and linear segments listed in Table III, the computed δ should be scaled by the corresponding f. For other partial curves, the computed δ may be scaled by a factor f which is an interpolated function given below.

$$f = \begin{cases} 1.74 - 0.172\,((r_{eff}/r) - 0.44) & \text{for } (r_{eff}/r) \in [0\ 0.44] \\ 1.54 - 0.594((r_{eff}/r) - 0.77) & \text{for } (r_{eff}/r) \in [0.44\ 0.77] \\ 1.0 - 2.34((r_{eff}/r) - 1.0) & \text{for } (r_{eff}/r) \in [0.77\ 1.0]. \end{cases}$$

Compensation for ellipses is a little more involved and is not discussed here. A similar method can be used in 3-D. However, in this case, one also needs to take foreshortening into account.

D. Examples of Validities

The computed surface densities for the circular clusters detected in Fig. 2(b) and for the elliptical clusters detected in Fig. 2(d) are shown in Tables V and VI. It can be seen that the value of the compensated δ is close to 1 for the good clusters, while being low for the spurious cluster. It is also relatively invariant to size and partiality of clusters.

We should note that using the possibilistic versions of our shell clustering algorithms yields memberships that are better suited for computing individual cluster validities. Fuzzy membership values can cause an underestimation of the sum of central members, particularly in cases where some of the points are shared by several clusters. Possibilistic memberships have the advantage of allowing each shared point of a shell cluster to contribute with a membership value close to 1.

TABLE V
SURFACE DENSITIES FOR THE CLUSTERS IN FIG. 2(b)

Cluster	Surface Density	Comp. Density
large circle	0.88	1.04
small circle	0.89	1.04
half circle	0.57	0.97

TABLE VI
SURFACE DENSITIES FOR THE CLUSTERS IN FIG. 2(d)

Cluster	surface density	comp. density
left ellipse	0.92	0.93
right ellipse	0.92	0.93
spurious ellipse	0.50	0.32

VI. THE UNSUPERVISED POSSIBILISTIC PLANO-SPHERICAL SHELLS ALGORITHM

We used the progressive clustering idea described in Section IV to find the optimal number of a mixture of linear/planar and circular/spherical clusters in a data set. The resulting algorithm is called the Unsupervised Possibilistic Plano-Spherical Shells (UPPSS) algorithm. In the elimination step, cluster β_i was considered spurious if $\delta_i < \delta_{low} = 0.40$. In the merging step, two spherical shell clusters β_i and β_j with centers c_i and c_j are considered compatible if $\|c_i - c_j\| < \varepsilon_1$ and $|r_i - r_j| < \varepsilon_2$. One the other hand, two linear clusters β_i and β_j are considered compatible if they satisfy the conditions used in the Compatible Cluster Merging (CCM) algorithm [6]. After the merging has been performed a cluster β_i was identified as a "good" cluster if $\delta_i > \delta_{high} = 0.85$. In the removing step, a data point x_k is removed temporarily if x_k if its possibilistic membership in a good cluster exceeds 0.5 or if it has its highest possibilistic membership in a good cluster.

Due to space limitation, here we show only one example of the results obtained by the UPPSS algorithm. Fig. 3(a) shows a 200×200 image of some parts with linear and circular boundaries of different sizes. Fig. 3(b) shows the edge image, obtained by applying the Sobel operator and then thinning the resulting edges to reduce the number of pixels in the data set and to ensure more or less uniform edge thickness. The thinned edge image has about 1500 points. The UPCPSS algorithm when started with C=20 succeeds in detecting all clusters correctly, and the prototypes (each one in a different shade of gray) superimposed on the edge image are shown in Fig. 3(c).

VII. CONCLUSIONS

In this paper, we have presented a new algorithm for linear/circular boundary detection and planar/spherical surface approximation. Unlike the previous algorithms it can handle the degenerate case of linear/planar clusters. The possibilistic version of the algorithm makes it robust in the presence of noise. The proposed individual cluster validity measure (surface density) is relatively invariant to size and partiality of clusters. This measure was used in developing an unsupervised algorithm. The proposed algorithm provides a very good initialization to more complex fits.

REFERENCES

[1] J. C. Bezdek,*Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.

[2] R. N. Davé, "Fuzzy Shell-Clustering and Application to Circle Detection in Digital Images," *International J. of General Systems*, vol. 16, pp. 343-355, 1990.

