



Article

Vector Fuzzy c-Spherical Shells (VFCSS) over Non-Crisp Numbers for Satellite Imaging

Iman Abaspor Kazerouni ^{1,2}, Hadi Mahdipour ^{3,*} , Gerard Dooly ² and Daniel Toal ² 

¹ CONFIRM Centre for SMART Manufacturing, University of Limerick, V94 C928 Limerick, Ireland; iman.abaspor@ul.ie

² Centre for Robotics and Intelligent Systems, Department of Electronics and Computer Engineering, University of Limerick, V94 T9PX Limerick, Ireland; Gerard.dooly@ul.ie (G.D.); Daniel.Toal@ul.ie (D.T.)

³ Chief Innovation Office, Sinenta Corp., La Cañada, 04120 Almeria, Spain

* Correspondence: mahdipourhadi@gmail.com

Abstract: The conventional fuzzy c-spherical shells (FCSS) clustering model is extended to cluster shells involving non-crisp numbers, in this paper. This is achieved by a vectorized representation of distance, between two non-crisp numbers like the crisp numbers case. Using the proposed clustering method, named vector fuzzy c-spherical shells (VFCSS), all crisp and non-crisp numbers can be clustered by the FCSS algorithm in a unique structure. Therefore, we can implement FCSS clustering over various types of numbers in a unique structure with only a few alterations in the details used in implementing each case. The relations of VFCSS applied to crisp and non-crisp (containing symbolic-interval, LR-type, TFN-type and TAN-type fuzzy) numbers are presented in this paper. Finally, simulation results are reported for VFCSS applied to synthetic LR-type fuzzy numbers; where the application of the proposed method in real life and in geomorphology science is illustrated by extracting the radii of circular agricultural fields using remotely sensed images and the results show better performance and lower cost computational complexity of the proposed method in comparison to conventional FCSS.

Keywords: imaging; fuzzy set; algorithms and clustering; satellite images



Citation: Abaspor Kazerouni, I.; Mahdipour, H.; Dooly, G.; Toal, D. Vector Fuzzy c-Spherical Shells (VFCSS) over Non-Crisp Numbers for Satellite Imaging. *Remote Sens.* **2021**, *13*, 4482. <https://doi.org/10.3390/rs13214482>

Academic Editors: Przemysław Kupidura and Joanna Pluto-Kossakowska

Received: 1 September 2021

Accepted: 3 November 2021

Published: 8 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Use of clustering models is a wide field of research in image processing and pattern recognition and they have been applied in different areas such as business, geology, engineering systems, etc. [1–3]. The objective of clustering is to explore the structure of the data and to partition the data set into groups with similar individuals. While the proposed method in this paper is objective-function based, generally clustering models may be objective-function, hierarchical or heuristic based. Hard clustering methods restrict each point of the data set to exactly one cluster [4]. Since Zadeh [5] presented fuzzy sets which introduce the idea of partial membership of belonging defined by a membership function, fuzzy clustering has been applied in various areas. After that, research in this field has been extended to apply fuzzy states to crisp cases. In the literature on fuzzy clustering, the fuzzy c-mean (FCM) clustering algorithms are the best-known methods [2,6]. Hadi et al. [7–9] presented some models to apply the FCM clustering model to crisp and non-crisp numbers. Using these models, the FCM can be stated in a single structure like the conventional FCM case. Sasha and Das [10] proposed an axiomatic extension of the possibilistic fuzzy clustering model in three directions: joint contribution function, choice of the dissimilarity measure, and the penalty function. Yu et al. [11] proposed a Suppressed Possibilistic C-Means (S-PCM) clustering model by creating a suppressed competitive learning approach into the Possibilistic C-means (PCM) to address the shortcoming of the PCM so as to develop the between-cluster relationships. Sasha and Das [12] presented a class of generalized FCM algorithms. They stated that the Consistent Membership-degree

Weighting Function (CMWF) based clustering scheme can be generalized to other FCM variants with different distance measures.

The fuzzy C-shells (FCS) presented by Dave is a novel algorithm in clustering spherical shells and it has been simplified to adaptive FCS (AFCS) for elliptical shells [13,14]. Krishnapuram et al. utilized the fuzzy C-spherical shells algorithm (FCSS) [15] to decrease the computational costs of FCS using an algebraic (non-Euclidean) distance measure. In this model, the prototypes can be determined directly, and the coupled nonlinear equations solution is not necessary. For two-dimensional (2D) cases, Man and Gath utilized the fuzzy C-rings (FCR) algorithm [16] for clustering ring data, while Gath and Hoory presented the fuzzy C-ellipses (FCE) model [17] for ellipse data. Krishnapuram et al. created the fuzzy C-quadric shells (FCQS) model [18], that detects quadrics like ellipses, circles, lines, or hyperbolas. The clustering models for detecting rectangular shells have been developed in the literature, such as the norm-induced shell prototypes (NISP) model by Bezdek et al. [19] and the fuzzy C-rectangular shell (FCRS) model by Hoepfner [20]. Wang [21] proposed a type of alternating optimization-based possibilistic c-shell model for clustering template-based shapes. Song et al. [22] proposed the information fuzzy C-spherical shells (IFCSS) model that addresses the intertwined robust fuzzy clustering problems of outlier detection. The model-based fuzzy c-shells clustering proposed by Hadi et al. [23] to cluster any shells (in 2-dimensions) that can be demonstrated by a fixed structure model in polar coordinates (but with arbitrary scale and centre for each shell). Song et al. [22] has been utilized the basic FCSS model for the clustering phase to reduce the difficulty of prototype initialization and minimize the number of hyper-parameters.

In this paper, the vector form of fuzzy c-Spherical Shells (VFCSS) over non-crisp numbers is presented. Using the proposed VFCSS, all crisp and non-crisp numbers can be clustered by the Fuzzy c-Spherical Shells algorithm. This approach can be applied on fuzzy and non-fuzzy numbers with unique structures. In this paper, the proposed VFCSS is applied over crisp numbers, symbolic numbers, fuzzy numbers (LR-type fuzzy numbers, TFN-type fuzzy numbers and normal fuzzy numbers). Simulations are applied on all introduced number classes and results are presented. Furthermore, the VFCC is utilized in the simulation results section to extract the radii of circular agriculture fields from remotely sensed images.

The rest of this paper is arranged as follows. The conventional FCSS clustering that is applied over crisp numbers is presented in Section 2. This section defines different types of non-crisp numbers and reviews the different metrics for them and presents the proposed VFCSS method. Simulation results of VFCSS applied over LR-type fuzzy numbers and satellite images (represented as symbolic-interval numbers) and discussion are presented in Section 3. Finally, the conclusion is presented in Section 4.

2. Methods

The goal of the FCSS algorithm is minimizing the following objective function regarding fuzzy membership u_{ik} and cluster centroid v_i [15]:

$$J_m(U, V, R) = \sum_{k=1}^n \sum_{i=1}^c u_{i,k}^m d^2(x_k, v_i, r_i) \quad (1)$$

$$d^2(x_k, v_i, r_i) = \left(\|x_k - v_i\|^2 - r_i^2 \right)^2 = ((x_k - v_i)^T (x_k - v_i) - r_i^2)^2 \quad (2)$$

where $X = \{x_1, x_2, \dots, x_n\}$ is a set of features vectors and k is a finite set of p -dimensional vectors over the crisp numbers $(x_k = [x_{k,1}, x_{k,2}, \dots, x_{k,p}]^T$ for, $k = 1, 2, \dots, n$). c is the

number of clusters, and $m > 1$ is the fuzziness index. The matrix $U = [u_{i,k}]_{c \times n}$ is called the fuzzy membership degree that has the following constraint:

$$\begin{aligned} u_{i,k} &\in [0, 1], \quad i = 1, 2, \dots, c, k = 1, 2, \dots, n \\ \sum_{i=1}^c u_{i,k} &= 1, \quad k = 1, 2, \dots, n \end{aligned} \quad (3)$$

where $u_{i,k}$ is the grad of membership of the k -th number to the i -th cluster. $V = \{v_1, v_2, \dots, v_c\}$ is the cluster prototypes set. $v_i = [v_{i,1}, v_{i,2}, \dots, v_{i,p}]^T \in R^p, i = 1, 2, \dots, c$ is the centre of the i -th cluster. $R = \{r_1, r_2, \dots, r_c\}$ is the set of cluster radii.

