

Ellipse Detection with Hard c -Regression Models and Random Initializations

Hidetomo Ichihashi, Li Chieu Lam, Katsuhiro Honda, Akira Notsu
Department of Computer Science and Intelligent Systems
Osaka Prefecture University
Sakai, Osaka
Email: ichi@cs.osakafu-u.ac.jp

Abstract—Shell clustering methods partition data sets into several shell-shape clusters by extracting local circles or ellipses as prototypes of clusters. This paper proposes hard c -regression models (HCRMs) for shell clustering. The procedure is a defuzzified switching regression models. HCRMs successfully detect ellipses by using random initializations. We report the performance using 20 data sets each of which consists of two ellipses. The detection time on average is 14 milliseconds on DELL PRECISION T5400 3.16GHz.

Index Terms—ellipse detection, clustering, switching regression, random initializations.

I. INTRODUCTION

Figs. 1-2 illustrates an application of ellipse detection, where a robot grasps a saucer placed upside-down. First, Sobel edge detection method is applied on the gray scale image. Then the coordinate of pixels on the edges are picked by thresholding the gray scale values and thinned by sampling every 10 points. The data set consisted of the coordinate of points on edge is used for ellipse detection. The robot can know the relative position of the saucer precisely and can grasp it correctly, since the mathematical formula of the ellipses are obtained.

A standard method of detecting ellipses is the Hough transform (HT). The Hough transform extracts geometrical parameters of an image and maps them into the Hough domain. Standard HT, however, is not really used for fast ellipse detection since it requires a lot of computational processing. An ellipse has five parameters and requires intensive memory.

The 5-dimensional parameter space is decomposed into several subspaces of fewer dimensions via parallel edge points in [3]. A multistage HT approach using the polar and pole definition of ellipses has been proposed in [4]. Ellipses were extracted by relating local geometric features of a parametric representation to the local features in an image in [5]. In [6] a global geometric symmetry method was utilized to locate the centers of ellipses and circles and then used the accumulative concept of the Hough transform to extract all ellipses and circles in an image. Another HT variant is the Randomized Hough Transform (RHT) [7] which is well-suited to ellipse detection due to its lessened usage of memory and fast processing characteristic as proposed in [8].

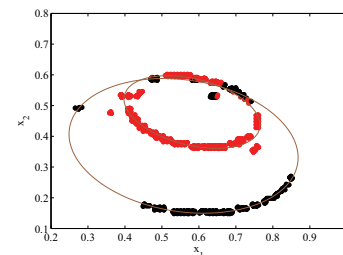
Some non-Hough transform methods were also proposed for detecting ellipses. Dave [9] used the fuzzy c -shells to detect circles and ellipses. Shell clustering methods divide



A saucer placed upside-down



Sobel edge detection



Ellipse detection after thresholding intensity and thinning the edges by sampling

Fig. 1. Detecting saucer placed upside-down

data into clusters, each one representing a circle or an ellipse. Fuzzy c -Shells (FCS) [9] and Adaptive Fuzzy c -Shells (AFCS) [10] are represented by a prototype of nonlinear function; FCS is effective for dividing data into circle clusters and into ellipse clusters. However for each iteration, it is necessary to solve a set of nonlinear equations, so there is a problem in terms of complexity. R. Krishnapuram's Fuzzy c -Spherical

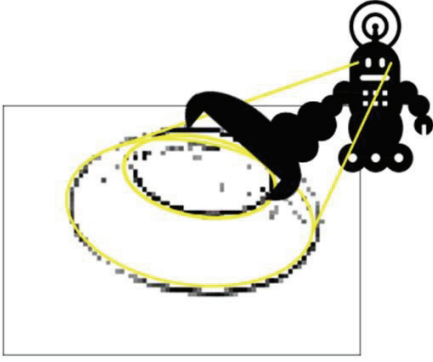


Fig. 2. Robot grasps the saucer

Shells (FCSS) [11] improves the computational efficiency by calculating prototypes analytically with "algebraical" distance. Fuzzy c -Quadratic Shells (FCQS) adopts quadratic polynomial as a distance function, and extracts clusters in the shape of ellipsoid, hyperbola, and parabola. Probabilistic approach for FCQS and robust FCQS which are less affected by noise also have been studied [12], [13], [14]. FCV-based robust shell clustering is proposed but there are problems in terms of computational time [15].

