General Type-2 Fuzzy C-Means Algorithm for Uncertain Fuzzy Clustering

Ondrej Linda, *Student Member, IEEE*, Milos Manic, *Senior Member, IEEE*

Abstract— Pattern recognition in real world data is subject to various sources of uncertainty that should be appropriately managed. The focus of this paper is the management of uncertainty associated with parameters of fuzzy clustering algorithms. Type-2 Fuzzy Sets (T2 FSs) received increased research interest in the past decade primarily due to their potential to model various uncertainties. However, because of the computational intensity of the processing of General T2 (GT2) Fuzzy Sets (FSs), only their constrained version, the Interval T2 (IT2) FSs, were typically used. Fortunately, the recently introduced concepts of α*-planes* and *zSlices* allow for efficient representation and computation with GT2 FSs. Following this recent development, this paper presents a novel approach for uncertain fuzzy clustering using the General Type-2 Fuzzy C-Means (GT2 FCM) algorithm. The proposed method builds on top of the previously published IT2 FCM algorithm, which is extended via the α*-planes* representation theorem. The fuzzifier parameter of the FCM algorithm can be expressed using linguistic terms such as "*Small*" or "*High*", modeled as T1 FSs. This linguistic fuzzifier value is then used to construct the GT2 FCM cluster membership functions. The linguistic uncertainty is transformed into uncertain fuzzy positions of the extracted clusters. The GT2 FCM algorithm was found to balance the performance of T1 FCM algorithms in various uncertain pattern recognition tasks and provide increased robustness in situations where noisy or insufficient training data are present.

***Index Terms*— General Type-2 Fuzzy Sets, α-Planes Representation, Fuzzy C-Means, Uncertainty, Pattern Recognition**

I. Introduction

PATTERN recognition algorithms are commonly subject to various sources of uncertainty that should be appropriately managed. Several forms of uncertainties are typically recognized. For instance, the uncertainty related to the input data themselves (e.g. clustering heterogeneous input data such as real numbers, intervals and linguistic terms), the uncertainty related to the interpretation of the computed result and the uncertainty related to the suitable parameters' values of the pattern recognition algorithms. The scope of this work is the appropriate uncertainty management for fuzzy clustering algorithms [1]-[4].

Ondrej Linda is with the Computer Science Department, University of Idaho, Idaho Falls, ID 83402 USA phone: 208-227-3919; e-mail: olinda@uidaho.edu.

Dr. Milos Manic is with the Computer Science Department, University of Idaho, Idaho Falls, ID 83402 USA

Examples of the management of the first type of uncertainty are the nonparametric and parametric models for fusing heterogeneous fuzzy data developed by Pedrycz and Hathaway et al [5], [6]. In this work, the heterogeneous input data are encoded into a uniform internal representation that allows for application of common processing methods such as fuzzy clustering. Subsequently, the internal representation can be decoded back to the original form. While this approach enables the use of standard pattern recognition tools to heterogeneous input data, it requires a selection of suitable algorithm parameters (e.g. the fuzzifier parameter for FCM algorithm). An example of managing the uncertainty related to interpretation of the results is the concept of shadowed sets proposed by Pedrycz [7], [8]. In this work, the shadowed sets were used to interpret the fuzzy cluster partition and distinguish between cluster cores and cluster shadows with ambiguously assigned data points. Finally, an example of managing the uncertainty related to algorithm parameters' values is the work of Rhee and Hwang on uncertain fuzzy clustering with IT2 FCM algorithm [4], [9], [10]. This paper extends the IT2 FCM algorithm into a GT2 FCM algorithm via using GT2 fuzzy cluster membership functions.

Type-2 Fuzzy Logic (FL) became the scope of work for many researchers in recent years [11]-[16]. T2 FL has been successfully applied in many engineering areas, demonstrating improved performance and robustness of T2 FL relative to T1 FL when confronted with various sources of data uncertainties [17]-[22]. Unlike the T1 FL, the T2 FL systems use individual fuzzy sets with membership grades that are themselves fuzzy sets. These secondary fuzzy grades provide additional degrees of freedom for modeling and coping with dynamic input uncertainties.

However, the early representations of General T2 (GT2) Fuzzy Sets (FSs) did not provide computationally feasible algorithms [23]. Instead, the Interval T2 (IT2) FSs, which use constrained secondary membership functions, were used [24]. The IT2 FL offers a compromise between the computational inexpensiveness of T1 FL and the uncertainty modeling capability of GT2 FL. Many researchers proposed to extend the T1 FL based algorithms by incorporating the IT2 FL. Mitchell proposed the extension of pattern recognition using IT2 FSs [25]. Wu and Mendel extended the uncertainty measures for T1 FSs to IT2 FSs in [26]. Zeng et al. proposed the IT2 Gaussian mixture models [27]. Rhee and Hwang published several papers discussing the extension of several T1 fuzzy pattern recognition algorithms into IT2 FL, namely the IT2 fuzzy C-Spherical shells algorithm [28], the IT2 fuzzy perceptron [29], the IT2 fuzzy K-nearest neighbor algorithm [30] and the IT2 FCM algorithm [4], [9], [10].

The recently introduced representations of α*-planes* [31], [32] and *zSlices* [33] offer a computational efficient and viable framework for representing and computing with the GT2 FSs. In both cases, the α*-planes* and the *zSlices* representation theorems allow for treating the GT2 FSs as a composition of multiple IT2 FSs, each raised to the respective level of α or *z*. The operations on GT2 FSs become a multiple application of the efficient arithmetic of IT2 FSs. This development allows for extending the previously developed IT2 FL algorithms using the α*-planes* representation theorem and thus managing the uncertainty with fully-developed GT2 FSs. An example of this development is the work of Zhai and Mendel on uncertainty measures for GT2 FSs [34].

This paper proposes a novel method for managing the uncertainty associated with the selection of the fuzzifier paremeter *m* for the FCM algorithm. The value of *m* has a direct impact on the location and quality of the cluster partition. However, it is difficult to express the notion of fuzziness in the input data using precise real value for T1 FCM algorithm or even as an interval value for IT2 FCM algorithm [1]-[4]. To alleviate this issue, the proposed GT2 FCM algorithm allows for linguistically expressing the fuzzifier value using terms such as "*Small*" or "*High*" modeled as T1 FSs. The resulting cluster membership functions are implemented as GT2 FSs represented using the α*-planes* theorem. A novel hard-partitioning rule is proposed for the final input-cluster assignment. In addition, the Quasi-T2 (QT2) FCM algorithm is also introduced as a simplified version of the GT2 FCM method.

The rest of the paper is organized as follows. Section II discusses the background of GT2 FSs and the α*-plane* representation theorem. The original T1 FCM algorithm and its IT2 FL extension are reviewed in Section III. Section IV and Section V present the novel GT2 and QT2 FCM algorithms. Experimental results and comparisons are presented in Section VI. Finally, the paper is concluded in Section VII.

II. General Type-2 Fuzzy Sets

This section provides background overview of GT2 FSs and the fundamentals of the α*-plane* representation.

A. General Type-2 Fuzzy Sets

A GT2 FS $\tilde{A}$ can be expressed on the universe of discourse *X* using its T2 fuzzy membership function $\mu_{\tilde{A}}(x,u)$, where $x \in X$ and $u \in J_x$ [11]:

$$\tilde{A} = \int_{x\in X}\int_{u\in J_x} \mu_{\tilde{A}}(x,u)/(x,u) \quad J_x \subseteq [0,1] \tag{1}$$

Here, variable *x* and *u* are the primary and the secondary variables and J_x denotes the support of the secondary membership function also called the primary membership of *x*. Operator $\iint$ denotes union over all possible values of *x* and *u*, and $\mu_{\tilde{A}}(x,u) \in [0,1]$. Two representations of GT2 FSs are commonly adopted; the *vertical-slice* representation and the *wavy-slice* representations.

First, the *vertical-slice* representation is considered. By instantiating a specific value for $x = x'$, a vertical slice $\mu_{\tilde{A}}(x',u)$ of the fuzzy membership function $\mu_{\tilde{A}}(x,u)$ can be obtained. This vertical slice defines a secondary membership function $\mu_{\tilde{A}}(x = x',u)$ for $x' \in X$ and $\forall u \in J_{x'} \subseteq [0,1]$:

$$\mu_{\tilde{A}}(x = x',u) \equiv \mu_{\tilde{A}}(x') = \int_{u\in J_{x'}} f_{x'}(u)/u \quad J_{x'} \subseteq [0,1] \tag{2}$$

Here, $f_{x'}(u)$ denotes the secondary grade or the amplitude of the secondary membership function and $f_{x'}(u) \in [0,1]$. Assuming that the domain of primary variable *x* is discretized using *N* samples, the GT2 FS $\tilde{A}$ can be represented as a composition of all its vertical slices:

$$\tilde{A} = \sum_{i=1}^{N}\left[\int_{u\in J_{x_i}} f_{x_i}(u)/u\right]\Big/ x_i \tag{3}$$

Next, the *wavy-slice* representation is considered. The GT2 FS $\tilde{A}$ can be constructed as a composition of its embedded FSs $\tilde{A}_e$. Again, for a discrete universe of discourse with *N* elements, the embedded FS $\tilde{A}_e$ can be described as:

$$\tilde{A}_e = \sum_{i=1}^{N}\left[f_{x_i}(\theta_i)/\theta_i\right]\Big/ x_i \quad \theta_i \in J_{x_i} \subseteq U \in [0,1] \tag{4}$$

According to the Mendel and John representation theorem [35], the GT2 FS $\tilde{A}$ can be described as a union of all of its *n* embedded T2 FSs:

$$\tilde{A} = \bigcup_{j=1}^{n} \tilde{A}_e^j \tag{5}$$

For the discretized primary domain *X*, the centroid $C_{\tilde{A}}$ of a GT2 FS $\tilde{A}$ can be calculated using the *Extension Principle* and by enumerating all of the embedded fuzzy sets [16]:

$$C_{\tilde{A}} = \int_{\theta_1\in J_{x_1}}\ldots\int_{\theta_N\in J_{x_N}} \left[f_{x_1}(\theta_1)\wedge\ldots\wedge f_{x_N}(\theta_N)\right]\Bigg/ \frac{\sum_{i=1}^{N} x_i\theta_i}{\sum_{i=1}^{N}\theta_i} \tag{6}$$

Here, every possible combination of variables $\theta_1,\ldots,\theta_N$ forms an embedded FS, which has a secondary grade of $f_{x_1}(\theta_1)\wedge\ldots\wedge f_{x_N}(\theta_N)$. Operator $\wedge$ is the specific t-norm used (e.g. the minimum operator).

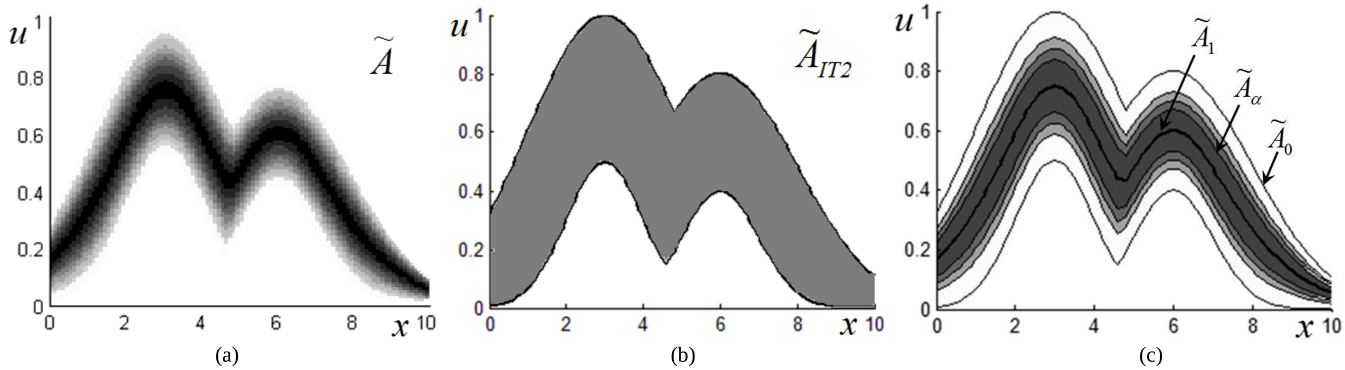


Fig. 1 General T2 fuzzy set $\tilde{A}$ (a), its IT2 variant $\tilde{A}_{IT2}$ (b) and its α -plane representation (c).