By creating a Lagrange function, we can minimize $J_m(U, V, R)$ subject to the constraints in (3) and conclude updated relations as follows:

$$L(V, R, U, \lambda) = \sum_{k=1}^n \sum_{i=1}^c u_{i,k}^m ((x_k - v_i)^T (x_k - v_i) - r_i^2)^2 - \sum_{k=1}^n \lambda_k \left(\sum_{i=1}^c u_{i,k} - 1 \right), \quad (4)$$

$$u_{i,k} = \left(\sum_{j=1}^c \left(\frac{d^2(x_k, v_i, r_i)}{d^2(x_k, v_j, r_j)} \right)^{\frac{1}{m-1}} \right)^{-1} = \left(\sum_{j=1}^c \left(\frac{((x_k - v_i)^T (x_k - v_i) - r_i^2)}{((x_k - v_j)^T (x_k - v_j) - r_j^2)} \right)^{\frac{2}{m-1}} \right)^{-1}, \text{ for } i = 1, 2, \dots, c \text{ and } k = 1, 2, \dots, n$$

$$\begin{cases} v_i = -\frac{1}{2} [q_{i,1}, q_{i,2}, \dots, q_{i,p}]^T \\ r_i = \sqrt{v_i^T v_i - q_{i,p+1}} \end{cases}, \text{ for } i = 1, 2, \dots, c. \quad (5a)$$

$$\text{Where : } \begin{cases} q_i = -\frac{1}{2} H_i^{-1} \omega_i & (5b) \\ H_i = \sum_{k=1}^n u_{i,k}^m g_k g_k^T & (5c) \\ \omega_i = \sum_{k=1}^n u_{i,k}^m s_k & (5d) \\ s_k = 2(x_k^T x_k) g_k, \text{ for } k = 1, 2, \dots, n & (5e) \\ g_k = [x_{k,1}, x_{k,2}, \dots, x_{k,p}, 1]^T, \text{ for } k = 1, 2, \dots, n & (5f) \end{cases}$$

In this part, we first define concepts of various types of non-crisp numbers. Then we review some dissimilarity definitions between two non-crisp numbers [24–33].

2.1. Definition of Various Non-Crisp Numbers

A symbolic number (SN) \tilde{X} is said to be a symbolic number where its membership function can be expressed as:

$$\mu_{\tilde{X}}(x) = \begin{cases} 1, \alpha_{\tilde{X}} \leq x \leq \beta_{\tilde{X}} \\ 0, \text{Otherwise} \end{cases}. \quad (6)$$

We denote a symbolic number (SN) \tilde{X} with its start point ($\alpha_{\tilde{X}}$) and its end point ($\beta_{\tilde{X}}$) with $\tilde{X} = (\alpha_{\tilde{X}}, \beta_{\tilde{X}})_{SN}$.

Let L (and R) be decreasing, shape functions from R^+ to $[0, 1]$ with $0 \leq L(x) \leq 1$ for all $x > 0$; $L(x) = 1$ for $x = 0$; $L(x) = 0$ for all $x \geq 1$. A fuzzy number \tilde{X} is called LR-type if shape functions L and R and four parameters $(m_{1\tilde{X}}, m_{2\tilde{X}}) \in R^2, (\alpha_{\tilde{X}}, \beta_{\tilde{X}}) \in R^{+2}$ exist and the membership function of \tilde{X} is as follows:

$$\mu_{\tilde{X}}(x) = \begin{cases} L\left(\frac{m_{1\tilde{X}} - x}{\alpha_{\tilde{X}}}\right), m_{1\tilde{X}} \geq x \\ 1, m_{1\tilde{X}} \leq x \leq m_{2\tilde{X}} \\ R\left(\frac{x - m_{2\tilde{X}}}{\beta_{\tilde{X}}}\right), m_{2\tilde{X}} \leq x \end{cases} \quad (7)$$

where $\alpha_{\tilde{X}} > 0, \beta_{\tilde{X}} > 0$ are called the left and right spreads respectively. Symbolically, \tilde{X} is denoted by $(m_{1\tilde{X}}, m_{2\tilde{X}}, \alpha_{\tilde{X}}, \beta_{\tilde{X}})_{LR}$.

Let (TFN) \tilde{X} be a trapezoidal fuzzy number. The parameterization of \tilde{X} is denoted by $\tilde{X} = (x_{1\tilde{X}}, x_{2\tilde{X}}, x_{3\tilde{X}}, x_{4\tilde{X}})_{TFN}$ where $x_{1\tilde{X}}, x_{2\tilde{X}}, x_{3\tilde{X}}$ and $x_{4\tilde{X}}$ are called the center, inner diameter, left outer radius and right outer radius respectively. The TFN \tilde{X} is demonstrated in Figure 1.

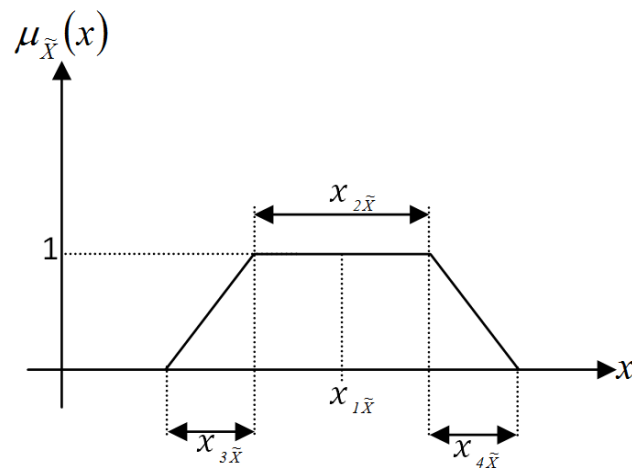


Figure 1. Parameterization of TFN.

Let (TAN) \tilde{X} be a triangular fuzzy number. The expressed membership function for $\tilde{X} = (x_{\tilde{X}}^L, x_{\tilde{X}}^C, x_{\tilde{X}}^R)_{TAN}$ is as follows:

$$\mu_{\tilde{X}}(x) = \begin{cases} 0, & \text{for } x \leq x^L \\ \frac{x - x_{\tilde{X}}^L}{x_{\tilde{X}}^C - x_{\tilde{X}}^L}, & \text{for } x^L \leq x \leq x^C \\ \frac{x_{\tilde{X}}^R - x}{x_{\tilde{X}}^R - x_{\tilde{X}}^C}, & \text{for } x^C \leq x \leq x^R \\ 0, & \text{for } x \geq x^R \end{cases} \quad (8)$$

We will present various metrics of papers [24–33].

2.2. Various Metrics for Symbolic Numbers

Suppose \tilde{x}_k and \tilde{v}_i are two symbolic numbers in p dimensions space, $\tilde{x}_k = \{\tilde{x}_{k,1}, \tilde{x}_{k,2}, \dots, \tilde{x}_{k,p}\}$, $\tilde{x}_{k,j} = (\alpha_{\tilde{x}_{k,j}}, \beta_{\tilde{x}_{k,j}})_{SN}$ for $j = 1, 2, \dots, p$ and $\tilde{v}_i = \{\tilde{v}_{i,1}, \tilde{v}_{i,2}, \dots, \tilde{v}_{i,p}\}$, $\tilde{v}_{i,j} = (\alpha_{\tilde{v}_{i,j}}, \beta_{\tilde{v}_{i,j}})_{SN}$ for $j = 1, 2, \dots, p$.