For improving computational time and detection accuracy, we defuzzify fuzzy c -regression models (FCRM) [16] and formulate hard c -regression models (HCRMs). Hence, the first half of our approach can be regarded as one of the switching regression models [1], [2] on an extended high dimensional space. The approach is based on an objective function. The procedure is repetitive and the termination condition by using another error evaluation function is adopted.

HCRMs quickly extract circles or ellipses by repeating random initializations. We test the performance using 20 data sets and report that the detection time on average is 14 milliseconds.

II. SHELL CLUSTERING WITH HARD c -REGRESSION MODELS

Fuzzy c -regression models (FCRM) regards c -regression equations as clustering prototypes. In this section, we formulate the methods which capture nonlinear structures of data sets efficiently by applying hard clustering to FCRM.

Two dimensional data (i.e., points on a plane), each of which is written as $\mathbf{x}_k = (x_{k1}, x_{k2})^\top$, are extended as:

$$\mathbf{x}_k^* = (x_{k1}, x_{k2}, x_{k1}^2, x_{k2}^2, x_{k1}x_{k2})^\top, \quad k = 1, \dots, n, \quad (1)$$

where $\mathbf{x}_k^*$ denotes the vector of dependent variables. We minimize the squared errors of the following regression equation

so that z_{ki} is as close to 1 as possible.

$$\begin{aligned} z_{ki} &= a_{i1}x_{k1} + a_{i2}x_{k2} + a_{i3}x_{k1}^2 \\ &\quad + a_{i4}x_{k2}^2 + a_{i5}x_{k1}x_{k2} \\ &= \mathbf{x}_k^{*\top} \mathbf{a}_i \\ \mathbf{a}_i &= (a_{i1}, a_{i2}, a_{i3}, a_{i4}, a_{i5})^\top, \end{aligned} \quad (2)$$

where i denotes the cluster number ($i = 1, \dots, c$). (2) represents an ellipse when z_{ki} is a positive constant. We choose $\mathbf{a}_i$ such that z_{ki} are all close to a constant value. The constant value can be 1, and the squared error between 1 and z_{ki} is defined as:

$$\begin{aligned} d_{ki}^2 &= (a_{i1}x_{k1} + a_{i2}x_{k2} + a_{i3}x_{k1}^2 \\ &\quad + a_{i4}x_{k2}^2 + a_{i5}x_{k1}x_{k2} - 1)^2 \\ &= (\mathbf{x}_k^* \mathbf{a}_i - 1)^2 \end{aligned} \quad (3)$$

and the objective function is defined as:

$$L = \sum_{i=1}^c \sum_{k=1}^n u_{ki} d_{ki}^2. \quad (4)$$

$u_{ki} \in [0, 1]$ is the membership and 1 or 0 is assigned to u_{ki} , since L is linear with respect to u_{ki} . For updating u_{ki} , following rule is adopted.

$$u_{ki} = \begin{cases} 1 & (i = \arg \min_{1 \leq j \leq c} d_{kj}^2), \\ 0 & (\text{otherwise}). \end{cases} \quad (5)$$

Regression coefficient $\mathbf{a}_i$ is the least squares estimator of the following simultaneous equations.

$$\begin{bmatrix} u_{1i} \mathbf{x}_1^{*\top} \mathbf{a}_i \\ u_{2i} \mathbf{x}_2^{*\top} \mathbf{a}_i \\ \vdots \\ u_{ni} \mathbf{x}_n^{*\top} \mathbf{a}_i \end{bmatrix} = \begin{bmatrix} u_{1i} \\ u_{2i} \\ \vdots \\ u_{ni} \end{bmatrix}. \quad (6)$$