Assuming that each primary membership J_{x_i} was discretized into M_i points, the number of embedded fuzzy sets that have to be enumerated in (6) is $n = \prod_{i=1}^{N} M_i$. Already for large discretization steps n becomes prohibitively large.

The crisp output value y can be obtained by applying one of the available defuzzification methods to the type-reduced centroid $C_{\tilde{A}}$. As an example, the centroid defuzzifier is commonly used [11]:

$$y = \frac{\sum_{i=1}^{n} y_i C_{\tilde{A}}(y_i)}{\sum_{i=1}^{n} C_{\tilde{A}}(y_i)} \tag{7}$$

Here, n denotes the number of discretized samples in the output domain of variable y and y_i is the discretized sample.

B. α -Plane Representation for GT2 Fuzzy Sets

The following notation for the α *-plane* representation was adopted from [32], [36], [37]. The α *-plane* representation was independently developed in the work of several authors [31], [33], [38].

An α *-plane* $\tilde{A}_\alpha$ of a GT2 Fs $\tilde{A}$ can be defined as the union of all primary memberships of $\tilde{A}$ with secondary grades greater than or equal to α :

$$\tilde{A}_\alpha = \int_{\forall x \in X} \int_{\forall u \in J_x} \{(x,u) \mid f_x(u) \geq \alpha\} \tag{8}$$

An α *-cut* of the secondary membership function $\mu_{\tilde{A}}(x)$ can be denoted as $S_{\tilde{A}}(x|\alpha)$ and expressed as:

$$S_{\tilde{A}}(x|\alpha) = \left[s_{\tilde{A}}^{L}(x|\alpha), s_{\tilde{A}}^{R}(x|\alpha)\right] \tag{9}$$

Hence, an α *-plane* $\tilde{A}_\alpha$ is a composition of all α -cuts of all secondary membership functions:

$$\tilde{A}_\alpha = \int_{\forall x \in X} S_{\tilde{A}}(x|\alpha) = \int_{\forall x \in X} \left(\int_{\forall u \in \left[s_{\tilde{A}}^{L}(x|\alpha), s_{\tilde{A}}^{R}(x|\alpha)\right]} u \right) \Big/ x \tag{10}$$

It is apparent that the well known Footprint of Uncertainty (FOU) of the GT2 FS $\tilde{A}$ is equivalent to:

$$FOU(\tilde{A}) = \tilde{A}_0 \tag{11}$$

Each α *-plane* $\tilde{A}_\alpha$ is bounded from above by its upper membership function $\bar{\mu}_{\tilde{A}}(x|\alpha)$ and from below by its lower membership function $\underline{\mu}_{\tilde{A}}(x|\alpha)$. Using the α *-cuts* boundaries of each vertical slice (9), these bounds can be expressed as:

$$\bar{\mu}_{\tilde{A}}(x|\alpha) = \int_{\forall x \in X} s_{\tilde{A}}^{R}(x|\alpha) \tag{12}$$

$$\underline{\mu}_{\tilde{A}}(x|\alpha) = \int_{\forall x \in X} s_{\tilde{A}}^{L}(x|\alpha) \tag{13}$$

By raising the α *-plane* $\tilde{A}_\alpha$ to the level of α , a special IT2 FS is created. This FS was named α *-level* T2 FS $R_{\tilde{A}_\alpha}(x,u)$ in [31], [39] and denoted as:

$$R_{\tilde{A}_\alpha}(x,u) = \alpha / \tilde{A}_\alpha, \ \forall x \in X, \forall u \in J_x \tag{14}$$

The different discussed variants of GT2 FSs are depicted in Fig. 1. Finally, according to Liu's representation theorem, the GT2 FS $\tilde{A}$ can be constructed as a composition of all of its individual α *-level* T2 FSs [31]:

$$\tilde{A} = \bigcup_{\alpha \in [0,1]} \alpha / \tilde{A}_\alpha \tag{15}$$

It should be noted here, that the $\bigcup$ symbol denotes the union set-theoretic operations, which for all points computes the maximum membership grade for all α *-planes*. Furthermore, Liu used the α *-plane* representation theorem to express the centroid $C_{\tilde{A}}(x)$ of GT2 FS $\tilde{A}$ as a composition of individual centroids $C_{\tilde{A}_\alpha}(x)$ of the respective α *-level* T2 FSs $R_{\tilde{A}_\alpha}(x,u)$:

$$C_{\tilde{A}}(x) = \bigcup_{\alpha\in[0,1]} \alpha / C_{\tilde{A}_\alpha}(x) \qquad (16)$$

Because each α*-level* T2 FS $R_{\tilde{A}_\alpha}(x,u)$ is an interval-valued set, centroid $C_{\tilde{A}_\alpha}(x)$ will become an interval set completely determined by its left and right boundaries $C_{\tilde{A}_\alpha}(x) = [c^L_{\tilde{A}}(\alpha), c^R_{\tilde{A}}(\alpha)]$. Hence, it follows that:

$$C_{\tilde{A}}(x) = \bigcup_{\alpha\in[0,1]} \alpha \Big/ [c^L_{\tilde{A}}(\alpha), c^R_{\tilde{A}}(\alpha)] \qquad (17)$$

Equation (16) showed a new way for computing the centroid $C_{\tilde{A}}(x)$ of a GT2 FS $\tilde{A}$ via type-reducing each α-*level* T2 FS $R_{\tilde{A}_\alpha}(x,u)$ and then fusing the results together.

Several algorithms are applicable for the type-reduction of GT2 FSs represented using α*-planes*. Originally, independent application of the Karnik-Mendel (KM) or the Enhanced KM algorithms at each α*-plane* was proposed [31]-[33]. Recently, new and faster algorithms exploiting the structural dependencies of neighboring α*-planes* have been emerging, namely the Centroid Flow algorithm [36], [37], the Enhanced Type-Reduction algorithm [40] or the Monotone Centroid Flow algorithm [41].

III. Type-1 and Interval Type-2 Fuzzy C-Means

This section provides an overview of the T1 FCM algorithm [1]-[3] and the uncertain clustering using IT2 FCM algorithm proposed by Hwang and Rhee [4], [10].

A. Type-1 Fuzzy C-Means

The original FCM algorithm seeks an exhaustive partition of the input data set into a non-empty set of clusters [1]-[3]. The employed fuzzy membership grades allow each point to maintain gradual membership to multiple clusters. Consider an input data set $X = \{\mathbf{x}_1, ..., \mathbf{x}_n\}, X \subseteq \Re^d$, where n is the number of data points and d denotes the input space dimensionality. The membership of pattern $\boldsymbol{x}_j$ into a set C of c clusters can be defined by the membership vector $\mathbf{u}(\mathbf{x}_i)$ as [3]:

$$\mathbf{u}(\mathbf{x}_i) = (u_1(\mathbf{x}_i) ..., u_c(\mathbf{x}_i)) \qquad (18)$$

Subject to constraints:

$$u_j(\mathbf{x}_i) \in [0,1],\ \sum_{j=1}^{c} u_j(\mathbf{x}_i) = 1, \forall i \in \{1,..., n\} \qquad (19)$$

The complete assignment of all data points to the set of clusters can be described by a $c \times n$ fuzzy partition matrix U defined as:

$$U = (\mathbf{u}(\mathbf{x}_1), ..., \mathbf{u}(\mathbf{x}_n)),\ \sum_{j=1}^{n} u_j(\mathbf{x}_i) > 0, \forall j \in \{1,..., c\} \qquad (20)$$

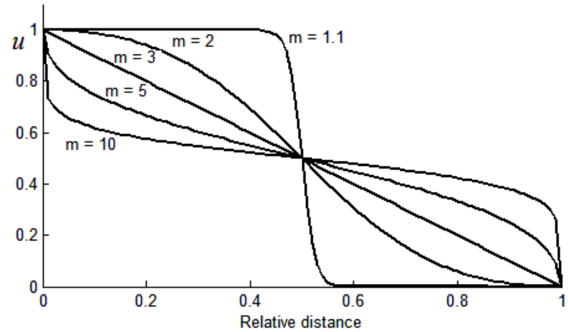


Fig. 2 T1 FCM membership functions.

The optimality of the membership matrix U with respect to the data set X and the cluster set C is achieved via the constrained minimization of the following objective function:

$$J(X,U,C) = \sum_{j=1}^{c}\sum_{i=1}^{n} u_j(\mathbf{x}_i)^m\, d_{ij}^2 \qquad (21)$$

Here, d_{ij} denotes the chosen distance norm (e.g. Euclidean) and parameter m, $m > 1$, is referred to as the fuzzifier, which controls the "fuzziness" of the extracted clusters. The greater the value of fuzzifier m, the softer the cluster boundaries become (depicted in Fig. 2). The objective function J can be minimized by iteratively re-computing the membership matrix U and updating the cluster positions $\boldsymbol{v}_j$ as follows:

$$u_j(\mathbf{x}_i) = \frac{1}{\sum_{l=1}^{c}\left(d_{ij}/d_{il}\right)^{2/(m-1)}} \qquad (22)$$

$$\mathbf{v}_j = \frac{\sum_{i=1}^{n} \mathbf{x}_i\, u_j(\mathbf{x}_i)^m}{\sum_{i=1}^{n} u_j(\mathbf{x}_i)^m} \qquad (23)$$

The FCM clustering algorithm starts from a randomly initialized cluster positions and converges when the accumulated change in cluster positions falls bellow certain threshold.

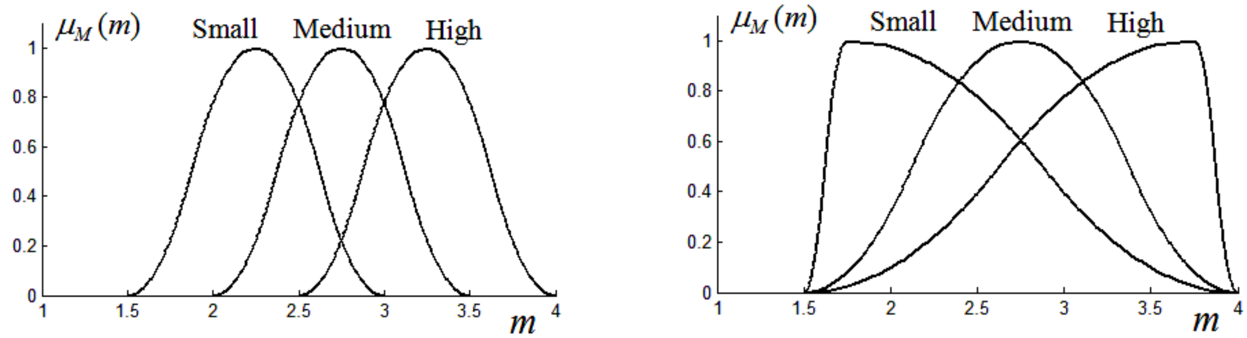
A hard-partitioning can be applied to determine the final assignment of pattern $\boldsymbol{x}_i$ to its designated cluster j as follows:

$$\begin{aligned}&\textbf{IF}\ (u_j(\mathbf{x}_i) > u_k(\mathbf{x}_i)),\ k = 1,..., c, k \neq j\\ &\qquad\qquad \textbf{THEN}\ \mathbf{x}_i \text{ belongs to cluster } j\end{aligned} \qquad (24)$$

B. Interval Type-2 Fuzzy C-Means

The value of fuzzifier m should reflect the "fuzziness" of the input data. However, in a majority of applications it is difficult or impossible to precisely specify the appropriate value of m. Furthermore, its specification using a precise numerical value is rather counter-intuitive as the amount of "fuzziness" can hardly be precisely expressed. In order to alleviate this issue, Hwang and Rhee proposed the IT2 FCM algorithm, which was designed to handle the uncertainty in fuzzy clustering using the tools of IT2 fuzzy logic [4], [10].