Dissimilarity defined for symbolic numbers in [24] is as follows:

$$d^2(\tilde{x}_k, \tilde{v}_i) = \sum_{j=1}^p \left((1-z)^2 (\alpha_{\tilde{x}_{k,j}} - \alpha_{\tilde{v}_{i,j}})^2 + z^2 (\beta_{\tilde{x}_{k,j}} - \beta_{\tilde{v}_{i,j}})^2 \right) \quad (9)$$

where $0 \leq z \leq 0.5$ is center distance weight. In [25,26], dissimilarity defined for symbolic numbers can be obtained from the following equation:

$$d^2(\tilde{x}_k, \tilde{v}_i) = \sum_{j=1}^p \gamma_j \left((\alpha_{\tilde{x}_{k,j}} - \alpha_{\tilde{v}_{i,j}})^2 + (\beta_{\tilde{x}_{k,j}} - \beta_{\tilde{v}_{i,j}})^2 \right) \quad (10)$$

where γ_j is weight vector and $\Pi_{j=1}^p \gamma_j = 1$. In the section Allocation step: definition of the best partition of [25,26], dissimilarity is defined as follows:

$$d^2(\tilde{x}_k, \tilde{v}_i) = \sum_{j=1}^p \gamma_{i,j} \left(\begin{array}{c} (\alpha_{\tilde{x}_{k,j}} - \alpha_{\tilde{v}_{i,j}})^2 \\ + (\beta_{\tilde{x}_{k,j}} - \beta_{\tilde{v}_{i,j}})^2 \end{array} \right) \quad (11)$$

where $\Pi_{j=1}^p \gamma_{i,j} = 1$. The last dissimilarity defined for symbolic numbers is suggested in [27] as follows:

$$d^2(\tilde{x}_k, \tilde{v}_i) = \sum_{j=1}^p \left(\begin{array}{c} (\alpha_{\tilde{x}_{k,j}} - \alpha_{\tilde{v}_{i,j}})^2 \\ + (\beta_{\tilde{x}_{k,j}} - \beta_{\tilde{v}_{i,j}})^2 \end{array} \right) \quad (12)$$

2.3. Various Metrics for Fuzzy Numbers

Suppose we have two LR-type fuzzy numbers \tilde{x}_k and \tilde{v}_i in a p dimensions space, $\tilde{x}_k = \{\tilde{x}_{k,1}, \tilde{x}_{k,2}, \dots, \tilde{x}_{k,p}\}$, $\tilde{x}_{k,j} = (m_{1\tilde{x}_{k,j}}, m_{2\tilde{x}_{k,j}}, \alpha_{\tilde{x}_{k,j}}, \beta_{\tilde{x}_{k,j}})_{LR}$ for $j = 1, 2, \dots, p$ and $\tilde{v}_i = \{\tilde{v}_{i,1}, \tilde{v}_{i,2}, \dots, \tilde{v}_{i,p}\}$, $\tilde{v}_{i,j} = (m_{1\tilde{v}_{i,j}}, m_{2\tilde{v}_{i,j}}, \alpha_{\tilde{v}_{i,j}}, \beta_{\tilde{v}_{i,j}})_{LR}$ for $j = 1, 2, \dots, p$. Yang et al. [29,30,32] expressed the metric $d(\tilde{x}_k, \tilde{v}_i)$ with the following definition:

$$d^2(\tilde{x}_k, \tilde{v}_i) = \sum_{j=1}^n d^2(\tilde{x}_{k,j}, \tilde{v}_{i,j})$$

$$d^2(\tilde{x}_{k,j}, \tilde{v}_{i,j}) = (m_{1\tilde{x}_{k,j}} - m_{1\tilde{v}_{i,j}})^2 + (m_{2\tilde{x}_{k,j}} - m_{2\tilde{v}_{i,j}})^2 +$$

$$\left((m_{1\tilde{x}_{k,j}} - l\alpha_{\tilde{x}_{k,j}}) - (m_{1\tilde{v}_{i,j}} - l\alpha_{\tilde{v}_{i,j}}) \right)^2 + \left((m_{2\tilde{x}_{k,j}} + r\beta_{\tilde{x}_{k,j}}) - (m_{2\tilde{v}_{i,j}} + r\beta_{\tilde{v}_{i,j}}) \right)^2 \quad (13)$$

where $l = \int_0^1 L^{-1}(w)dw$ and $r = \int_0^1 R^{-1}(w)dw$.

In the TFN numbers case, the Yang distance definition for two TFN numbers \tilde{x}_k and \tilde{v}_i in p dimensions space, $\tilde{x}_k = \{\tilde{x}_{k,1}, \tilde{x}_{k,2}, \dots, \tilde{x}_{k,p}\}$, $\tilde{x}_{k,j} = (x_{1\tilde{x}_{k,j}}, x_{2\tilde{x}_{k,j}}, x_{3\tilde{x}_{k,j}}, x_{4\tilde{x}_{k,j}})_{TFN}$ for $j = 1, 2, \dots, p$ and $\tilde{v}_i = \{\tilde{v}_{i,1}, \tilde{v}_{i,2}, \dots, \tilde{v}_{i,p}\}$, $\tilde{v}_{i,j} = (v_{1\tilde{v}_{i,j}}, v_{2\tilde{v}_{i,j}}, v_{3\tilde{v}_{i,j}}, v_{4\tilde{v}_{i,j}})_{TFN}$ for $j = 1, 2, \dots, p$, is obtained from (13) by setting $l = r = \frac{1}{2}$, $m_1 = \frac{2x_1 - x_2}{2}$, $m_2 = \frac{2x_1 + x_2}{2}$, $\alpha = x_3$ and $\beta = x_4$ as follows:

$$d^2(\tilde{x}_k, \tilde{v}_i) = \sum_{j=1}^p d^2(\tilde{x}_{k,j}, \tilde{v}_{i,j})$$

$$d^2(\tilde{x}_{k,j}, \tilde{v}_{i,j}) = \left(\frac{2x_{1\tilde{x}_{k,j}} - x_{2\tilde{x}_{k,j}}}{2} - \frac{2v_{1\tilde{v}_{i,j}} - v_{2\tilde{v}_{i,j}}}{2} \right)^2$$

$$+ \left(\left(\frac{2x_{1\tilde{x}_{k,j}} - x_{2\tilde{x}_{k,j}}}{2} - \frac{1}{2}x_{3\tilde{x}_{k,j}} \right) - \left(\frac{2v_{1\tilde{v}_{i,j}} - v_{2\tilde{v}_{i,j}}}{2} - \frac{1}{2}v_{3\tilde{v}_{i,j}} \right) \right)^2 + \left(\left(\frac{2x_{1\tilde{x}_{k,j}} + x_{2\tilde{x}_{k,j}}}{2} + \frac{1}{2}x_{4\tilde{x}_{k,j}} \right) - \left(\frac{2v_{1\tilde{v}_{i,j}} + v_{2\tilde{v}_{i,j}}}{2} + \frac{1}{2}v_{4\tilde{v}_{i,j}} \right) \right)^2 \quad (14)$$

Dissimilarity defined for two TFN numbers \tilde{x}_k, \tilde{v}_i in [28] is as follows:

$$d^2(\tilde{x}_k, \tilde{v}_i) = \sum_{j=1}^p \left(\begin{array}{c} (x_{1\tilde{x}_{k,j}} - v_{1\tilde{v}_{i,j}})^2 + (x_{2\tilde{x}_{k,j}} - v_{2\tilde{v}_{i,j}})^2 \\ + (x_{3\tilde{x}_{k,j}} - v_{3\tilde{v}_{i,j}})^2 + (x_{4\tilde{x}_{k,j}} - v_{4\tilde{v}_{i,j}})^2 \end{array} \right) \quad (15)$$

In the next state for LR-type fuzzy numbers suppose $m_{1\tilde{x}_{k,j}} = m_{2\tilde{x}_{k,j}} = m_{\tilde{x}_{k,j}}$ and $m_{1\tilde{v}_{i,j}} = m_{2\tilde{v}_{i,j}} = m_{\tilde{v}_{i,j}}$. The Yang distance definition for this state is as follows:

$$d^2(\tilde{x}_k, \tilde{v}_i) = \sum_{j=1}^p \left[\begin{aligned} & \left(m_{\tilde{x}_{k,j}} - m_{\tilde{v}_{i,j}} \right)^2 \\ & + \left(\left(m_{\tilde{x}_{k,j}} - l\alpha_{\tilde{x}_{k,j}} \right) - \left(m_{\tilde{v}_{i,j}} - l\alpha_{\tilde{v}_{i,j}} \right) \right)^2 \\ & + \left(\begin{aligned} & \left(m_{\tilde{x}_{k,j}} + r\beta_{\tilde{x}_{k,j}} \right) \\ & - \left(m_{\tilde{v}_{i,j}} + r\beta_{\tilde{v}_{i,j}} \right) \end{aligned} \right)^2 \end{aligned} \right] \quad (16)$$

To achieve the robust clustering method versus noisy input numbers, Yang et al. in [31] define an exponential based metric. Dissimilarity defined for LR-type fuzzy number \tilde{x} and prototype \tilde{v} in [31] is as follows:

$$d^2(\tilde{x}_k, \tilde{v}_i) = 1 - \exp(-bd^2(\tilde{x}_k, \tilde{v}_i)) \quad (17)$$

where $d^2(\tilde{x}_k, \tilde{v}_i)$ is obtained from (16) and b is a fixed coefficient [see [31]].