[3] R. N. Davé, "New measures for evaluating fuzzy partitions induced through C-shells clustering," *Proc. of the SPIE Conf. on Int. Robots and Computer Vision X: Algs. and Techniques*, Boston, Nov. 1991, pp. 406-414.

[4] R. N. Davé and K. Bhaswan, "Adaptive Fuzzy C-Shells Clustering and Detection of Ellipses," *IEEE Trans. on Neural Networks*, vol. 3, no. 5, Sept. 1992, pp. 643-652.

[5] I. Gath and A. B. Geva, "Unsup. Optimal Fuzzy Clustering," *IEEE Trans. on PAMI.*, vol. 11, no. 7, pp. 773-781, 1989.

[6] R. Krishnapuram and C.-P. Freg, "Fitting unknown number of lines and planes to image data through compatible cluster merging," *Pattern Recog.*, vol. 25, no. 4, 1992, pp. 385-400.

[7] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," to appear in *IEEE Trans. on Fuzzy Syst.*, 1993.

[8] R. Krishnapuram, H. Frigui, and O. Nasraoui, " New fuzzy shell clustering algorithms for boundary detection and pattern recognition," *Proc. of SPIE Conf. on Intell .Robots and Comp. Vision X*: Algs. and Techniques, Boston, Nov. 91, pp 458-465.

[9] R. Krishnapuram, O. Nasraoui, and H. Frigui, "The fuzzy C spherical shells algorithm: A new approach," *IEEE Trans. on Neural Networks*, vol. 3, no. 5, Sept. 1992, 663-671.

[10] R. Krishnapuram, H. Frigui, and O. Nasraoui, "A fuzzy clustering algorithm to detect planar and quadric shapes," *Proc. of the North American Fuzzy Information Society Workshop*, Puerto Vallarta, Dec. 1992., pp. 59-68

[11] V. Pratt, "Direct least squares fitting of algebraic surfaces," *Comp. Graphics*, vol. 21, no. 4, pp.145-152, 1987.

[12] G. Taubin, "Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations," *IEEE Trans. PAMI*, vol. 13, no. 11, pp.1115-38, 1991.

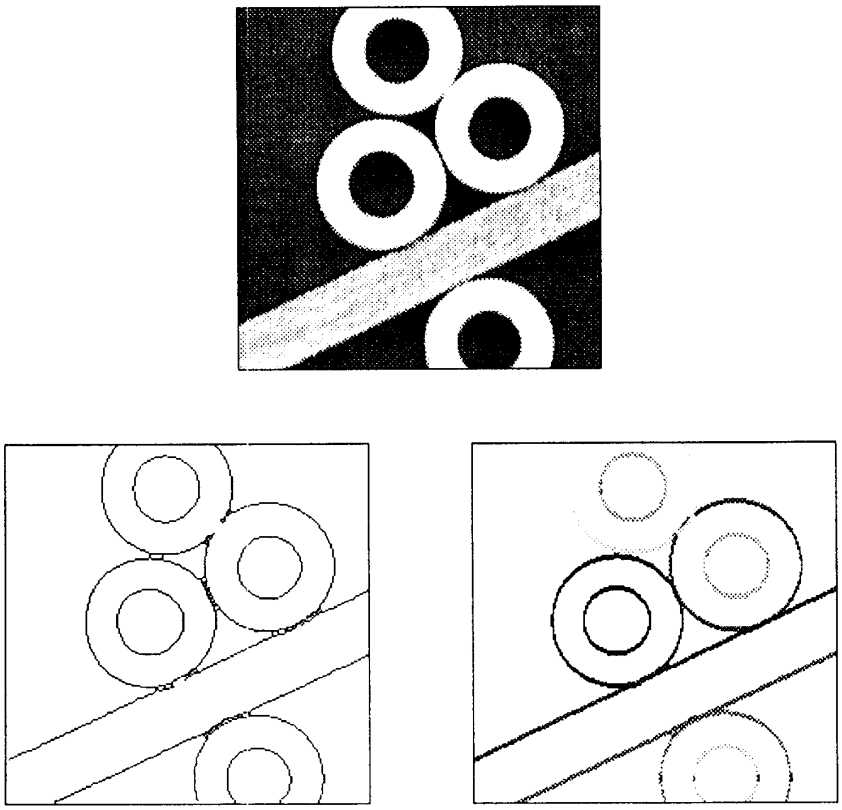

Fig. 3 Results of the UPCPSS algorithm. (Top left) Original image. (Top right) Edge image. (Bottom) The prototypes superimposed on the edge image. Each prototype is shown in a different shade of gray.