The IT2 FCM method considers an interval-valued fuzzifier $[m_L, m_R]$ rather than a precise numerical value. The interval



(a) (b)

Fig. 3 Two possible linguistic representation of the fuzzifier M using T1 fuzzy sets.

membership $[\underline{u}_j(\mathbf{x}_i), \bar{u}_j(\mathbf{x}_i)]$ of pattern $\boldsymbol{x}_i$ to cluster $\boldsymbol{v}_j$ can then be computed as:

$$\bar{u}_j(\mathbf{x}_i) = \max\left(\frac{1}{\sum_{l=1}^{c}\left(d_{ij}/d_{il}\right)^{2/(m_L-1)}}, \frac{1}{\sum_{l=1}^{c}\left(d_{ij}/d_{il}\right)^{2/(m_R-1)}}\right) \tag{25}$$

$$\underline{u}_j(\mathbf{x}_i) = \min\left(\frac{1}{\sum_{l=1}^{c}\left(d_{ij}/d_{il}\right)^{2/(m_L-1)}}, \frac{1}{\sum_{l=1}^{c}\left(d_{ij}/d_{il}\right)^{2/(m_R-1)}}\right) \tag{26}$$

The update of the cluster positions must take into account the interval membership grades, which results in an interval cluster coordinates. Using the definition of the interval centroid of IT2 FSs, the interval cluster position becomes:

$$\tilde{\mathbf{v}}_j = [\mathbf{v}_j^L, \mathbf{v}_j^R] = \sum_{u(x_1)\in J_{x_1}} \cdots \sum_{u(x_1)\in J_{xn}} 1 \Bigg/ \frac{\sum_{i=1}^{n} \mathbf{x}_i u_j(\mathbf{x}_i)^m}{\sum_{i=1}^{n} u_j(\mathbf{x}_i)^m} \tag{27}$$

The value of fuzzifier m in (27) switches from m_L or m_R according to (25) and (26). The individual values of the left and right cluster boundaries in each dimension can be computed by first sorting the order of patterns in particular dimension and then applying the Karnik-Mendel iterative procedure [23].

The precise position of the center of gravity can be obtained by defuzzifying the interval centroid $\tilde{\mathbf{v}}_j$ as follows:

$$\mathbf{v}_j = \frac{\mathbf{v}^L + \mathbf{v}^R}{2} \tag{28}$$

The hard-partitioning method for the IT2 FCM algorithm proceeds according to the T1 FCM algorithm once the clusters membership grades are defuzzified. The method proposed by Hwang and Rhee computes the precise membership of pattern $\boldsymbol{x}_i$ to cluster j as:

$$u_j(\mathbf{x}_i) = \frac{u_j^L(\mathbf{x}_i) + u_j^R(\mathbf{x}_i)}{2} \tag{29}$$

Where the left and right membership values are computed as:

$$u_j^L(\mathbf{x}_i) = \frac{\sum_{l=1}^{d} u_{jl}(\mathbf{x}_i)}{d} \tag{30}$$

where $u_{jl}(\mathbf{x}_i) = \begin{cases} \bar{u}_j(\mathbf{x}_i), & \text{if } \mathbf{x}_{il} \text{ uses } \bar{u}_j(\mathbf{x}_i) \text{ for } v_j^L \\ \underline{u}_j(\mathbf{x}_i) & otherwise \end{cases}$

$$u_j^R(\mathbf{x}_i) = \frac{\sum_{l=1}^{d} u_{jl}(\mathbf{x}_i)}{d}, \tag{31}$$

where $u_{jl}(\mathbf{x}_i) = \begin{cases} \bar{u}_j(\mathbf{x}_i), & \text{if } \mathbf{x}_{il} \text{ uses } \bar{u}_j(\mathbf{x}_i) \text{ for } v_j^R \\ \underline{u}_j(\mathbf{x}_i) & otherwise \end{cases}$

IV. General Type-2 Fuzzy C-Means Algorithm

This section introduces the GT2 FCM algorithm for uncertain fuzzy clustering. The value of the fuzzifier m has a direct impact on the obtained location and quality of the cluster partition. However, it is difficult to express the uncertain notion of fuzziness in the input data using precise values. The GT2 FCM algorithm allows for expressing the notion of fuzziness using linguistic terms modeled as T1 FSs. The linguistic fuzzifier value is then used to construct the GT2 fuzzy cluster membership functions. Using the α*-planes* representation, the input uncertainty is transformed into the uncertain fuzzy position of the extracted clusters.

A. Constructing GT2 fuzzy membership functions

The original T1 FCM algorithm requires specification of a precise fuzzifier value m. The IT2 FCM algorithm accepts an interval-valued fuzzifier [m_L, m_R], which resembled a uniform uncertainty about the appropriate value of fuzzifier m. The proposed GT2 FCM algorithm accepts a linguistic description of the fuzzifier value expressed as a T1 fuzzy set (e.g. "*Small"* or "*High*"). The linguistic fuzzifier value is here denoted as a T1 fuzzy set M. Fig. 3 illustrates two examples of encoding the linguistic notion of the appropriate fuzzifier value for the GT2 FCM algorithm using three linguistic terms.

Following the notation introduced in (9), the linguistic fuzzifier M can be expressed using its α*-cuts* as follows:

$$M = \bigcup_{\alpha\in[0,1]} \alpha / S_M(\alpha), \tag{32}$$

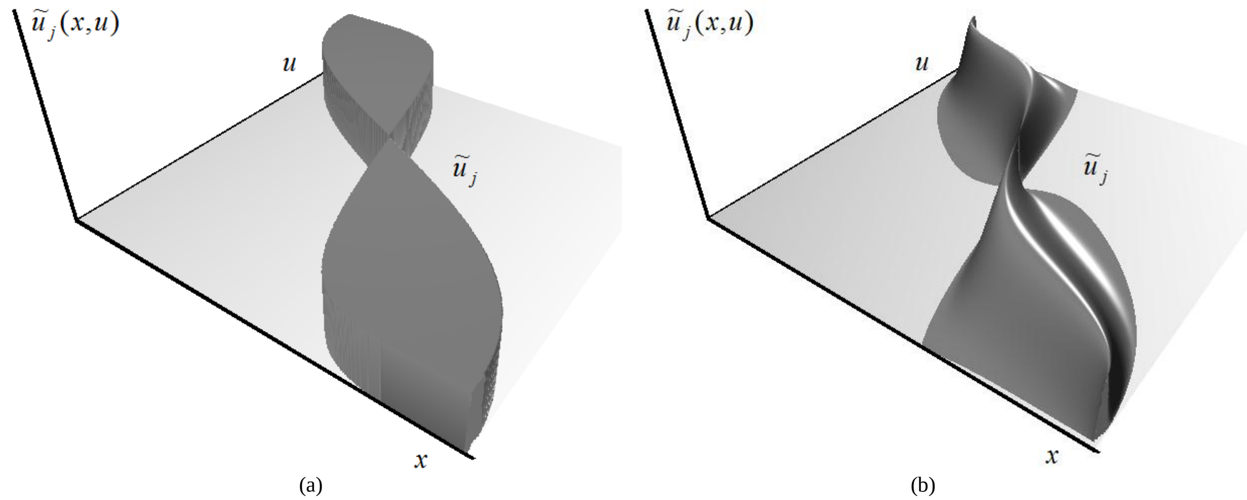


Fig. 4 The IT2 (a) and GT2 (b) FCM membership functions.

where

$$S_M(\alpha)=[s_M^L(\alpha),s_M^R(\alpha)] \tag{33}$$

The proposed GT2 FCM algorithm uses the linguistic fuzzifier M to construct secondary membership functions of the GT2 fuzzy partition matrix $\tilde{U}$. The degree of belonging of pattern $\mathbf{x}_i$ to cluster $\mathbf{v}_j$ can now be expressed using a membership grade $\tilde{u}_j(\mathbf{x}_i)$ expressed as T1 FS. Combining the T1 fuzzy memberships from all input patterns, the GT2 fuzzy membership function for cluster $\mathbf{v}_j$ can be obtained:

$$\tilde{u}_j=\sum_{\mathbf{x}_i\in X}\tilde{u}_j(\mathbf{x}_i) \tag{34}$$

The T1 fuzzy membership grades of individual patterns $\tilde{u}_j(\mathbf{x}_i)$ can be understood as a secondary membership functions of the GT2 cluster membership function $\tilde{u}_j$ sampled at the particular locations of patterns $\mathbf{x}_i$. The secondary membership function $\tilde{u}_j(\mathbf{x}_i)$ can be expressed using its α-*cuts* as follows:

$$\tilde{u}_j(\mathbf{x}_i)=\bigcup_{\alpha\in[0,1]}\alpha/S_{\tilde{u}_j}(\mathbf{x}_i|\alpha)=\bigcup_{\alpha\in[0,1]}\alpha/[s_{\tilde{u}_j}^L(\mathbf{x}_i|\alpha),s_{\tilde{u}_j}^R(\mathbf{x}_i|\alpha)] \tag{35}$$

By combining all α-*cuts* $S_{\tilde{u}_j}(\alpha)$ for a specific value of α over all input patterns, an α-*plane* of the GT2 cluster membership function $\tilde{u}_j$ can be obtained as:

$$\tilde{u}_j(\alpha)=\sum_{\mathbf{x}_i\in X}S_{\tilde{u}_j}(\mathbf{x}_i|\alpha) \tag{36}$$

The left and right boundaries $s_{\tilde{u}_j}^L(\mathbf{x}_i|\alpha)$ and $s_{\tilde{u}_j}^R(\mathbf{x}_i|\alpha)$ of α-*plane* $\tilde{u}_j(\alpha)$ at a specific position of pattern $\mathbf{x}_i$ can be computed by evaluating the interval fuzzy membership of pattern $\mathbf{x}_i$ to cluster $\mathbf{v}_j$ using the interval-valued fuzzifier $[s_M^L(\alpha),s_M^R(\alpha)]$. Note, that this interval-valued fuzzifier value is obtained from the α-*cuts* representation of the linguistic fuzzifier M as in (33). Hence:

$$s_{\tilde{u}_j}^R(\mathbf{x}_i|\alpha)=\max\left(\frac{1}{\sum_{l=1}^{c}\left(d_{ij}/d_{il}\right)^{2/(s_M^L(\alpha)-1)}},\ \frac{1}{\sum_{l=1}^{c}\left(d_{ij}/d_{il}\right)^{2/(s_M^R(\alpha)-1)}}\right) \tag{37}$$

$$s_{\tilde{u}_j}^L(\mathbf{x}_i|\alpha)=\min\left(\frac{1}{\sum_{l=1}^{c}\left(d_{ij}/d_{il}\right)^{2/(s_M^L(\alpha)-1)}},\ \frac{1}{\sum_{l=1}^{c}\left(d_{ij}/d_{il}\right)^{2/(s_M^R(\alpha)-1)}}\right) \tag{38}$$

Hence, the α-*cuts* of the linguistic fuzzifier M are used to construct α-*planes* of the GT2 cluster membership function $\tilde{u}_j$. A comparison of the constructed IT2 and the GT2 fuzzy membership functions is depicted in Fig. 4. Note that in Fig. 4(b) the GT2 FCM membership function was constructed using linguistic fuzzifier valued modeled as symmetrical Gaussian T1 FS.