In [33] Rong et al. offer two metrics for triangular fuzzy numbers. Dissimilarity defined for triangular-type fuzzy numbers \tilde{x}_k and \tilde{v}_i in p dimensions space, $\tilde{x}_k = \{\tilde{x}_{k,1}, \tilde{x}_{k,2}, \dots, \tilde{x}_{k,p}\}$, $\tilde{x}_{k,j} = (x_{k,j}^L, x_{k,j}^C, x_{k,j}^R)_{TAN}$ for $j = 1, 2, \dots, p$ and $\tilde{v}_i = \{\tilde{v}_{i,1}, \tilde{v}_{i,2}, \dots, \tilde{v}_{i,p}\}$, $v_{i,j} = (v_{i,j}^L, v_{i,j}^C, v_{i,j}^R)_{TAN}$ for $j = 1, 2, \dots, p$, is as follows:

$$\begin{aligned} d^2(\tilde{x}_{k,j}, \tilde{v}_{i,j}) &= \frac{\alpha_1}{6} (x_{k,j}^L - v_{i,j}^L)^2 + \left(\frac{\alpha_1}{6} + \alpha_2 + \frac{\alpha_3}{6} \right) (x_{k,j}^C - v_{i,j}^C)^2 + \frac{\alpha_3}{6} (x_{k,j}^R - v_{i,j}^R)^2 \\ &+ \frac{\alpha_1}{6} \left((x_{k,j}^L + x_{k,j}^C) - (v_{i,j}^L + v_{i,j}^C) \right)^2 + \frac{\alpha_3}{6} \left((x_{k,j}^R + x_{k,j}^C) - (v_{i,j}^R + v_{i,j}^C) \right)^2 \\ d_1^2(\tilde{x}_k, \tilde{v}_i) &= \sum_{j=1}^p d^2(\tilde{x}_{k,j}, \tilde{v}_{i,j}) (18-1) \\ d_2^2(\tilde{x}_k, \tilde{v}_i) &= 1 - \exp(-b.d_1^2(\tilde{x}_k, \tilde{v}_i)) (18-2) \end{aligned} \quad (18)$$

where α_1, α_2 and α_3 are positive, fixed and known coefficients while $\alpha_1 + \alpha_2 + \alpha_3 = 1$ and b is a fixed coefficient also.

The main key point in the proposed Vector Fuzzy c-Spherical Shells (VFCSS) Clustering is how to define crisp vectors from parameters of affected numbers. According to any used metric, we define a crisp vectors set $Y = \{y_1, y_2, \dots, y_n\}$ a corresponding input numbers set (\tilde{X} for non-crisp numbers and X for crisp numbers) and $W = \{w_1, w_2, \dots, w_c\}$ and a corresponding centers set (\tilde{V} for non-crisp numbers and V for crisp numbers). These definitions allow us to demonstrate a distance and a Lagrange function in the proposed VFCSS as follows:

$$d^2 = \left(\|y_k - w_i\|^2 - r_i^2 \right)^2 = ((y_k - w_i)^T (y_k - w_i) - r_i^2)^2 \quad (19)$$

$$L(W, U, \lambda) = \sum_{k=1}^n \sum_{i=1}^c u_{i,k}^m ((y_k - w_i)^T (y_k - w_i) - r_i^2)^2 + \sum_{k=1}^n \lambda_k \left(\sum_{i=1}^c u_{i,k} - 1 \right) \quad (20)$$

By this definition, similar to the crisp case and (5), the membership value of $u_{i,k}$ can update from the next equation in the proposed VFCSS:

$$u_{i,k} = \left(\sum_{j=1}^c \left(\frac{d^2(y_k, w_i, r_i)}{d^2(y_k, w_j, r_j)} \right)^{\frac{1}{m-1}} \right)^{-1} = \left(\sum_{j=1}^c \left(\frac{((y_k - w_i)^T (y_k - w_i) - r_i^2)}{((y_k - w_j)^T (y_k - w_j) - r_j^2)} \right)^{\frac{2}{m-1}} \right)^{-1} \quad (21)$$

Let the independent parameters of the i th center (except radius r_i) be denoted by the set $\{t_i\}$. After creating the Lagrange function (20), we must calculate the derivation of the

Lagrange function with respect to parameters t_i, r_i and set the resulting equation equal to zero for all t_i 's as follows:

$$\left. \begin{aligned} \frac{\partial L}{\partial t_i} = \frac{\partial L}{\partial w_i} \cdot \frac{\partial w_i^T}{\partial t_i} &= 0, \text{ for all } t_i \\ \frac{\partial L}{\partial r_i} = \frac{\partial L}{\partial w_i} \cdot \frac{\partial w_i^T}{\partial r_i} &= 0 \end{aligned} \right\} \text{ for } i = 1, 2, \dots, c \quad (22)$$

As t_i s are independent and we have $\frac{\partial L}{\partial w_i} \cdot \frac{\partial w_i^T}{\partial t_i} = 0$ for all t_i s, furthermore $\frac{\partial L}{\partial w_i}$ is fixed while $\frac{\partial w_i^T}{\partial t_i}$ is changing because of changing the metric (in fixed t_i), therefore from (22) we can conclude that $\frac{\partial L}{\partial w_i} = 0$. Suppose the vectors $\{w_i\}_{i=1}^c$ and $\{y_k\}_{k=1}^n$ are \hat{p} -dimensional, then we can extract the updating relation of w_i and r_i similar to v_i and r_i in the crisp numbers case as follows:

$$\begin{cases} w_i = -\frac{1}{2}[q_{i,1}, q_{i,2}, \dots, q_{i,\hat{p}}]^T \\ r_i = \sqrt{w_i^T w_i - q_{i,\hat{p}+1}} \end{cases}, \text{ for } i = 1, 2, \dots, c \quad (23a)$$

$$\text{Where : } \begin{cases} q_i = -\frac{1}{2}H_i^{-1}\omega_i & (23b) \\ H_i = \sum_{k=1}^n u_{i,k}^m g_k g_k^T & (23c) \\ \omega_i = \sum_{k=1}^n u_{i,k}^m s_k & (23d) \\ s_k = 2(y_k^T y_k)g_k, \text{ for } k = 1, 2, \dots, n & (23e) \\ g_k = [y_{k,1}, y_{k,2}, \dots, y_{k,\hat{p}}, 1]^T, \text{ for } k = 1, 2, \dots, n & (23f) \end{cases}$$

Subsequently by multiplying both sides of the first phrase of (23a) ($w_i = -\frac{1}{2}[q_{i,1}, q_{i,2}, \dots, q_{i,\hat{p}}]^T$) by $\frac{\partial w_i^T}{\partial t_i}$, we can get the updating equation of the t_i parameter. For simplicity, we replace $\frac{\partial w_i^T}{\partial t_i}$ by ξ_{t_i} , that is the normalized vector of the $\frac{\partial w_i^T}{\partial t_i}$ vector. Therefore, the updated relation for the t_i parameter can be obtained as follows:

$$\xi_{t_i} = \text{Norm}\left(\frac{\partial w_i^T}{\partial t_i}\right), \text{ for all } t_i \text{ s and } i = 1, 2, \dots, c \quad (24)$$

$$\xi_{t_i} w_i = -\frac{1}{2}\xi_{t_i}[q_{i,1}, q_{i,2}, \dots, q_{i,\hat{p}}]^T \Rightarrow t_i = ?, \text{ for all } t_i \text{ s and } i = 1, 2, \dots, c. \quad (25)$$

To complete the VFCSS description, we will apply the proposed VFCSS clustering method over various crisp and non-crisp numbers.

In this part, we apply the proposed VFCSS clustering of Section 4 over crisp and various non-crisp numbers with the metrics that are introduced in Section 3.