Let us rewrite (6) as:

$$X \mathbf{a}_i = \mathbf{u}_i. \quad (7)$$

Then, $\mathbf{a}_i$ is computed with singular value decomposition

$$X = Z \Lambda P^\top, \quad (8)$$

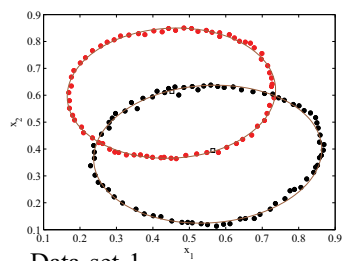
where Z is an $n \times 5$ matrix, Λ is the diagonal matrix of singular values $\lambda_j, j = 1, \dots, 5$, and $P^\top$ denotes the transpose of P , an 5×5 matrix. Column vectors of Z and P are all length 1.

By using the pseudoinverse, the solution becomes:

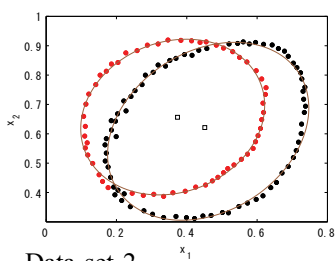
$$\mathbf{a}_i = P \Lambda^+ Z^\top \mathbf{u}_i, \quad (9)$$

where Λ^+ is an inverse of Λ and $1/\lambda_j$ is replaced by zero when $\lambda_j = 0$. This solution procedure is embedded by certain programming languages, such as Matlab.

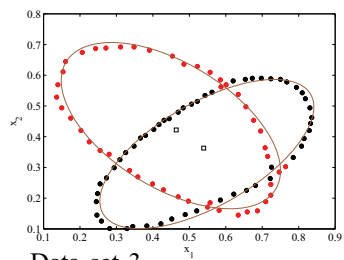
As $\mathbf{a}_i$ and u_{ki} are updated repeatedly, the above mentioned regression method can be seen as one of the switching regression models [1], [2]. In our proposed method, initial values of $\mathbf{a}_i$ are given randomly and then the initial membership values are calculated. We choose $\mathbf{a}_i$ which minimize the objective