B. Cluster Position Update

The cluster position update of the T1 FCM algorithm determines the new cluster locations based on the membership matrix U. In case of the introduced GT2 FCM algorithm, the proposed cluster position update method consists of first type-reducing the GT2 fuzzy cluster memberships into their T1 fuzzy centroids and then defuzzifying these centroids into the precise cluster positions.

The proposed algorithm is based on Liu's α-*planes* centroid theorem as expressed in (16) and (17). One of the fundamental ideas underlying the α-*plane* representation theorem for GT2 FSs is that each α-*plane* and its respective α-*level* set can be treated as IT2 FSs.

In case of the GT2 fuzzy membership function $\tilde{u}_j$ the fuzzy position of cluster $\tilde{\mathbf{v}}_j$ can be obtained as its centroid, which itself is a T1 fuzzy set. Hence:

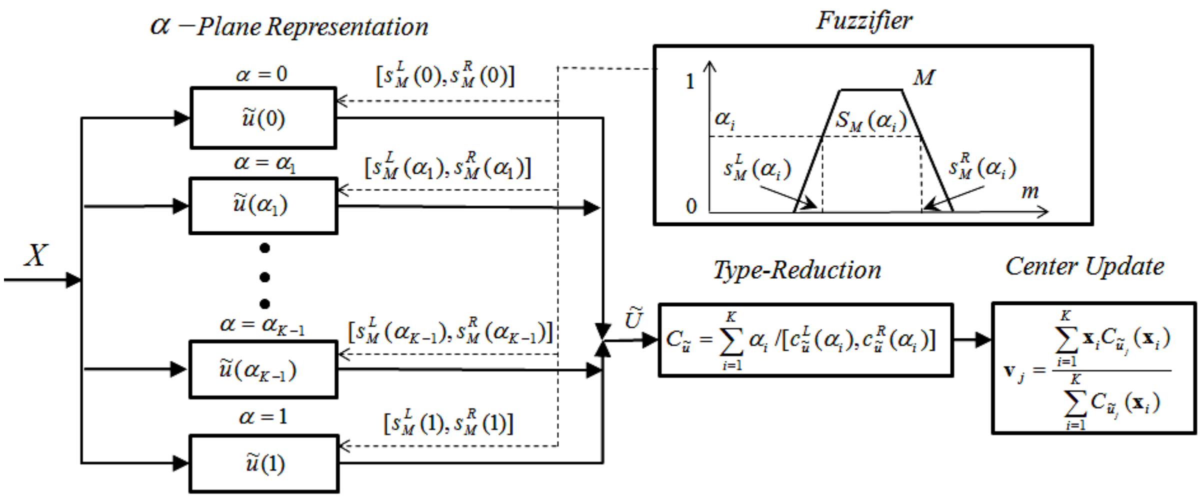


Fig. 5 Schematic view of the GT2 FCM algorithm.

$$\tilde{\mathbf{v}}_j = C_{\tilde{u}_j} = \sum_{\tilde{u}_j(x_1)\in J_{x_1}} \cdots \sum_{\tilde{u}_j(x_n)\in J_{xn}} \Big(f(\tilde{u}_j(x_1)) * \cdots * f(\tilde{u}_j(x_N)) \Big) \Big/ \frac{\sum_{i=1}^{n} \mathbf{x}_i \tilde{u}_j(\mathbf{x}_i)}{\sum_{i=1}^{n} \tilde{u}_j(\mathbf{x}_i)} \tag{39}$$

The computation of fuzzy position of cluster $\tilde{\mathbf{v}}_j$ resembles (6), were * denotes the selected t-norm operation (e.g. minimum).

According to Liu's theorem, centroid $C_{\tilde{u}_j}$ can be calculated as a weighted composition of the interval centroids of individual α *-planes*.

$$C_{\tilde{u}_j} = \bigcup_{\alpha\in[0,1]} \alpha \Big/ [c_{\tilde{u}_j}^L(\alpha), c_{\tilde{u}_j}^R(\alpha)] \tag{40}$$

The computation of individual interval centroids $[c_{\tilde{u}_j}^L(\alpha), c_{\tilde{u}_j}^R(\alpha)]$ is straightforward as it follows the well-established arithmetic of type-reduction for IT2 FSs. In the presented implementation, the independent application of the Enhanced Karnik-Mendel algorithms was used [31]-[33].

The precise cluster position is computed by defuzzifying the cluster centroid $C_{\tilde{u}_j}$:

$$\mathbf{v}_j = \frac{\sum_{i=1}^{K} \mathbf{y}_i C_{\tilde{u}_j}(\mathbf{y}_i)}{\sum_{i=1}^{K} C_{\tilde{u}_j}(\mathbf{y}_i)} \tag{41}$$

Here, K is the number of discretization steps of the domain of the centroid and $\mathbf{y}_i$ denotes the position vector of those discretized steps. The value of K is directly determined by the number of α *-planes* used. An illustration of the GT2 FCM algorithm is depicted in Fig. 5.

C. Hard-Thresholding for GT2 FCM Algorithm

In some applications it is desirable to determine the hard assignment of an input pattern $\mathbf{x}_i$ to the most representative cluster. The hard-partitioning rule of the GT2 FCM algorithm can be expressed as:

$$\begin{aligned} &\textbf{IF}\big(\tilde{u}_j(\mathbf{x}_i) > \tilde{u}_k(\mathbf{x}_i)\big),\ k = 1,\ldots,c, k \neq j \\ &\qquad \textbf{THEN}\ \ \mathbf{x}_i \text{ belongs to cluster } j \end{aligned} \tag{42}$$

However, the fuzzy membership grades $\tilde{u}_j(\mathbf{x}_i)$ of pattern $\mathbf{x}_i$ to different clusters are themselves T1 FSs and must be first defuzzified. The approach presented in [10] for the IT2 FCM algorithm could be extended from IT2 FSs to GT2 FSs via Liu's representation theorem. However, some deficiencies of the hard-partitioning approach proposed by Hwang and Rhee in [10] can be identified.

The previously proposed method, which was described in (29)-(31), calculates the left and right membership $u_j^L(\mathbf{x}_i)$ and $u_j^R(\mathbf{x}_i)$ of pattern $\mathbf{x}_i$ with respect to cluster j based on pattern's contribution either to the left or to the right cluster interval position in each dimension. Two problems can be identified as follows. Firstly, since the membership of pattern $\mathbf{x}_i$ to cluster j was calculated in the multidimensional space using the Euclidian distance norm, it seems redundant to separately aggregate identical membership values for each dimension. Secondly, the proposed method commonly results in values of the left membership $u_j^L(\mathbf{x}_i)$ being greater than the right membership $u_j^R(\mathbf{x}_i)$. The authors consider this as counterintuitive with respect to the basic principles of T2 fuzzy logic and thus propose a novel hard-partitioning scheme.

In the proposed GT2 FCM algorithm the hard-partitioning is performed based on the defuzzified value of the T1 fuzzy membership grade. Hence, the following rule can be applied:

$$\begin{aligned} &\textbf{IF}\big(c(\tilde{u}_j(\mathbf{x}_i)) > c(\tilde{u}_k(\mathbf{x}_i))\big),\ k = 1,\ldots,c, k \neq j \\ &\qquad \textbf{THEN}\ \mathbf{x}_i \text{ belongs to cluster } j \end{aligned} \tag{43}$$

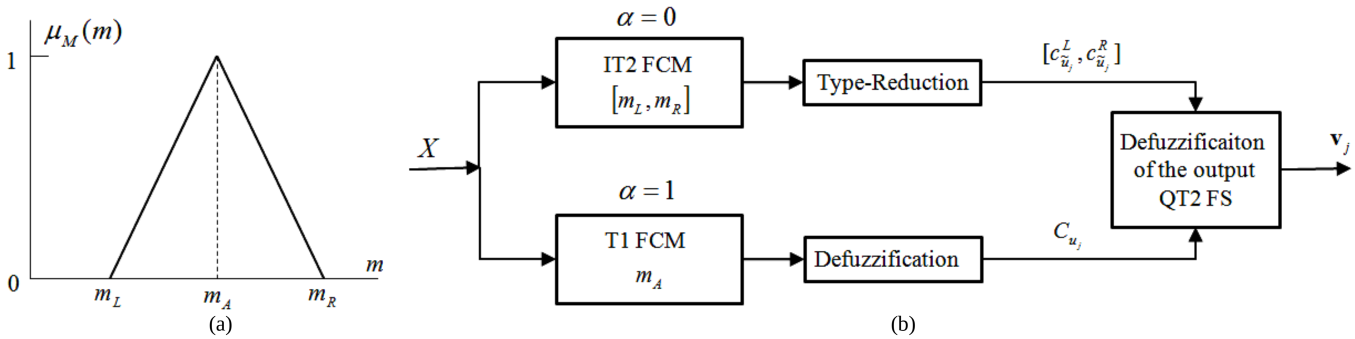


Fig. 6 Fuzzifier value M modeled as QT2 FS (a) and a schematic view of the QT2 FCM algorithm.

For the sake of completeness, the centroid of the T1 fuzzy membership grade $c(\tilde{u}_j(\mathbf{x}_i))$ can be computed as follows:

$$c(\tilde{u}_j(\mathbf{x}_i)) = \frac{\sum_{i=1}^{K} \mathbf{y}_i \tilde{u}_j(\mathbf{y}_i)}{\sum_{i=1}^{K} \tilde{u}_j(\mathbf{y}_i)} \quad (44)$$

Again, K is the number of discretization steps of the domain of the fuzzy cluster membership grade and $\mathbf{y}_i$ denotes the position vector of those discretized steps. The value of K is directly determined by the number of α*-planes* used.

D.Computational Complexity

Assume that the number of input patterns is denoted as n, number of clusters is c and the number of training iterations is I. Each update of a single element in the T1 FCM membership matrix of size $c \times n$ requires computation of mutual distances between particular input pattern and all other clusters as in (22). Hence, the asymptotical complexity of the T1 FCM algorithm can be summarized as $O(Inc^2)$.

The IT2 FCM algorithm uses interval-valued fuzzifier, which requires computation of the lower and the upper cluster membership functions. In addition, the EKM algorithm is used to compute the left and right cluster boundaries. It has been shown that the EKM algorithm converges super-exponentially fast, however its asymptotical time complexity is still $O(nm)$, where m is the number of iterations required. Since the number of iterations is typically very low, the complexity of the EKM algorithm is here approximated as $O(n)$ [42]. Hence, the asymptotical complexity of the IT2 FCM algorithm can be summarized as $O(I(2nc^2 + nc))$.

The GT2 FCM algorithm proceeds by computing the interval membership values and the interval cluster positions for each α*-plane*. When K α*-planes* are used the asymptotical complexity of the GT2 FCM algorithm becomes $O(IK(2nc^2 + nc))$. Hence, when compared to the IT2 FCM algorithms, the GT2 FCM algorithm linearly increases the computational complexity with the number of α*-planes* used.