2.4. VFCSS Applied to Crisp Numbers

The VFCSS can be applied over crisp numbers. In this state according to the definition of Euclidean distance, the definition of the Y and W sets are as follows:

$$\begin{aligned} Y &= \{y_1, y_2, \dots, y_n\}, y_k = [y_{k,1}, y_{k,2}, \dots, y_{k,p}]^T, y_{k,j} = x_{k,j} \\ W &= \{w_1, w_2, \dots, w_c\}, w_i = [w_{i,1}, w_{i,2}, \dots, w_{i,p}]^T, w_{i,j} = v_{i,j} \end{aligned} \quad (26)$$

where $k = 1, 2, \dots, n, i = 1, 2, \dots, c$ and $j = 1, 2, \dots, p$. In this case $Y = X$ and $W = V$. Therefore, we can conclude $\hat{p} = p$. In this case $\xi_{t_i} = \text{Norm}\left(\frac{\partial w_i^T}{\partial t_i}\right) = \frac{\partial w_i^T}{\partial t_i}$ that is obtained from Table 1.

Table 1. ζ for VFCSS applying over crisp numbers.

dt_i	$v_{i,j}, \text{ for } j = 1, 2, \dots, p$
$\zeta_{t_i} = \text{Norm}\left(\frac{\partial w_i^T}{\partial t_i}\right) = \frac{\partial w_i^T}{\partial t_i}$	$\begin{bmatrix} \underbrace{0, 0, \dots, 0}_{(j-1)}, \underbrace{1, 0, 0, \dots, 0}_{(p-j)} \end{bmatrix}$

Then using (25), parameter t_i can be obtained as (5). We can conclude the VFCSS applied over crisp numbers is equivalent with this concept in the first Equation of (5a).

2.5. The VFCSS Applied to Fuzzy Numbers

In this part, we apply VFCSS clustering to various introduced fuzzy type numbers, see Section 3.

2.5.1. The VFCSS Applied to LR-Type Fuzzy Numbers

For LR-type fuzzy numbers, when the metric (13) is used, we define Y and W sets as below:

$$\begin{aligned} Y &= \{y_1, y_2, \dots, y_n\}, y_k = [y_{k,1}, y_{k,2}, \dots, y_{k,p}]^T, y_{k,j} \\ &= [m_{1\tilde{x}_{k,j}}, m_{2\tilde{x}_{k,j}}, m_{1\tilde{x}_{k,j}} - l\alpha_{\tilde{x}_{k,j}}, m_{2\tilde{x}_{k,j}} + r\beta_{\tilde{x}_{k,j}}]^T \\ W &= \{w_1, w_2, \dots, w_c\}, w_i = [w_{i,1}, w_{i,2}, \dots, w_{i,p}]^T, w_{i,j} \\ &= [m_{1\tilde{v}_{i,j}}, m_{2\tilde{v}_{i,j}}, m_{1\tilde{v}_{i,j}} - l\alpha_{\tilde{v}_{i,j}}, m_{2\tilde{v}_{i,j}} + r\beta_{\tilde{v}_{i,j}}]^T \end{aligned} \quad (27)$$

where $k = 1, 2, \dots, n$, $i = 1, 2, \dots, c$ and $j = 1, 2, \dots, p$. It is observed in this case $\hat{p} = 4p$. In this case, for all t_i s ($t_i \in \left\{m_{1\tilde{v}_{i,j}}, m_{2\tilde{v}_{i,j}}, \alpha_{\tilde{v}_{i,j}}, \beta_{\tilde{v}_{i,j}}\right\}_{j=1}^p$), $\frac{\partial w_i^T}{\partial t_i}$ and ζ_{t_i} are obtained in Tables 2 and 3.

Table 2. $\frac{\partial w_i^T}{\partial t_i}$ for VFCSS applying over LR-type fuzzy numbers when metric (13) is used.

t_i	$m_{1\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$m_{2\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$\alpha_{\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$\beta_{\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$
$\frac{\partial w_i^T}{\partial t_i}$	$\begin{bmatrix} \underbrace{0, 0, \dots, 0}_{4(j-1)}, \underbrace{1, 0, 1, 0, 0, 0, \dots, 0}_{4(p-j)} \end{bmatrix}$	$\begin{bmatrix} \underbrace{0, 0, \dots, 0}_{4(j-1)}, \underbrace{0, 1, 0, 1, 0, 0, \dots, 0}_{4(p-j)} \end{bmatrix}$	$\begin{bmatrix} \underbrace{0, 0, \dots, 0}_{4(j-1)+2}, \underbrace{-l, 0, 0, \dots, 0}_{4(p-j)+1} \end{bmatrix}$	$\begin{bmatrix} \underbrace{0, 0, \dots, 0}_{4(j-1)+3}, \underbrace{r, 0, 0, \dots, 0}_{4(p-j)} \end{bmatrix}$

Table 3. ζ for VFCSS applying over LR-type fuzzy numbers when metric (13) is used.

t_i	$m_{1\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$m_{2\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$\alpha_{\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$\beta_{\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$
ζ_{t_i}	$\begin{bmatrix} \underbrace{0, 0, \dots, 0}_{4(j-1)}, \underbrace{1, 0, 1, 0, 0, 0, \dots, 0}_{4(p-j)} \end{bmatrix}$	$\begin{bmatrix} \underbrace{0, 0, \dots, 0}_{4(j-1)}, \underbrace{0, 1, 0, 1, 0, 0, \dots, 0}_{4(p-j)} \end{bmatrix}$	$\begin{bmatrix} \underbrace{0, 0, \dots, 0}_{4(j-1)+2}, \underbrace{1, 0, 0, \dots, 0}_{4(p-j)+1} \end{bmatrix}$	$\begin{bmatrix} \underbrace{0, 0, \dots, 0}_{4(j-1)+3}, \underbrace{1, 0, 0, \dots, 0}_{4(p-j)} \end{bmatrix}$

Finally, we can arrive at updated relations of parameters $m_{1\tilde{v}_{i,j}}, m_{2\tilde{v}_{i,j}}, \alpha_{\tilde{v}_{i,j}}$ and $\beta_{\tilde{v}_{i,j}}$ from (25) as follows:

$$\left. \begin{aligned} m_{1\tilde{v}_{i,j}} &= -\frac{1}{4}(q_{i,4j-3} + q_{i,4j-1}) + \frac{l}{2}\alpha_{\tilde{v}_{i,j}} \\ m_{2\tilde{v}_{i,j}} &= -\frac{1}{4}(q_{i,4j-2} + q_{i,4j}) - \frac{r}{2}\beta_{\tilde{v}_{i,j}} \\ \alpha_{\tilde{v}_{i,j}} &= \frac{1}{2l}q_{i,4j-1} + \frac{1}{l}m_{1\tilde{v}_{i,j}} \\ \beta_{\tilde{v}_{i,j}} &= -\frac{1}{2r}q_{i,4j} - \frac{1}{r}m_{2\tilde{v}_{i,j}} \end{aligned} \right\} \text{ for } \begin{cases} i = 1, 2, \dots, c \\ j = 1, 2, \dots, p \end{cases} \quad (28)$$

In the other state of LR-type fuzzy numbers, suppose $m_{1\tilde{x}_{k,j}} = m_{2\tilde{x}_{k,j}} = m_{\tilde{x}_{k,j}}$ and $m_{1\tilde{v}_{i,j}} = m_{2\tilde{v}_{i,j}} = m_{\tilde{v}_{i,j}}$. Regarding the Yang distance definition of (16), Y and W are defined as follows:

$$\begin{aligned} Y &= \{y_1, y_2, \dots, y_n\}, y_k = [y_{k,1}, y_{k,2}, \dots, y_{k,p}]^T, \\ y_{k,j} &= [m_{\tilde{x}_{k,j}}, m_{\tilde{x}_{k,j}} - l\alpha_{\tilde{x}_{k,j}}, m_{\tilde{x}_{k,j}} + r\beta_{\tilde{x}_{k,j}}]^T \\ W &= \{w_1, w_2, \dots, w_c\}, w_i = [w_{i,1}, w_{i,2}, \dots, w_{i,p}]^T, \\ w_{i,j} &= [m_{\tilde{v}_{i,j}}, m_{\tilde{v}_{i,j}} - l\alpha_{\tilde{v}_{i,j}}, m_{\tilde{v}_{i,j}} + r\beta_{\tilde{v}_{i,j}}]^T \end{aligned} \quad (29)$$

In this case $\hat{p} = 3p$. For this case ξ_{t_i} and updated relations of $m_{\tilde{v}_{i,j}}$, $\alpha_{\tilde{v}_{i,j}}$ and $\beta_{\tilde{v}_{i,j}}$ are as in Table 4 and (30).