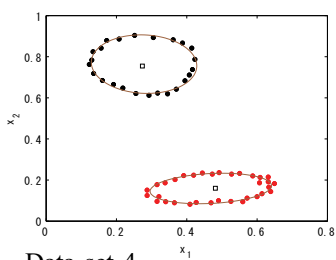
Data set 1



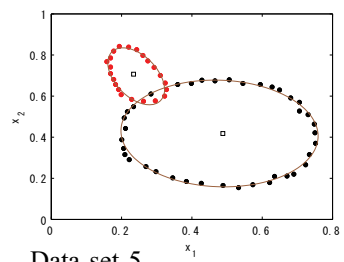
Data set 2



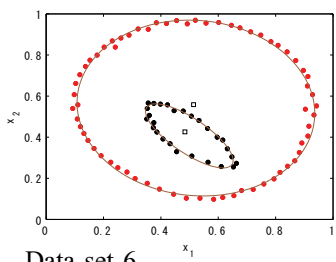
Data set 3



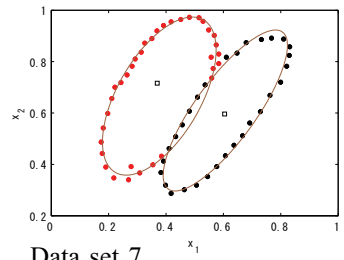
Data set 4



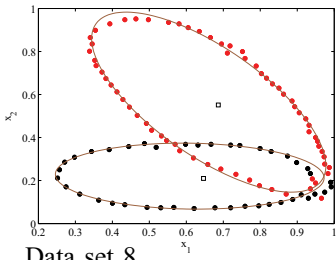
Data set 5



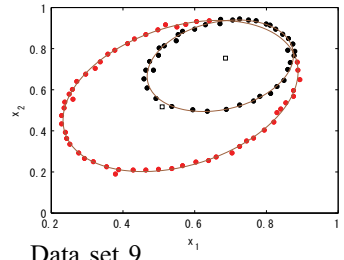
Data set 6



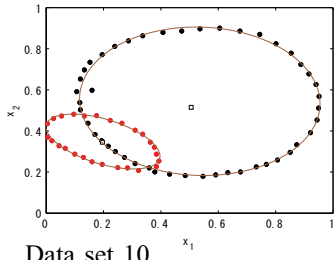
Data set 7



Data set 8

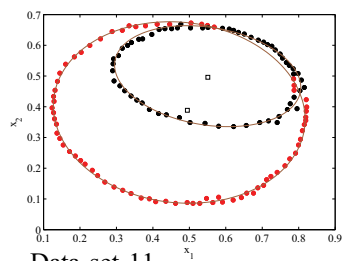


Data set 9

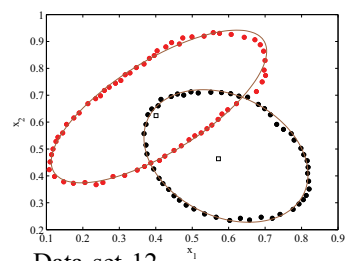


Data set 10

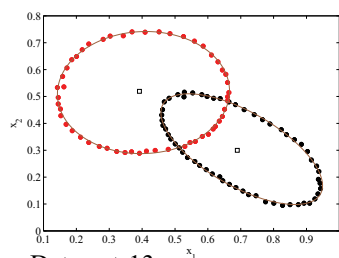
Fig. 3. Data sets 1 - 10



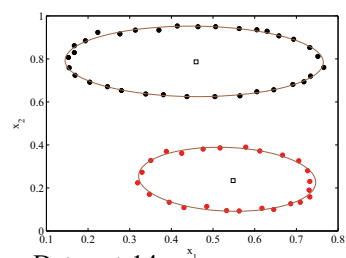
Data set 11



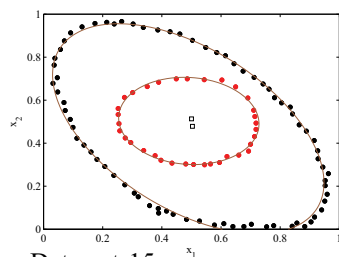
Data set 12



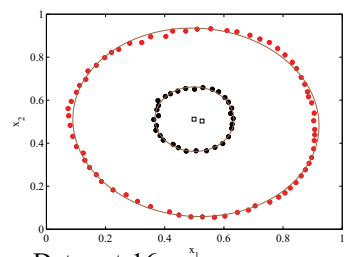
Data set 13



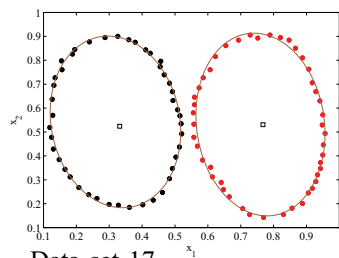
Data set 14



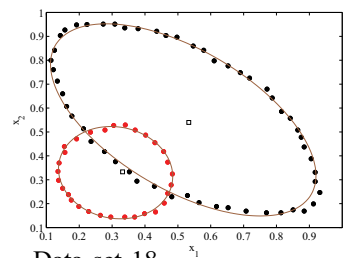
Data set 15



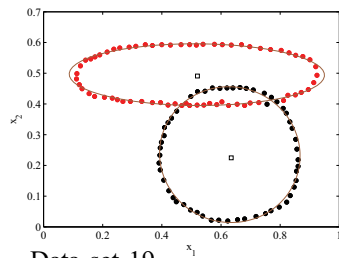
Data set 16



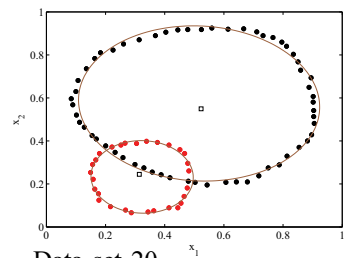
Data set 17



Data set 18



Data set 19



Data set 20

Fig. 4. Data sets 11 - 20

function L . But, for example, for data set 4 in Fig. 3, L often becomes smaller when the detection is false than when the detection is correct. So, we repeatedly change initial values until the following evaluation function E becomes less than a given threshold e .