V. Quasi Type-2 Fuzzy C-Means Algorithm

Several authors proposed the concept of QT2 FSs as an intermediate step between IT2 FSs and the full-blown GT2 FSs [43], [44]. Here, the GT2 FS is approximated by considering only two α*-planes* of each FS $\tilde{A}$, namely the bottom and top α*-planes* $\tilde{A}_0$ and $\tilde{A}_1$. Experimental studies demonstrated that especially in case of triangular secondary membership functions the QT2 FL systems compute close approximation of the GT2 FL systems. Similar idea is used here to develop the QT2 FCM algorithm.

The QT2 FCM algorithm calculates the partition of the input data set X into a set of clusters C via fusing the calculation of the T1 FCM and the IT2 FCM algorithms. Here, the geometry of the linguistic fuzzifier M is restricted only to triangular fuzzy membership functions. Hence, fuzzifier M can be described using three parameters $\{m_L, m_A, m_R\}$ encoding the base and the apex of the triangular membership function as depicted in Fig. 6(a) and subject to constraint $m_L \le m_A \le m_R$.

The QT2 FCM algorithm uses the T1 FCM algorithm as described in Section III.A with the fuzzifier value m_A and the IT2 FCM algorithm as described in Section III.B with the interval-valued fuzzifier [m_L, m_R]. The fuzzy position of cluster $\tilde{\mathbf{v}}_j$ for the QT2 FCM algorithm is then computed as the centroid of the fuzzy cluster membership function by fusing the singleton cluster position of the T1 FCM algorithm with the IT2 centroid of the IT2 FCM algorithm as follows:

$$\tilde{\mathbf{v}}_j = C_{\tilde{u}_j} = 0/c^L_{\tilde{u}_j}(0) + 1/C_{u_j}(1) + 0/c^R_{\tilde{u}_j}(0) \quad (45)$$

The precise position of cluster j can then be determined using the simplified defuzzification of the centroid [42]:

$$\mathbf{v}_j = \frac{1}{3}\left[c^L_{\tilde{u}_j}(0) + C_{u_j}(1) + c^R_{\tilde{u}_j}(0)\right] \quad (46)$$

The schematic of the QT2 FCM algorithm is depicted in Fig. 6(b).

VI. Experimental Results

This section presents the experimental testing of the proposed GT2 FCM algorithm. First, the extraction of patterns using GT2 fuzzy membership functions is demonstrated by visualizing the extracted uncertain cluster positions. Second, the performance of the GT2 FCM algorithm is compared against T1, IT2 and QT2 FCM algorithms on benchmark data sets with uncertain cluster position and densities. Finally, the GT2 FCM algorithm is applied to the task of pattern

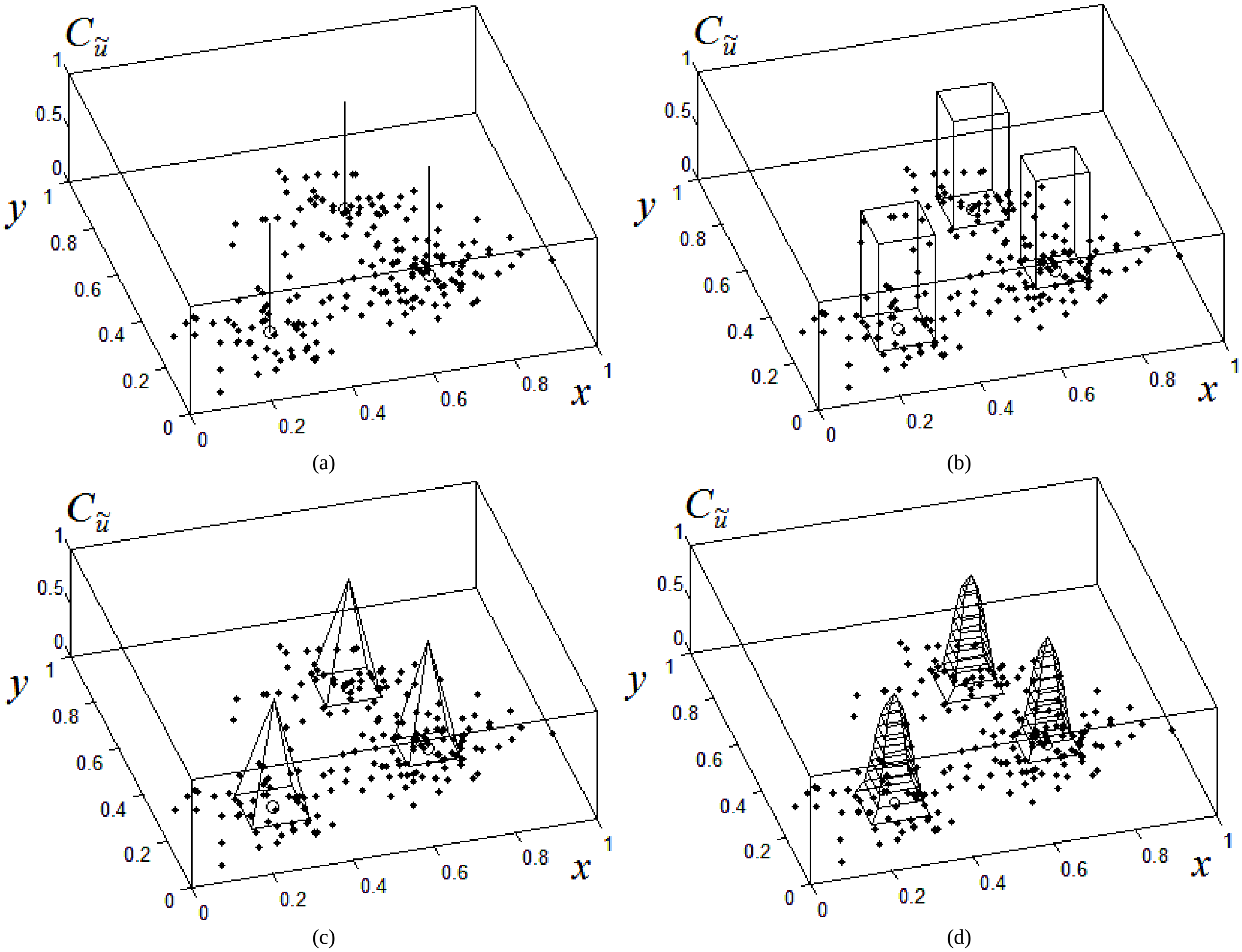


Fig. 7 Cluster position uncertainty for T1 FCM (a), IT2 FCM (b), QT2 FCM (c) and GT2 FCM (d) algorithms.

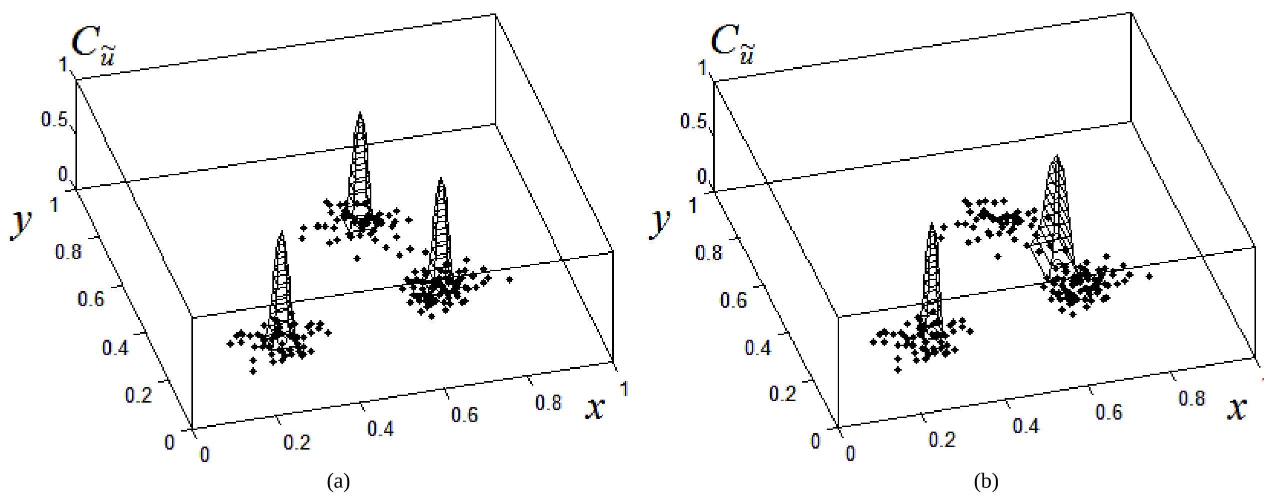


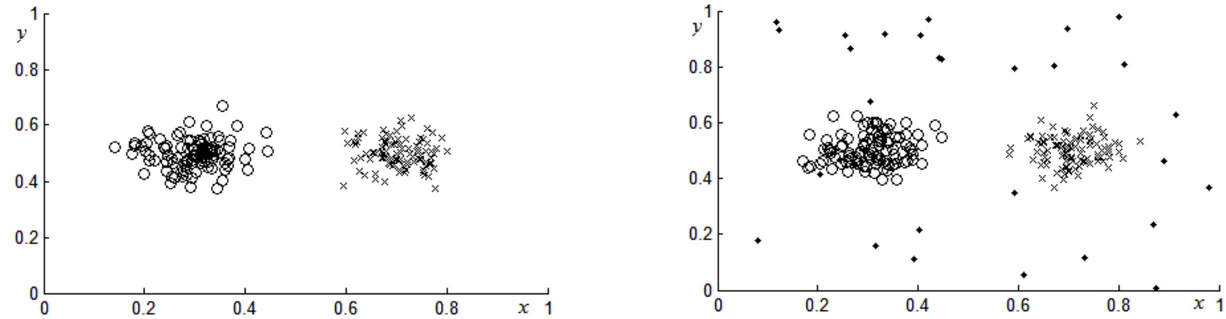
Fig. 8 GT2 FCM cluster position uncertainty for clusters with lower dispersion (a) and for insufficient number of clusters (b).

recognition in several well-known multi-dimensional benchmark data sets.

A. Extracting Uncertain Patterns

Experiment 1: Consider a problem of clustering 2D input data distribution composed of three Gaussian clusters. The cluster positions extracted using the T1 FCM algorithm with c = 3 and a real valued fuzzifier parameter are shown in Fig. 7(a). The third dimension in Fig. 7 denotes the uncertainty about the cluster position. Since T1 FCM algorithm does not provide any mechanism for handling uncertainty, the extracted cluster positions are certain and denoted as singletons in the 2D input space.

The IT2 FCM algorithm handles interval uncertainty in the used fuzzifier value. Through the algorithm proposed by Rhee and Hwang, this interval uncertainty is translated into interval position of the cluster centroids, as depicted in Fig. 7(b). Due to the interval fuzzy memberships, the precise cluster location



(a) (b)

Fig. 9 Two overlapping Gaussian clusters for cluster position estimation with no noise (a) and with 30 uniformly distributed noise points (b).

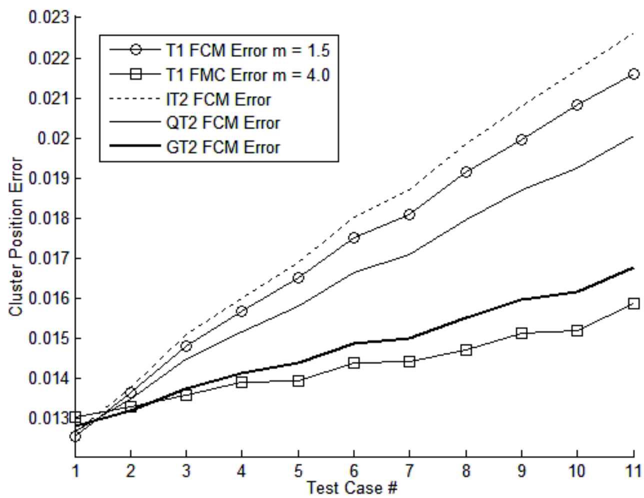


Fig. 10 Cluster position estimation error for different FCM algorithms.

is equally likely within the interval region. Thus it is computed as the center of the interval centroid.