Table 4. ξ for VFCSS applying over LR-type fuzzy numbers when metric (16) is used.

t_i	$m_{\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$\alpha_{\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$\beta_{\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$
ξ_{t_i}	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{3(j-1)}, \overbrace{1, 1, 1, 0, 0, \dots, 0}^{3(p-j)} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{3(j-1)+1}, \overbrace{1, 0, 0, \dots, 0}^{3(p-j)+1} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{3(j-1)+2}, \overbrace{1, 0, 0, \dots, 0}^{3(p-j)} \end{bmatrix}$

$$\left. \begin{aligned} m_{\tilde{v}_{i,j}} &= -\frac{1}{6}(q_{i,3j-2} + q_{i,3j-1} + q_{i,3j}) + \frac{l}{3}\alpha_{\tilde{v}_{i,j}} - \frac{r}{3}\beta_{\tilde{v}_{i,j}} \\ \alpha_{\tilde{v}_{i,j}} &= \frac{1}{2l}q_{i,3j-1} + \frac{1}{l}m_{\tilde{v}_{i,j}} \\ \beta_{\tilde{v}_{i,j}} &= -\frac{1}{2r}q_{i,3j} - \frac{1}{r}m_{\tilde{v}_{i,j}} \end{aligned} \right\} \text{for } \begin{cases} i = 1, 2, \dots, c \\ j = 1, 2, \dots, p \end{cases} \quad (30)$$

As we cannot define Y and W to match (17) to (19), therefore the proposed VFCSS cannot be applied on (19).

2.5.2. The VFCSS Applied to TFN-Type Fuzzy Numbers

For TFN numbers, when the metric (14) is used, we define Y and W sets as below:

$$\begin{aligned} Y &= \{y_1, y_2, \dots, y_n\}, y_k = [y_{k,1}, y_{k,2}, \dots, y_{k,p}]^T, y_{k,j} \\ &= \left[\frac{2x_{1\tilde{x}_{k,j}} - x_{2\tilde{x}_{k,j}}}{2}, \frac{2x_{1\tilde{x}_{k,j}} + x_{2\tilde{x}_{k,j}}}{2}, \frac{2x_{1\tilde{x}_{k,j}} - x_{2\tilde{x}_{k,j}} - x_{3\tilde{x}_{k,j}}}{2}, \frac{2x_{1\tilde{x}_{k,j}} + x_{2\tilde{x}_{k,j}} + x_{4\tilde{x}_{k,j}}}{2} \right]^T \\ W &= \{w_1, w_2, \dots, w_c\}, w_i = [w_{i,1}, w_{i,2}, \dots, w_{i,p}]^T, w_{i,j} \\ &= \left[\frac{2v_{1\tilde{v}_{i,j}} - v_{2\tilde{v}_{i,j}}}{2}, \frac{2v_{1\tilde{v}_{i,j}} + v_{2\tilde{v}_{i,j}}}{2}, \frac{2v_{1\tilde{v}_{i,j}} - v_{2\tilde{v}_{i,j}} - v_{3\tilde{v}_{i,j}}}{2}, \frac{2v_{1\tilde{v}_{i,j}} + v_{2\tilde{v}_{i,j}} + v_{4\tilde{v}_{i,j}}}{2} \right]^T \end{aligned} \quad (31)$$

It is observed $\hat{p} = 4p$. In this case ξ_{t_i} and updated relations of $v_{1\tilde{v}_{i,j}}$, $v_{2\tilde{v}_{i,j}}$, $v_{3\tilde{v}_{i,j}}$ and $v_{4\tilde{v}_{i,j}}$ are as in Table 5 and (32).

$$\left. \begin{aligned} v_{1\tilde{v}_{i,j}} &= -\frac{1}{8}(q_{i,4j-3} + q_{i,4j-2} + q_{i,4j-1} + q_{i,4j} - v_{3\tilde{v}_{i,j}} + v_{4\tilde{v}_{i,j}}) \\ v_{2\tilde{v}_{i,j}} &= -\frac{1}{4}(-q_{i,4j-3} + q_{i,4j-2} - q_{i,4j-1} + q_{i,4j} + v_{3\tilde{v}_{i,j}} + v_{4\tilde{v}_{i,j}}) \\ v_{3\tilde{v}_{i,j}} &= q_{i,4j-1} + 2v_{1\tilde{v}_{i,j}} - v_{2\tilde{v}_{i,j}} \\ v_{4\tilde{v}_{i,j}} &= -q_{i,4j} - 2v_{1\tilde{v}_{i,j}} - v_{2\tilde{v}_{i,j}} \end{aligned} \right\} \text{for } \begin{cases} i = 1, 2, \dots, c \\ j = 1, 2, \dots, p \end{cases} \quad (32)$$

Table 5. ξ for VFCSS applying over TFN-type fuzzy numbers when metric (14) is used.

t_i	$v_{1\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$v_{2\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$v_{3\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$	$v_{4\tilde{v}_{i,j}}, \text{ for } j = 1, 2, \dots, p$
ξ_{t_i}	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{4(j-1)}, \overbrace{1, 1, 1, 1, 0, 0, \dots, 0}^{4(p-j)} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{4(j-1)}, \overbrace{-1, 1, -1, 1, 0, 0, \dots, 0}^{4(p-j)} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{4(j-1)+2}, \overbrace{1, 0, 0, \dots, 0}^{4(p-j)+1} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{4(j-1)+3}, \overbrace{1, 0, 0, \dots, 0}^{4(p-j)} \end{bmatrix}$

For TFN numbers when metric (15) is used, we define the Y and W sets as follows:

$$Y = \{y_1, y_2, \dots, y_n\}, y_k = [y_{k,1}, y_{k,2}, \dots, y_{k,p}]^T, y_{k,j} = [x_{1\tilde{x}_{k,j}}, x_{2\tilde{x}_{k,j}}, x_{3\tilde{x}_{k,j}}, x_{4\tilde{x}_{k,j}}]^T \quad (33)$$

$$W = \{w_1, w_2, \dots, w_c\}, w_i = [w_{i,1}, w_{i,2}, \dots, w_{i,p}]^T, w_{i,j} = [v_{1\tilde{v}_{i,j}}, v_{2\tilde{v}_{i,j}}, v_{3\tilde{v}_{i,j}}, v_{4\tilde{v}_{i,j}}]^T$$

In this case $\hat{p} = 4p$, furthermore ξ_{t_i} and updated relations of $v_{1\tilde{v}_{i,j}}, v_{2\tilde{v}_{i,j}}, v_{3\tilde{v}_{i,j}}$ and $v_{4\tilde{v}_{i,j}}$ are as in Table 6 and (34).

$$\left. \begin{aligned} v_{1\tilde{v}_{i,j}} &= -\frac{1}{2}q_{i,4j-3} \\ v_{2\tilde{v}_{i,j}} &= -\frac{1}{2}q_{i,4j-2} \\ v_{3\tilde{v}_{i,j}} &= -\frac{1}{2}q_{i,4j-1} \\ v_{4\tilde{v}_{i,j}} &= -\frac{1}{2}q_{i,4j} \end{aligned} \right\} \text{for } \begin{cases} i = 1, 2, \dots, c \\ j = 1, 2, \dots, p \end{cases} \quad (34)$$

Table 6. ξ for VFCSS over applying over TFN-type fuzzy numbers when metric (14) is used.

t_i	$v_{1\tilde{v}_{i,j}}, \text{for } j = 1, 2, \dots, p$	$v_{2\tilde{v}_{i,j}}, \text{for } j = 1, 2, \dots, p$	$v_{3\tilde{v}_{i,j}}, \text{for } j = 1, 2, \dots, p$	$v_{4\tilde{v}_{i,j}}, \text{for } j = 1, 2, \dots, p$
ξ_{t_i}	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{4(j-1)} & \overbrace{1, 0, 0, \dots, 0}^{4(p-j)+3} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{4(j-1)+1} & \overbrace{1, 0, 0, \dots, 0}^{4(p-j)+2} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{4(j-1)+2} & \overbrace{1, 0, 0, \dots, 0}^{4(p-j)+1} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{4(j-1)+3} & \overbrace{1, 0, 0, \dots, 0}^{4(p-j)} \end{bmatrix}$