$$E = \sum_{i=1}^c \left(\sum_{k=1}^n u_{ki} \beta_{ki} | \mathbf{x}_k - \mathbf{v}_i | / \sum_{k=1}^n u_{ki} \right) \quad (10)$$

$\mathbf{v}_i$ is a center and $|\cdot|$ denotes the Euclidean distance. β_{ki} is calculated such that $\mathbf{y}_k$ is the projection of $\mathbf{x}_k$ on the oval.

$$\begin{aligned} \mathbf{y}_k &= \mathbf{x}_k + \beta_{ki}(\mathbf{x}_k - \mathbf{v}_i) \\ \mathbf{y}_k^{*\top} \mathbf{a}_i &= 1, \end{aligned} \quad (11)$$

where $\mathbf{y}_k^*$ are extended 5 dimensional data of $\mathbf{y}_k$. The smallest absolute value of β_{ki} is chosen.

The procedure is repeated to a maximum of r -times if the condition $E < e$ is not met, then $\mathbf{a}_i$ which minimizes E is chosen.

The detection algorithm using the above mentioned update rules is as follows.

1. Extend $\mathbf{x}_k$ to $\mathbf{x}_k^*$.
2. Initialize regression coefficients $\mathbf{a}_i$ randomly.
3. Compute the errors by (3).
4. Update the membership value u_{ki} by (5).
5. Update $\mathbf{a}_i$ by (6).
6. If the change in the objective function values (4) becomes small enough ($|L^{old} - L^{new}| < 10^{-7}$), then go to 7, else go to 3.
7. Compute evaluation function (10) for detecting ellipses.
8. If repetition time is less than r and $E > e$, then go to 2, else go to 9.
9. Select $\mathbf{a}_i$ which minimizes the evaluation function (10).

In the next section we set as $r = 200$ and $e = 0.02$.

III. DETECTION PERFORMANCE

To evaluate the effectiveness of the proposed method, experiments were conducted using the 20 different data sets shown in Figs. 3-4. Table II-I shows the results. Initial values of $\mathbf{a}_i$ are given as:

$$\begin{aligned} \mathbf{a}_1 &= (2.10561.3352 - 1.6725 - 1.6585 - 0.1724) \times 1.1 \\ &\quad + (rand - 0.5) \times 2 \\ \mathbf{a}_2 &= (2.10561.3352 - 1.6725 - 1.6585 - 0.1724) \times 1.8 \\ &\quad + (rand - 0.5) \times 2 \end{aligned}$$

where $rand$ denotes a vector of uniform random numbers $rand \in [0, 1]$. The program stops when E in (10) is less than 0.02.

Each data set is tested 10 times and values of E , detection times and the number of repetitions (i.e., the numbers of initializations of $\mathbf{a}_i$ until $E < 0.02$ is satisfied) are given in Tables II-III. In all cases, the algorithm terminated before

TABLE I
DETECTION TIME ON ALL DATA SET

	E	time (s)	repetitions (initializations)
Max	0.0197	0.155	155
Mean	0.0139	0.014	11.2
Min	0.0075	0.002	1

reaching the final termination condition ($r = 200$). For Data set 1, $E=0.0107$ is attained in 18 milliseconds on average. The detected ellipses are shown in Fig. 3 upper left. For example, the maximum detection time for Data set 10 is 6 milliseconds, the mean time is 4 milliseconds and the minimum time is 3 milliseconds. Table I shows the results on all data sets. The mean detection time on all the data sets is 14 milliseconds. Experiments are performed on DELL PRECISION T5400 3.16GHz 4GB.