The proposed GT2 FCM algorithm uses linguistic description of the fuzzifier value *M*. This parameter uncertainty transforms to T1 fuzzy centroids, which more accurately model the uncertainty about the cluster position using T1 FSs. The cluster position uncertainty for GT2 FCM algorithm with 10 α *-planes* is depicted in Fig. 7(d).

Finally, the proposed QT2 FCM algorithm clusters the input data using linguistic fuzzifier specified as a QT2 FS *M*. Thus this uncertainty is transformed into the uncertainty about cluster position modeled using the QT2 fuzzy centroid depicted in Fig. 7(c).

It should be noted here that the uncertain cluster positions visualized in Fig. 7 resemble the 3D terminal FCM prototypes obtained by clustering the fused heterogeneous fuzzy data in [5], [6]. However, the fundamental difference between both approaches is that the uncertainty in cluster positions in Fig. 7 is due to the uncertain fuzzy parameters of the fuzzy clustering algorithm, while in latter case, the uncertain cluster positions are due the uncertainty of the input data themselves.

Experiment 2: The uncertainty about cluster positions can also provide some vital clues about the structure of the input data or the quality of the obtained solution.

Fig. 8(a) shows uncertain cluster position extracted using the GT2 FCM algorithm from a 2D data set with three Gaussian distributions as in the previous experiment. The Gaussian distributions have identical means, but reduced standard deviations. It can be observed in Fig. 8(a), that as the

TABLE I

Relative Error Improvement for Different FCM Algorithms

FCM Algorithm	Test Case 1		Test Case 11	
	Error	Relative Improvement	Error	Relative Improvement
T1 FCM m = 1.5	0.0126	3.82%	0.0216	-
T1 FCM m = 4.0	0.0131	-	0.0159	26.39%
GT2 FCM	0.0128	2.29%	0.0168	22.22%

dispersion of the data points is substantially lower, the uncertainty about the position of the extracted patterns was also significantly reduced (compare to Fig. 7(d)).

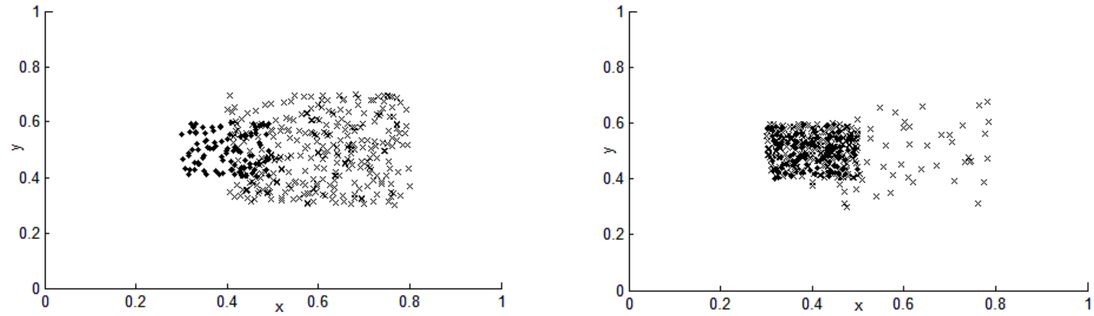
Fig. 8(b) illustrates the behavior of the cluster position uncertainty in case of a pre-selected inappropriate number of clusters when c = 2. It can be seen, that while one cluster was located successfully, the second cluster cannot cover the position of the two remaining Gaussian distributions and is thus significantly more uncertain, which is reflected in the geometry of the fuzzy centroid.

These two observations suggest that uncertainty measures of GT2 FSs [34] can provide important clues for cluster quality monitoring or as validation indexes.

B. Uncertain Cluster Position Estimation

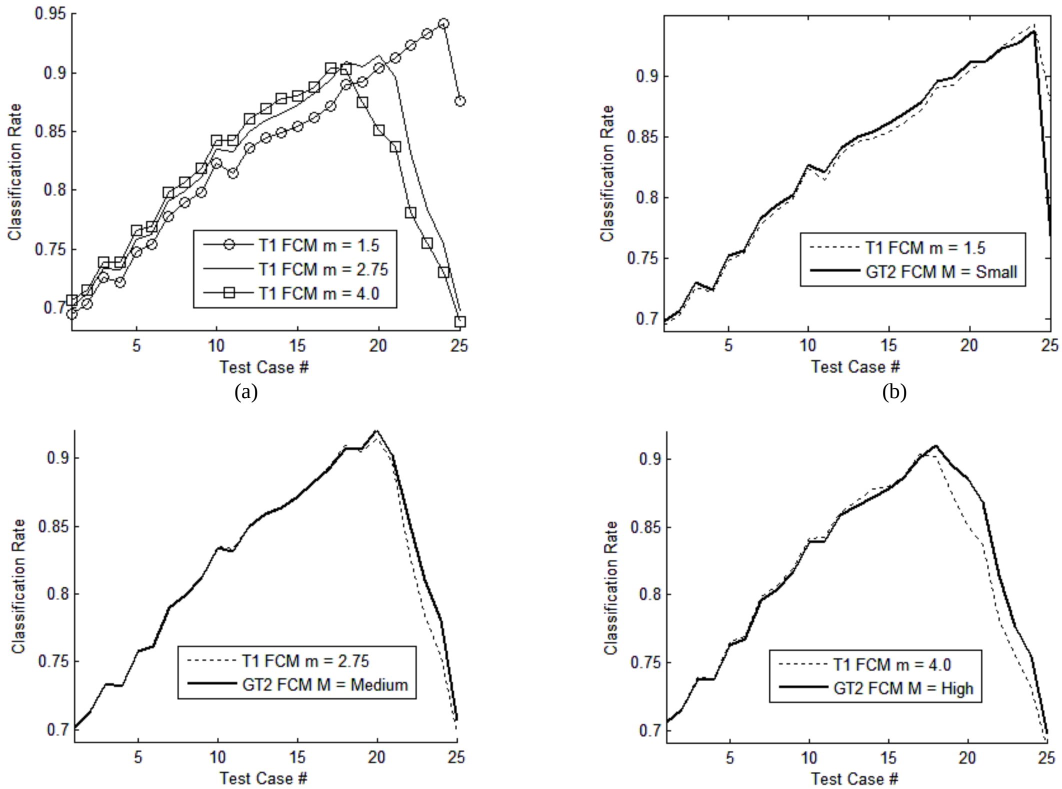
Experiment 3: Next, the performance of the GT2 FCM algorithm was tested on a problem of uncertain cluster position estimation. Here, two non-overlapping 2D clusters were created by sampling 100 data points from two Gaussian distributions with specified mean and standard deviation. The task of the FCM algorithms was to estimate the center of the Gaussian distribution via the clustering process.

First, a clean data set was considered as depicted in Fig. 9(a). Next, an increasing amount of uniformly distributed noise was added into the distribution all the way to 30 noisy points as shown in Fig. 9(b). Five algorithms were tested, 1) T1 FCM with m = 1.5, 2) T1 FCM with m = 4.0, 3) IT2 FCM with m_L = 1.5 and m_R = 4.0, 4) QT2 FCM with m_L = 1.5, m_R = 4.0 and m_A = 2.75, and 5) GT2 FCM with linguistic fuzzifier *M* =”*Medium*” modeled as symmetrical Gaussian T1 FS as shown in Fig. 3(b)). The experiment was repeated 500 times for each level of noise to reduce the variance of the results. The used quality measure was the sum of the position error of the estimated precise clusters’ positions. Fig. 10 shows the calculated position error for different test cases, where test case 1 was the input data without noise and test case 11 contained the maximum amount of 30 uniformly distributed noisy points.



(a) (b)

Fig. 11 Two uniform clusters for test case 1 (a) and test case 25 (b).



(a) (b) (c) (d)

Fig. 12 Classification rate of T1 FCM algorithms (a), comparison of T1 and GT2 FCM with small (b), medium (c) and high (d) values of the fuzzifier parameter.

The following observations can be made based on Fig. 10 and the summarized errors for test case 1 and test case 11 presented in Table I. The T1 FCM algorithm with smaller fuzzifier value performed the best for noise-less data (test case 1), while the T1 FCM algorithm with high fuzzifier value performed the worst. However, for data with high amount of noise (test case 11), the T1 FCM algorithm with high fuzzifier value significantly outperformed the T1 FCM algorithm with small fuzzifier value. The GT2 FCM algorithm combined the ability of T1 FCM algorithm with small fuzzifier value to correctly cluster noise-less and non-overlapping data with the ability of T1 FCM algorithm with large fuzzifier value to correctly cluster noisy over-lapping data sets. The GT2 FCM algorithm provided performance improvement regardless of the noise level. This is demonstrated in Table I, where the relative improvement denotes the improvement in determining the cluster position with respect to the worse of the two T1 FCM algorithms used. Hence, the GT2 FCM algorithm provides 2.29% lower error when compared to T1 FCM algorithm with higher fuzzifier value for test case 1 and 22.22% lower error when compared to T1 FCM algorithm with lower fuzzifier value for test case 11.

The IT2 FCM algorithm outperformed the worse T1 FCM algorithm for lower amounts of noise. Quite interestingly, its performance was the worst for increased amount of noise. This can be likely attributed to the fact that the IT2 FCM algorithm assigns uniform weight to all fuzzifier values in the large considered fuzzifier interval. The QT2 FCM algorithm performed in-between the GT2 and the IT2 FCM algorithms, ensuring steady robust performance and providing performance improvement when compared to the T1 FCM algorithm with inadequate choice of fuzzifier value.

C. Uncertain Pattern Recognition

Experiment 4: The following experiment analyzed the performance of the FCM algorithms on the task of uncertain pattern recognition. Here, two partially over-lapping 2D clusters were created by drawing a total of 400 points from two uniform distributions. The two clusters varied in the size and the density of the data points. Altogether 25 test cases were constructed, where the number of points (density) of each cluster was varied for each test case. The first test case is depicted in Fig. 11(a), where the smaller left cluster is composed of 100 points, while the larger right cluster contains 300 data points. The test case number 25 is depicted in Fig. 11(b), where the number of points in the left and right cluster is 350 and 50, respectively. The intermediate test cases were constructed by linearly increasing/decreasing the number of points in the left/right cluster. In this manner, the uncertainty in the pattern recognition task was due to cluster overlap, different cluster sizes and different cluster densities.

Six FCM algorithms with c=2 clusters were applied to the task of pattern recognition as follows: 1) T1 FCM with m = 1.5, 2) T1 FCM with m = 2.75, 3) T1 FCM with m = 4.0, 4) GT2 FCM with linguistic fuzzifier M="*Small*", 5) GT2 FCM with linguistic fuzzifier M="*Medium*", and 6) GT2 FCM with linguistic fuzzifier M="*High*" modeled according to Fig. 3(b). After the cluster positions were identified, the hard-partitioning process was applied to determine the final assignment of each data point to a specific cluster. The average classification rate was calculated over 10 runs for different distributions of points.

In general, the FCM algorithms were more successful in locating the smaller left cluster, hence significantly higher classification rates were achieved when this cluster contained more points. Fig. 12(a) plots the classification rates of all three T1 FCM algorithms. It can be observed that the T1 FCM algorithm with high fuzzifier value performs superior up to test case 17 and then its performance sharply deteriorates as the larger cluster becomes loosely defined by the decreasing number of data points. On the other hand, the T1 FCM algorithm with low fuzzifier value outperforms the other two algorithms for test cases 20-25, but its performance is worse for the first 20 test cases. This observation demonstrates that the optimal choice of the fuzzifier parameter m depends on the distribution of the input data points.