2.5.3. The VFCSS Applied to TAN-Type Fuzzy Numbers

If we use (18.1) as the metric when we apply VFCSS over TAN numbers, the definition of Y and W is as follows:

$$Y = \{y_1, y_2, \dots, y_n\}, y_k = [y_{k,1}, y_{k,2}, \dots, y_{k,p}]^T, y_{k,j} = \left[\sqrt{\frac{\alpha_1}{6}} x_{k,j}^L, \sqrt{\frac{\alpha_1}{6} + \alpha_2 + \frac{\alpha_3}{6}} x_{k,j}^C, \sqrt{\frac{\alpha_3}{6}} x_{k,j}^R, \sqrt{\frac{\alpha_1}{6}} (x_{k,j}^L + x_{k,j}^C), \sqrt{\frac{\alpha_3}{6}} (x_{k,j}^R + x_{k,j}^C) \right]^T \quad (35)$$

$$W = \{w_1, w_2, \dots, w_c\}, w_i = [w_{i,1}, w_{i,2}, \dots, w_{i,p}]^T, w_{i,j} = \left[\sqrt{\frac{\alpha_1}{6}} v_{i,j}^L, \sqrt{\frac{\alpha_1}{6} + \alpha_2 + \frac{\alpha_3}{6}} v_{i,j}^C, \sqrt{\frac{\alpha_3}{6}} v_{i,j}^R, \sqrt{\frac{\alpha_1}{6}} (v_{i,j}^L + v_{i,j}^C), \sqrt{\frac{\alpha_3}{6}} (v_{i,j}^R + v_{i,j}^C) \right]^T$$

In this case $\hat{p} = 5p$, furthermore ξ_{t_i} and updated relations of $v_{i,j}^L, v_{i,j}^C$ and $v_{i,j}^R$ are as in Table 7 and (36).

$$\left. \begin{aligned} v_{i,j}^L &= -\sqrt{\frac{3}{8\alpha_1}} (q_{i,5j-4} + q_{i,5j-1}) - \frac{1}{2}v_{i,j}^C \\ v_{i,j}^C &= -\frac{\sqrt{6}}{4(\alpha_1 + 3\alpha_2 + \alpha_3)} \left(\sqrt{(\alpha_1 + 6\alpha_2 + \alpha_3)} q_{i,5j-3} + \sqrt{\alpha_1} q_{i,5j-1} + \sqrt{\alpha_3} q_{i,5j} \right) - \frac{\alpha_1 v_{i,j}^L + \alpha_3 v_{i,j}^R}{2(\alpha_1 + 3\alpha_2 + \alpha_3)} \\ v_{i,j}^R &= -\sqrt{\frac{3}{8\alpha_1}} (q_{i,5j-2} + q_{i,5j}) - \frac{1}{2}v_{i,j}^C \end{aligned} \right\} \quad (36)$$

Table 7. ξ for VFCSS applying over TAN-type fuzzy numbers when metric (18.1) is used.

t_i	$v_{i,j}^L, \text{for } j = 1, 2, \dots, p$	$v_{i,j}^C, \text{for } j = 1, 2, \dots, p$	$v_{i,j}^R, \text{for } j = 1, 2, \dots, p$
ξ_{t_i}	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{5(j-1)} & \overbrace{1, 0, 0, \dots, 0}^{5(p-j)} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{5(j-1)} & 0, 1, 0, \sqrt{\frac{\alpha_1}{\alpha_1 + 6\alpha_2 + \alpha_3}}, \sqrt{\frac{\alpha_3}{\alpha_1 + 6\alpha_2 + \alpha_3}}, \overbrace{0, 0, \dots, 0}^{5(p-j)} \end{bmatrix}$	$\begin{bmatrix} \overbrace{0, 0, \dots, 0}^{5(j-1)} & 0, 0, 1, 0, 1, 0, \overbrace{0, 0, \dots, 0}^{5(p-j)} \end{bmatrix}$

However, the proposed VFCSS cannot be implemented for the exponential based metric of (18.1).

2.6. The VFCSS over Symbolic Numbers

Using the VFCSS method, the definition of the Y and W crisp vectors for dissimilarity definitions of (12) is as follows:

$$Y = \{y_1, y_2, \dots, y_n\}, y_k = [y_{k,1}, y_{k,2}, \dots, y_{k,p}]^T, y_{k,j} = [\alpha_{\tilde{x}_{k,j}}, \beta_{\tilde{x}_{k,j}}]^T \quad (37)$$

$$W = \{w_1, w_2, \dots, w_c\}, w_i = [w_{i,1}, w_{i,2}, \dots, w_{i,p}]^T, w_{i,j} = [\alpha_{\tilde{v}_{i,j}}, \beta_{\tilde{v}_{i,j}}]^T$$

In this case $\hat{p} = 2p$. For this case ξ_{t_i} and updated relations of $\alpha_{\tilde{v}_{i,j}}$ and $\beta_{\tilde{v}_{i,j}}$ are as in Table 8 and Equation (38).

$$\left. \begin{aligned} \alpha_{\tilde{v}_{i,j}} &= -\frac{1}{2}q_{i,2j-1} \\ \beta_{\tilde{v}_{i,j}} &= -\frac{1}{2}q_{i,2j} \end{aligned} \right\} \text{for } \begin{cases} i = 1, 2, \dots, c \\ j = 1, 2, \dots, p \end{cases} \quad (38)$$

Table 8. ξ for VFCSS applying over symbolic numbers when metric (12) is used.

t_i	$\alpha_{\tilde{v}_{i,j}}, \text{for } j = 1, 2, \dots, p$	$\beta_{\tilde{v}_{i,j}}, \text{for } j = 1, 2, \dots, p$
ξ_{t_i}	$\left[\underbrace{0, 0, \dots, 0}_{2(j-1)}, \underbrace{1, 0, 0, \dots, 0}_{2(p-j)+1} \right]$	$\left[\underbrace{0, 0, \dots, 0}_{2(j-1)+1}, \underbrace{1, 0, 0, \dots, 0}_{2(p-j)} \right]$

Using the VFCSS method, the definition of the Y and W vectors for metric (9) is as follows:

$$\begin{aligned} Y &= \{y_1, y_2, \dots, y_n\}, y_k = [y_{k,1}, y_{k,2}, \dots, y_{k,p}]^T, y_{k,j} = [(1-z)\alpha_{\tilde{x}_{k,j}}, z\beta_{\tilde{x}_{k,j}}]^T \\ W &= \{w_1, w_2, \dots, w_c\}, w_i = [w_{i,1}, w_{i,2}, \dots, w_{i,p}]^T, w_{i,j} = [(1-z)\alpha_{\tilde{v}_{i,j}}, z\beta_{\tilde{v}_{i,j}}]^T \end{aligned} \quad (39)$$

Using these definitions, dissimilarity between two symbolic numbers \tilde{x}_k, \tilde{v}_i is obtained from the Euclidean distance between the two crisp vectors y_k, w_i . As the constraint $0 \leq z \leq 0.5$ is added to constraint (3), therefore the Lagrange function (20) will change in this case. Similar to the mentioned case, we can get the updated relations by performing the same procedure.

For dissimilarity definitions of (10), using the VFCSS method, the definition of Y and W vectors are as follows:

$$\begin{aligned} Y &= \{y_1, y_2, \dots, y_n\}, y_k = [y_{k,1}, y_{k,2}, \dots, y_{k,p}]^T, y_{k,j} = [\sqrt{\gamma_j}\alpha_{\tilde{x}_{k,j}}, \sqrt{\gamma_j}\beta_{\tilde{x}_{k,j}}]^T \\ W &= \{w_1, w_2, \dots, w_c\}, w_i = [w_{i,1}, w_{i,2}, \dots, w_{i,p}]^T, w_{i,j} = [\sqrt{\gamma_j}\alpha_{\tilde{v}_{i,j}}, \sqrt{\gamma_j}\beta_{\tilde{v}_{i,j}}]^T \end{aligned} \quad (40)$$

Using these definitions, dissimilarity between two symbolic numbers \tilde{x}_k, \tilde{v}_i is obtained from the Euclidean distance between two crisp vectors y_k, w_i . As the constraint $\prod_{j=1}^p \gamma_j = 1$ is added to constraint (3), therefore the Lagrange function (20) will change in this case. Similar to the mentioned case, we can get the updated relations by performing the same procedure.