The methods based on Hough transform are most popular for ellipse detection. By comparison, the time for detecting two ellipses by Hough-based ellipse detection algorithm is about 1500 seconds and that of Randomized Hough Transform is about 15 seconds [8]. 10% detection error at minimum is reported in [8].

IV. CONCLUSION

The shell clustering method in this paper is based on the hard c -regression models with random initializations. An additional evaluation function is used for terminating the repetitive algorithm. With this enhancement, HCRM accurately detects circles or ellipses very quickly.

Our possible future work includes setting more effective initial random values, coping with a large number of clusters, and robustizing the method by using the noise clustering approach [17].

REFERENCES

- [1] S. M. Goldfeld and R. E. Quandt, "Techniques for estimating switching regressions," *Studies in Nonlinear Estimation* (eds. by S. M. Goldfeld and R. E. Quandt), pp. 3-35, Ballinger, Cambridge, Massachusetts, 1976.
- [2] R. E. Quandt, "A new approach to estimating switching regressions," *Journal of American Statistical Association*, vol. 67, pp. 306-310, 1972.
- [3] K. K. R. Yip, K. S. P. Tam and N. K. D. Leung, "Modification of Hough transform for circles and ellipses detection using a 2-dimensional array," *Pattern Recognition*, vol. 25, pp. 1007-1022, 1992.
- [4] J. H. Yoo and I. K. Sethi, "An ellipse detection method from the polar and pole definition of conics," *Pattern Recognition*, vol. 26, pp. 307-315, 1993.
- [5] A. S. Aguado, M. E. Montiel and M. S. Nixon, "Ellipse detection via gradient direction in the Hough transform," *Proceedings of 5th IEEE International Conference on Image Processing and its Applications*, Edinburgh, UK, pp. 375-378, 1995.
- [6] C. T. Ho, and L. H. Chen, "A fast ellipse/circle detector using geometry symmetry," *Pattern Recognition*, vol. 28, no. 1, pp. 117-124, 1995.
- [7] P. Kultanen, L. Xu, and E. Oja, "Randomized hough transform (rht)," *In Proc., 10th Int'l Conf. Pattern Recognition*, vol. 1, pp. 631-635, 1990.
- [8] R. A. McLaughlin, "Randomized hough tranform: Improved ellipse detection with comparison," *Pattern Recognition Letters*, vol. 19, no. 3-4, pp.299-305, 1998.
- [9] R. N. Dave, "Fuzzy shell-clustering and application to circle detection in digital images," *Int. J.Gen. syst.*, vol.16, pp.343-355, 1990.

TABLE II
DETECTION TIME ON DATA SETS 1 - 10

data set 1			
	E	time (s)	repetitions (initializations)
Max	0.0107	0.043	16
Mean	0.0107	0.018	6.4
Min	0.0107	0.003	1
data set 2			
	E	time (s)	repetitions (initializations)
Max	0.0168	0.155	110
Mean	0.0142	0.041	26.8
Min	0.0118	0.006	2
data set 3			
	E	time (s)	repetitions (initializations)
Max	0.0197	0.070	70
Mean	0.0197	0.022	20.8
Min	0.0197	0.002	1
data set 4			
	E	time (s)	repetitions (initializations)
Max	0.0138	0.015	16
Mean	0.0138	0.005	5.3
Min	0.0138	0.002	1
data set 5			
	E	time (s)	repetitions (initializations)
Max	0.0122	0.018	21
Mean	0.0121	0.004	4.0
Min	0.0121	0.002	1
data set 6			
	E	time (s)	repetitions (initializations)
Max	0.0158	0.154	155
Mean	0.0158	0.074	74.0
Min	0.0158	0.007	5
data set 7			
	E	time (s)	repetitions (initializations)
Max	0.0188	0.045	51
Mean	0.0188	0.017	18.4
Min	0.0186	0.004	2
data set 8			
	E	time (s)	repetitions (initializations)
Max	0.0192	0.032	17
Mean	0.0188	0.011	5.9
Min	0.0185	0.003	1
data set 9			
	E	time (s)	repetitions (initializations)
Max	0.0196	0.035	31
Mean	0.0148	0.014	11.8
Min	0.0095	0.003	1
data set 10			
	E	time (s)	repetitions (initializations)
Max	0.0116	0.006	5
Mean	0.0115	0.004	2.5
Min	0.0110	0.003	1