Figures 12(b)-12(d) compare the performance of the T1 FCM algorithms and the GT2 FCM algorithms with matching fuzzifier values (e.g. Fig. 12(b) shows T1 FCM with m=1.5 and GT2 FCM with M="*Small*"). It can be observed that the GT2 FCM algorithms balance the performance of the T1 FCM algorithms, resulting in more robust pattern recognition performance given the present uncertainties. Fig. 12(b) depicts the GT2 FCM algorithm with "*Small*" fuzzifier value offering increased classification rate for the first 22 test cases. The tradeoff is the performance deterioration for the last test cases. Fig. 12(c) and Fig. 12(d) show the improved performance of the GT2 FCM algorithms with "Medium" and "High" fuzzifier values, which is apparent as increased robustness of the algorithms to performance deterioration for later test cases.

In summary, using GT2 FCM algorithm can be observed to balance the performance of the T1 FCM algorithms since the GT2 cluster membership functions encapsulate many T1 fuzzy membership functions and combine their performance together.

TABLE II
SUMMARY OF MULTI-DIMENSIONAL DATA SETS

Data Set	Attributes	Number of Data Points	Classes
Iris	4	150	3
Wine	13	178	3
Pima Indians	8	768	2
Yeast	8	1,299	4
Shuttle	9	43,500	2
Magic	11	19,200	2

D. Pattern Recognition in Higher-Dimensional Data sets

Experiment 5: The performance of the proposed GT2 FCM algorithm was tested on the task of pattern recognition in multi-dimensional benchmark data sets obtained from the UCI Machine Learning repository [45]. Six data sets were selected, Iris, Wine, Pima Indians diabetes, Yeast, Statlog Shuttle and Magic Gamma Telescope data sets. The summary of the data sets is given in Table II. For the Yeast data set only the four most abundant classes have been used. Similarly, two classes have been created in the Shuttle data set by taking the most abundant data class (80% of data) as class 1 and the remaining less abundant data classes as class 2 (20% of data).

Identical FCM algorithms with the same parameter values were used as in *Experiment 3*. Individual FCM algorithms have been used to model the distribution of patterns in each class using c=3 clusters. The training process was considered to converge after 20 iterations of the FCM algorithms. The classification recognition rate was computed using the hard-partitioning scheme as described in Sections III and IV. To further accentuate the performance of the FCM algorithms in uncertain clustering, two additional sources of uncertainty that are common to real world pattern recognition tasks were considered. First, an insufficient number of training data points was used, by considering only 30% of the available input data for training and the remaining 70% for testing. Second, three test cases were constructed by considering clean input data and data with additional noise with SNR = 20db and SNR = 10db. In order to reduce the variance of the reported results, the experiment was repeated 10 times. Different set of training and testing data was selected for each experimental run. The classification rates are reported in Table III-V.

First, the following observation can be made by observing the performance of the T1 FCM algorithms. The optimal value of the fuzzifier parameter m for the T1 FCM algorithms varies for different data sets (e.g. clean Shuttle and Magic data sets in Table III) and different levels of noise in the same data set (e.g. clean and noisy SNR=20dB Shuttle data sets). Furthermore, the optimal value of fuzzifier parameter m also varies between the training and the testing data (e.g. Iris data set with noise SNR=20dB). Hence, T1 FCM algorithms with fuzzifier value m tuned based on the available training data might perform poorly on the previously unseen testing data. For ease of understanding, the performance of the worse of the two used T1 FCM algorithms has been highlighted in Table III-V.

Second, the performance of the GT2 FCM algorithm can be found to be less affected by the existing uncertainties in the data sets (e.g. unknown amount of noise or unknown optimal value of the fuzzifier parameter). By observing the results in

TABLE III
FCM CLASSIFICATION RATES FOR CLEAN DATA

FCM	Iris		Wine		Pima Indians		Yeast		Shuttle		Magic	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
T1 m=1.5	97.91%	**95.06**%	98.33%	95.10%	77.43%	**72.89**%	**60.21**%	**56.46**%	75.96%	75.92%	**74.81**%	**74.45**%
T1 m=4.0	**96.80**%	95.15%	**95.23**%	**94.54**%	**73.90**%	73.59%	60.24%	60.03%	**74.09**%	**74.02**%	77.44%	77.26%
IT2	97.28%	94.18%	96.08%	94.40%	75.30%	74.38%	54.99%	54.81%	75.98%	75.97%	75.97%	75.63%
QT2	97.53%	94.28%	95.43%	94.87%	75.46%	75.05%	56.04%	55.97%	74.89%	74.90%	77.58%	77.44%
GT2	97.49%	94.76%	94.68%	94.64%	75.44%	74.40%	58.67%	58.22%	74.75%	74.73%	78.24%	78.14%

TABLE IV
FCM CLASSIFICATION RATES FOR NOISE DATA SNR = 20DB

FCM	Iris		Wine		Pima Indians		Yeast		Shuttle		Magic	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
T1 m=1.5	91.38%	**81.54**%	96.00%	**90.11**%	71.84%	**66.75**%	45.10%	**39.45**%	**54.18**%	**54.19**%	**72.68**%	**72.32**%
T1 m=4.0	**89.68**%	86.07%	**94.25**%	90.94%	**70.10**%	67.94%	**44.97**%	42.15%	60.33%	60.36%	73.71%	73.75%
IT2	89.68%	85.00%	94.00%	89.78%	70.61%	68.10%	44.45%	42.21%	61.77%	61.79%	75.91%	75.58%
QT2	87.72%	83.90%	94.22%	90.57%	70.04%	68.36%	44.76%	42.42%	61.29%	61.32%	75.50%	75.48%
GT2	91.48%	83.96%	94.77%	90.63%	70.41%	68.14%	44.84%	42.36%	60.44%	60.48%	74.08%	74.04%

TABLE V
FCM CLASSIFICATION RATES FOR NOISE DATA SNR = 10DB

FCM	Iris		Wine		Pima Indians		Yeast		Shuttle		Magic	
	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test	Train	Test
T1 m=1.5	83.37%	**67.45**%	90.06%	**75.52**%	**62.18**%	**58.44**%	**35.57**%	**28.62**%	**50.81**%	**51.06**%	66.28%	65.97%
T1 m=4.0	**78.03**%	69.72%	**85.44**%	77.95%	64.71%	62.25%	37.53%	32.80%	50.85%	51.11%	**65.32**%	**65.13**%
IT2	76.44%	68.75%	84.18%	77.16%	62.70%	60.75%	35.64%	32.93%	50.91%	51.17%	65.85%	65.86%
QT2	77.02%	69.24%	84.38%	77.58%	63.89%	61.31%	36.13%	32.68%	50.88%	51.14%	65.69%	65.62%
GT2	80.42%	68.97%	84.81%	77.92%	64.58%	62.12%	37.20%	32.77%	50.84%	51.09%	65.34%	65.17%

Table III-V, the performance of the GT2 FCM algorithm on the unseen testing data can be found in most cases better than the T1 FCM algorithm with the inappropriate choice of the fuzzifier parameter (highlighted in bold). This fact can be attributed to the GT2 fuzzy membership functions combining several T1 fuzzy membership functions together and embedding them in the footprint of uncertainty of the GT2 cluster memberships. The performance of the IT2 and QT2 FCM algorithms can be found to be similar to the performance of GT2 FCM algorithm, but less consistent.

Hence, the experimental results obtained on the real-world higher-dimensional data sets demonstrate that when the algorithm parameters are uncertain and cannot be specified by real values (e.g. when insufficient training data are available or the data are corrupted by an unknown amount of noise), the GT2 FCM can deliver reasonable balanced performance despite these uncertainties.

E. Computational Complexity

Section IV.D presented the analysis of the asymptotical computational complexity of the different FCM algorithms. To further compare the computational requirements, the different algorithms have been implemented in C++ programming language and the actual computational time was measured on Dell Precision M4500, Intel Core i7 CPU Q720@1.60 GHz with 8.00 GB RAM and running Windows 7. The reported results have been averaged over 10 runs. Table VI compares the computational time for 50 iterations of the T1, IT2, QT2 and GT2 FCM algorithms with c=5, with 10 α-*planes* and applied to a 2D data sets of various sizes. The presented results demonstrate the linear increase of

TABLE VI
COMPUTATIONAL TIME [S] COMPARISON FOR DIFFERENT DATA SET SIZE

Data Set Size	T1 FCM	IT2 FCM	QT2 FCM	GT2 FCM
50	0.038	0.107	0.124	1.080
100	0.076	0.159	0.184	1.502
250	0.190	0.260	0.348	2.642
500	0.481	0.516	0.691	5.385
1000	0.756	1.040	1.484	10.685

TABLE VII
COMPUTATIONAL TIME [S] COMPARISON FOR DIFFERENT NUMBER OF CLUSTERS C

c	T1 FCM	IT2 FCM	QT2 FCM	GT2 FCM
1	0.022	0.057	0.070	0.476
2	0.049	0.074	0.097	0.747
5	0.222	0.341	0.487	3.048
10	0.770	1.217	1.436	11.294

computational time required by the GT2 FCM algorithm as a function of the number of α-*planes* used, when compared to the IT2 FCM algorithm. Furthermore, it can be also seen that the time of all algorithms is approximately linear with the size of the input data set.

From the theoretical analysis in Section V.D is apparent that the computational time is most strongly dependent on the number of clusters c. This behavior is verified in Table VII, when all four FCM algorithms have been applied to a data set with 300 data patterns and with 50 iterations and 10 α-*planes* for the GT2 FCM algorithm. It can be seen that the computational time increases faster than linear as a functions of c. Also the computational time of the GT2 FCM can be seen a number of α-*planes* longer than the IT2 FCM algorithm.

The presented results clearly demonstrate the tradeoff between the computational time required by the use of GT2 FSs and the improved uncertainty modeling capability of the GT2 FCM algorithm capable of computing with linguistically specified parameters.

VII. Conclusion

This paper presented a novel algorithm for uncertain pattern recognition using fuzzy clustering – the General Type-2 Fuzzy C-Means algorithm. The GT2 FCM algorithm manages the uncertainty associated with the appropriate choice of the fuzzifier parameter of the FCM algorithm. The GT2 FCM algorithm extends the previously published IT2 FCM algorithm via using the α-*planes* representation theorem. In the proposed framework the fuzziness of the input data can be expressed using linguistic terms such as "*Small*" or "*High*" modeled via T1 FSs. This linguistic fuzzifier is then used to construct the cluster membership functions as GT2 FSs. The modeled parameter uncertainty is then transformed into the uncertain fuzzy positions of the extracted clusters. A novel hard-thresholding method was also proposed for final input-cluster assignment. In addition, the Quasi-T2 FCM algorithm was introduced as a simplified version of the GT2 FCM method.

The advantages of the proposed GT2 FCM algorithm were demonstrated on several low-dimensional problems as well as on multi-dimensional benchmark data sets. It was shown that the GT2 FCM algorithm allows for extracting uncertain cluster position in the form of the centroids of the GT2 fuzzy cluster membership functions. In addition, the GT2 FCM algorithm was found to provide balanced performance in case of noisy data and clusters with varying density and weights. Performance evaluation on six multi-dimensional data sets showed that the GT2 FCM algorithm is capable of higher recognition rates on previously unseen testing data compared to T1 FCM algorithm with inappropriate choice of the fuzzifier value. The classification results of the GT2 FCM algorithm were found to be robust irrespective of the uncertainties in the pattern recognition task such as input noisy and insufficient amount of training data.

References

[1] J. Bezdek, "Fuzzy Mathematics in Pattern Recognition," PhD thesis, Applied Math. Center, Cornell University, Ithaca, USA, 1973.

[2] J. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum, 1981.

[3] J. Valente de Oliveira, W. Pedrycz (eds.), *Advances in Fuzzy Clustering and its Applications*, John Wiley & Sons, Ltd, 2007.

[4] F. Ch-H. Rhee, "Uncertain Fuzzy Clustering: Insights and Recommendations," in *IEEE Computational Intelligence Magazine*, vol. 2, issue: 1, pp. 44-56, Feb. 2007.

[5] R. Hathaway, J. C. Bezdek, W. Pedrycz, "A Parametric Model for Fusing Heterogeneous Fuzzy Data," in *IEEE Trans. on Fuzzy Systems*, vol. 4, no. 3, pp. 270-281, Aug. 1996.

[6] W. Pedrycz, J. C. Bezdek, R. J. Hathaway, G. W. Rogers, "Two Nonparametric Models for Fusing Heterogeneous Fuzzy Data," in *IEEE Trans. on Fuzzy Systems*, vol. 6, no. 3, pp. 411-425, Aug. 1998.

[7] W. Pedrycz, "Shadowed Sets: Representing and Processing Fuzzy Sets," in *IEEE Trans. on Systems, Man, and Cybernetics – Part B: Cybernetics*, vol. 28, no. 1, pp. 103-109, Feb. 1998.

[8] W. Pedrycz, "Interpretation of clusters in the framework of shadowed sets," in *Pattern Recognition Letters*, vol. 26, issue: 15, pp. 2439-2449, Nov. 2005.

[9] F. Rhee, C. Hwang, "A type-2 fuzzy C-means clustering algorithm," in *Proc. of 2001 Joint Conf. IFSA/NAFIPS*, pp. 1926-1919, July 2001.

[10] C. Hwang, F. Rhee, "Uncertain fuzzy clustering: interval type-2 fuzzy approach to C-means," in *IEEE Trans. on Fuzzy Systems*, vol. 15, no. 1, pp. 107-120, February 2007.

[11] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*, Prentice-Hall, Upper Saddle River, NJ, 2001.

[12] J. M. Mendel, "Advances in type-2 fuzzy sets and systems," in *Information Sciences*, vol. 177, pp. 84-110, 2007.

[13] H. Hagras, "Type-2 FLCs: A New Generation of Fuzzy Controllers," in *IEEE Computational Intelligence Magazine*, pp. 30-43, February 2007.

[14] N. N. Karnik, J. M. Mendel, "Type-2 Fuzzy Logic Systems," in *IEEE Trans. on Fuzzy Systems*, vol. 7, no. 6, pp. 643-658, December 1999.

[15] S. Coupland, R. John, "Geometric Type-1 and Type-2 Fuzzy Logic Systems," in *IEEE Trans. on Fuzzy Systems*, vol. 15, no. 1, pp. 3-15, February 2007.

[16] L. A. Zadeh, "The Concept of a Linguistic Variable and its Approximate Reasoning - II," in *Information Sciences*, No. 8, pp. 301-357, 1975.

[17] O. Linda, M. Manic, "Interval Type-2 Fuzzy Voter Design for Fault Tolerant Systems," in *Information Sciences*, vol. 181, issue: 14, pp. 2933-2950, July 2011.

[18] H. A. Hagras, "A Hierarchical Type-2 Fuzzy Logic Control Architecture for Autonomous Mobile Robots," in *IEEE Trans. Fuzzy Systems*, vol. 12, no. 4, pp. 524-539, 2004.

[19] J. Figueroa, J. Posada, J. Soriano, M. Melgarejo, S. Rojas, "A type-2 fuzzy logic controller for tracking mobile objects in the context of robotic soccer games," in *Proc. IEEE Intl' Conf. on Fuzzy Systems*, pp. 359-364, 2005.

[20] Ch. Lynch, H. Hagras, "Using Uncertainty Bounds in the Design of an Embedded Real-Time Type-2 Neuro-Fuzzy Speed Controller for Marine Diesel Engines," in *Proc. IEEE Intl' Conf. on Fuzzy Systems*, pp. 1446-1453, Vancouver, Canada, 2006.

[21] O. Linda, M. Manic, "Comparative Analysis of Type-1 and Type-2 Fuzzy Control in Context of Learning Behaviors for Mobile Robotics", in *Proc. IEEE IECON'10, 36th Annual Conference of the IEEE Industrial Electronics Society*, Glendale, Arizona, USA, Nov. 7-10, 2010.

[22] O. Linda, M. Manic, "Uncertainty-Robust Design of Interval Type-2 Fuzzy Logic Controller for Delta Parallel Robot," in *IEEE Transaction on Industrial Informatics*, vol. 7, no. 4, pp. 661-671, Nov. 2011.

[23] N. Karnik, J. M. Mendel, "Centroid of a type-2 fuzzy set," in *Information Sciences*, vol. 132, pp. 195-220, 2001.

[24] J. M. Mendel, R. John, F. Liu, "Interval Type-2 Fuzzy Logic Systems Made Simple," in *IEEE Trans. on Fuzzy Systems*, vol. 14, no. 6, pp. 808-821, 2006.

[25] H. B. Mitchell, "Pattern recognition using type-II fuzzy sets," in *Information Sciences*, vol. 170, issues: 2-4, pp. 409-418, February 2005.

[26] D. Wu, J. M. Mendel, "Uncertainty measures for interval type-2 fuzzy sets," in *Information Sciences*, vol. 177, issue: 23, pp. 5378-5393. December 2007.

[27] J. Zheng, L. Xie, Z.-Q. Liu, "Type-2 fuzzy Gaussian mixture models," in *Pattern Recognition*, vol. 41, issue: 12, pp. 3636-3643. Dec. 2008.

[28] C. Hwang, F. Rhee, "An interval type-2 fuzzy C spherical shells algorithm," in *Proc. 2004 Int. Conf. on Fuzzy Systems*, vol. 2, pp. 1117-1122, July 2004.

[29] F. Rhee, C. Hwang, "An interval type-2 fuzzy perceptron," in *Proc. 2002 Int. Conf. Fuzzy Systems*, vol. 2, pp. 1331-1335, May 2002.

[30] F. Rhee, C. Hwang, "An interval type-2 fuzzy K-nearest neighbor," n *Proc. 2003 Int. Conf. Fuzzy Systems*, vol. 2, pp. 802-807, May 2003.

[31] F. Liu, "An efficient centroid type-reduction strategy for general type-2 fuzzy logic system," in *Information Sciences*, vol. 178, pp. 2224-2236, 2008.

[32] J. M. Mendel, F. Liu, D. Zhai, "α-Plane Representation for Type-2 Fuzzy Sets: Theory and Applications," in *IEEE Transaction on Fuzzy Systems*, vol. 17, no. 5, pp. 1189-1207, October 2009.

[33] Ch. Wagner, H. Hagras, "Toward General Type-2 Fuzzy Logic Systems Based on zSlices," in *IEEE Transaction of Fuzzy Systems*, vol. 18, no. 4, pp. 637-660, August, 2010.

[34] D. Zhai, J. M. Mendel, "Uncertainty measures for general Type-2 fuzzy sets," in *Information Sciences*, vol. 181, issue: 3, pp. 503-518, February 2011.

[35] J. M. Mendel, R. I. John, "A fundamental decomposition of type-2 fuzzy sets," in *Proc. of IFSA World Congress and 20th NAFIPS International Conference*, pp.1896-1901, Canada, July, 2001.

[36] D. Zhai, J. M. Mendel, "Centroid of a General Type-2 Fuzzy Set Computed by Means of the Centroid-Flow Algorithm," in *Proc. WCCI 2010*, Barcelona, Spain, pp. 895-902, July, 2010.

[37] D. Zhai, J. M. Mendel, "Computing the Centroid of a General Type-2 Fuzzy Set by Means of the Centroid-Flow Algorithm," in *IEEE Trans. on Fuzzy Systems*, 2011, in press.

[38] H. Tahayori, A. G. B. Tettamanzi, G. D. Antoni, "Approximated Type-2 Fuzzy Set Operations," in *Proc. IEEE International Conference in Fuzzy Systems*, Canada, pp. 1910-1917, 2006.

[39] J. M. Mendel, "Comments on 'α-Plane Representation for Type-2 Fuzzy Sets: Theory and Applications,'" in *IEEE Trans. on Fuzzy Systems*, vol. 18, no. 1, pp. 229-230, February, 2010.

[40] C.Y. Yeh, W. H. Jeng, S. J. Lee, "An Enhanced Type-Reduction Algorithm for Type-2 Fuzzy Sets, " in *IEEE Trans. on Fuzzy Systems*, 2011, in press.

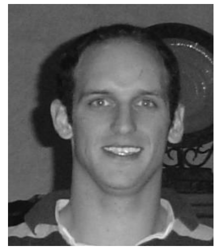
[41] O. Linda, M. Manic, "Monotone Centroid Flow Algorithm for Type-Reduction of General Type-2 Fuzzy Sets," in *IEEE Trans. on Fuzzy Systems*, in print, 2012.

[42] J. M. Mendel, F. Liu, "Super-Exponential Convergence of the Karnik-Mendel Algorithms for Computing the Centroid of an Interval Type-2 Fuzzy Set," in *IEEE Trans. on Fuzzy Systems*, vol. 15, no. 2, pp. 309-320, April, 2007.

[43] J. M. Mendel, F. Liu, "On new Quasi-type-2 fuzzy logic systems," in *Proc. Fuzz-IEEE*, Hong-Kong, China, pp. 354-360, June 2008.

[44] H. Hamrawi, S. Coupland, "Type-2 Fuzzy Arithmetic using Alpha-planes," in *Proc. IFSA-EUSFLAT 2009*, pp. 606-611, 2009.

[45] A. Frank, A. Asuncion, UCI Machine Learning Repository [http://archive.ics.uci.edu/ml], Irvine, CA: University of California, School of Informatics and Computer Science.



Ondrej Linda, (S'09) received his M.Sc. in Computer Graphics from Czech Technical University in Prague in 2010, and M.Sc. in Computational Intelligence from the University of Idaho at Idaho Falls in 2009. He received his B.Sc. in Electronic Engineering and Informatics from Czech Technical University in Prague in 2007. He is currently a Doctoral student at the University of Idaho in Idaho Falls. His research experience includes research assistant positions at Kansas State University and at the University of Idaho, and an internship with the Robotics Group at the Idaho National Laboratory. His fields of interest include machine learning, pattern recognition, intelligent control systems, data mining and computer graphics.

Dr. Milos Manic, (S'95-M'05-SM'06), Dr. Milos Manic, IEEE Senior Member, has been leading Computer Science Program at Idaho Falls and is a Director of Modern Heuristics Research Group. He received his Ph.D. degree in Computer Science from University of Idaho, Computer Science Dept. He received his M.S. and Dipl.Ing. in Electrical Engineering and Computer Science from the University of Nis, Faculty of Electronic Engineering, Serbia. He has over 20 years of academic and industrial experience, including an appointment at the ECE Dept. and Neuroscience program at University of Idaho. As university collaborator or principal investigator he lead number of research grants with the Idaho National Laboratory, NSF, EPSCoR, Dept. of Air Force, and Hewlett-Packard, in the area of data mining and computational intelligence applications in process control, network security and infrastructure protection. Dr. Manic is an Administrative Committee Member for the IEEE Industrial Electronics Society and member of several technical committees and boards of this Society. Dr. Manic has published over hundred refereed articles in international journals, books, and conferences.