TABLE III
DETECTION TIME ON DATA SETS 11 - 20

data set 11			
	E	time (s)	repetitions (initializations)
Max	0.0100	0.028	2
Mean	0.0098	0.006	1.2
Min	0.0095	0.003	1
data set 12			
	E	time (s)	repetitions (initializations)
Max	0.0125	0.042	31
Mean	0.0125	0.012	7.0
Min	0.0125	0.003	1
data set 13			
	E	time (s)	repetitions (initializations)
Max	0.0075	0.025	15.0
Mean	0.0075	0.008	3.7
Min	0.0075	0.003	1
data set 14			
	E	time (s)	repetitions (initializations)
Max	0.0115	0.004	4.0
Mean	0.0115	0.002	1.5
Min	0.0115	0.002	1
data set 15			
	E	time (s)	repetitions (initializations)
Max	0.0166	0.030	25
Mean	0.0166	0.010	7.1
Min	0.0166	0.003	1
data set 16			
	E	time (s)	repetitions (initializations)
Max	0.0126	0.075	38
Mean	0.0126	0.016	10.4
Min	0.0126	0.002	1
data set 17			
	E	time (s)	repetitions (initializations)
Max	0.0125	0.007	5
Mean	0.0125	0.004	2.7
Min	0.0125	0.003	1
data set 18			
	E	time (s)	repetitions (initializations)
Max	0.0135	0.007	5
Mean	0.0135	0.004	2.5
Min	0.0135	0.002	1
data set 19			
	E	time (s)	repetitions (initializations)
Max	0.0122	0.048	5
Mean	0.0122	0.008	1.9
Min	0.0122	0.002	1
data set 20			
	E	time (s)	repetitions (initializations)
Max	0.0187	0.023	24
Mean	0.0187	0.010	10.1
Min	0.0187	0.005	3

- [10] R. N. Dave and K. Bhaswan, "Adaptive fuzzy C -shells clustering and detection of ellipses," *IEEE Trans. on Neural Networks*, vol.3, No.5, pp.643-662, 1992.
- [11] R. Krishnapuram, O. Nasraoui and H. Frigui, "The fuzzy C spherical shells algorithms: A new approach," *IEEE Trans. on Neural Networks*, vol.3, no.5, pp.663-671, 1992.
- [12] R. Krishnapuram, H. Frigui and O. Nasraoui, "Quadric shell clustering algorithms and the detection of second degree curves," *Pattern Recognition Lett*, vol.14, no.7, pp.545-552, 1993.
- [13] R. Krishnapuram, H. Frigui and O. Nasraoui, "Fuzzy and possibilistic shell clustering algorithms and their application to boundary detection and surface approximation - partI," *IEEE Trans. on Fuzzy systems*, vol.3, no.1, pp.29-43, 1995.
- [14] R. Krishnapuram, H. Frigui and O. Nasraoui, "Fuzzy and possibilistic shell clustering algorithms and their application to boundary detection and surface approximation - partII," *IEEE Trans. on Fuzzy systems*, vol.3, no.1, pp.44-60, 1995.
- [15] K. Honda, N. Togo, H. Ichihashi, "Generalization of FCV and robust shell clustering based on least absolute deviations," *Journal of Japan Society for Fuzzy Theory and Intelligent informatics*, Vol.15, No.6, pp.693-701, 2003.
- [16] R. Hathaway and J. Bezdek, "Switching regression model and fuzzy clustering," *IEEE Trans. on Fuzzy systems*, vol. 1, pp. 195-204, 1993.
- [17] R. N. Dave, "Characterization and detection of noise in clustering," *Pattern Recognition Letters*, vol.12, pp. 657-664, 1991.