For dissimilarity definitions of (11), we cannot define two crisp vectors y_k, w_i such that $d^2(y_k, w_i) = d^2(\tilde{x}_k, \tilde{v}_i)$.

In the last two items, regarding the change in the Lagrange function caused by adding other constraints to (3), we cannot use (25) for accessing updating relations. We must rewrite the Lagrange function for each case separately. After this, we must solve the resulting system of equations from corresponding derivations. Finally, we can obtain updating relations of parameters similar to the introduced case in this paper.

3. Results and Discussion

Each presented non-crisp type number above with any of the corresponding metrics can be simulated. In this section, we apply the VFCSS clustering method over LR-type fuzzy numbers, while the Yang metric of (13) is used. Simulations are performed using MATLAB software.

In LR-type fuzzy numbers, decreasing functions of $L(x)$ and $R(x)$ are presumed linear ($L(x) = R(x) = 1 - x$). In order to create fuzzy numbers, we use 2-dimension crisp numbers $X = \{x_k\}_{k=1}^n$ that are produced randomly on the boundary of circles with a few alternates and these are named crisp equivalents (CE) of fuzzy numbers, while $0 \leq x_{k,j} \leq 1$

for $j = 1, 2$ (j represents each dimension). An LR-type fuzzy number \tilde{x}_k is generated as follows:

$$\tilde{x}_{k,j} = (x_{k,j} - 0.01, x_{k,j} + 0.01, 0.02x_{k,j}, 0.02x_{k,j})_{LR} \quad (41)$$

The noise added to crisp numbers has a max absolute of 0.1 times the corresponding circle radius. This is for performance evaluation of the proposed VFCSS performance versus noise. In demonstration of LR-type fuzzy numbers in simulation result figures, the distance between m_1, m_2 is in bold while α, β are demonstrated normally. To evaluate the VFCSS clustering accuracy we use a Confusion Matrix (CM). A CM is a $c \times c$ matrix, where $CM(i, j)$ is the number of numbers from the i th class that are clustered as the j th cluster. Furthermore, in the resulting membership grade matrix (U), it is assumed any input number belongs to a cluster that has the maximum membership of it. In this paper, VFCSS is applied to analyzed numbers in various states and results are reported as follows.

State 1. In the first state, in the first step the proposed VFCSS is applied to four classes of numbers that are overlapping. The simulation result for this step is provided in Figure 2. In the second step, VFCSS is applied to four classes of numbers that are complex and overlapping. The simulation result for this step is provided in Figure 3. The CM of these two steps is expressed in Table 9.

Table 9. CM of VFCSS applied over LR-type fuzzy numbers of Figures 2 and 3 (first and second steps of state 1).

Output Clusters	Step	
	First Step	Second Step
Confusion Matrix	$\begin{bmatrix} 20, 0, 0, 0 \\ 1, 18, 0, 1 \\ 0, 0, 20, 0 \\ 0, 1, 0, 19 \end{bmatrix}$	$\begin{bmatrix} 20, 0, 0, 0 \\ 0, 20, 0, 0 \\ 2, 1, 16, 1 \\ 0, 1, 1, 18 \end{bmatrix}$

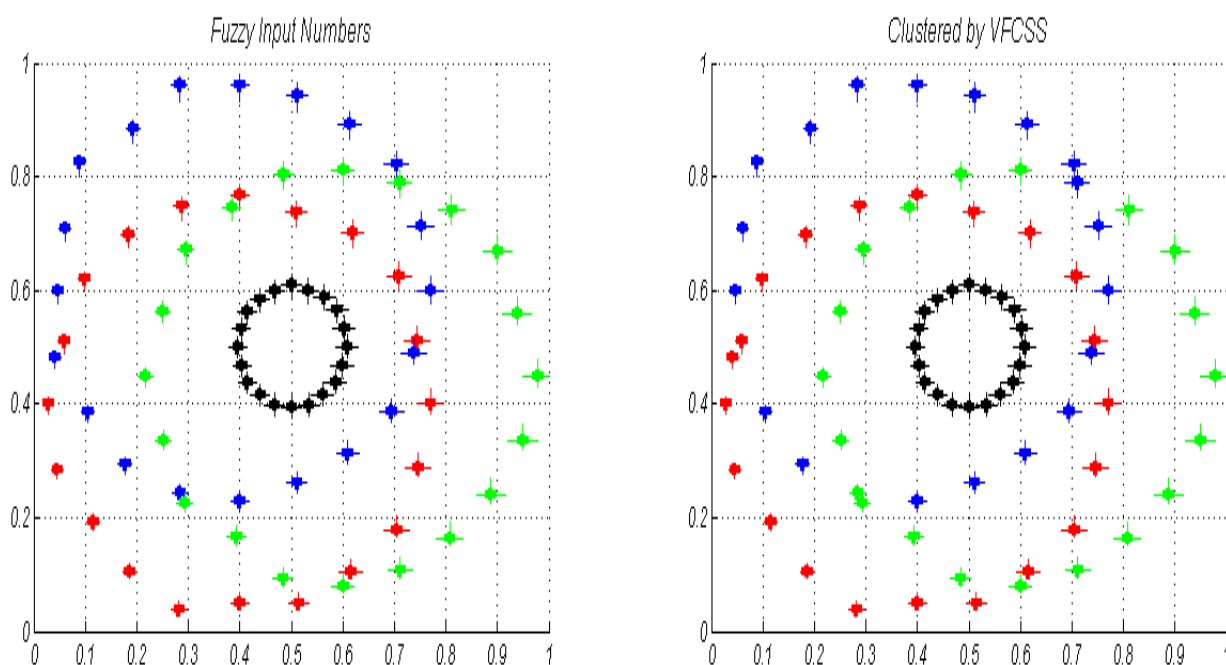


Figure 2. Proposed VFCSS clustering over input LR-type fuzzy numbers include four interference classes red, blue, black and green with (20,20,20,20) members, (first step of state 1).

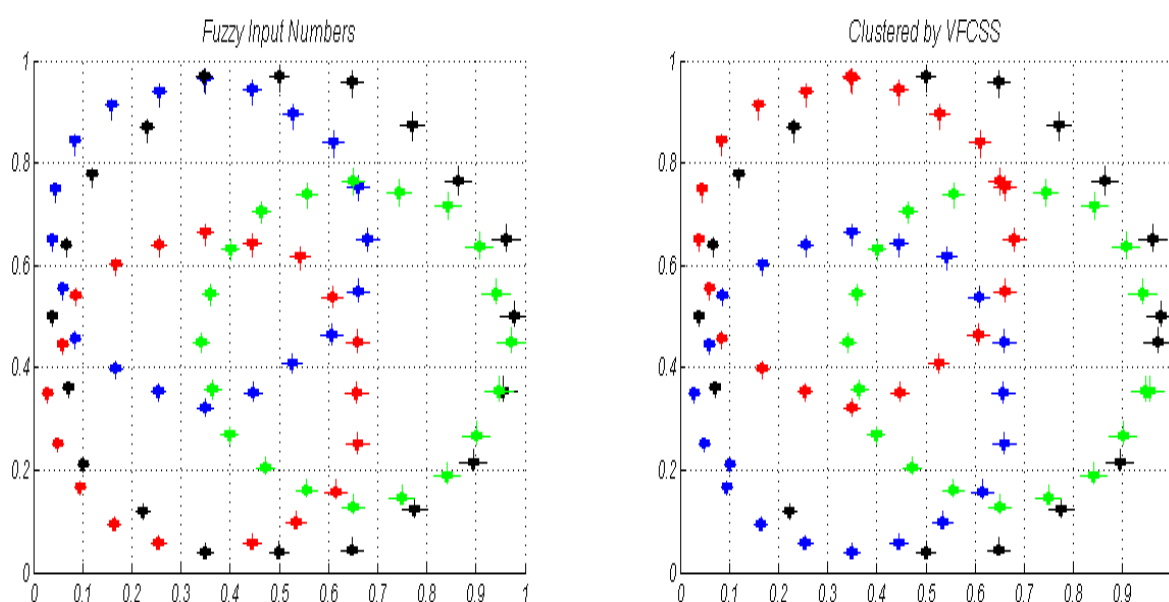


Figure 3. Proposed VFCSS clustering over input LR-type fuzzy numbers include four complexity interference classes red, blue, black and green with (20,20,20,20) members, (second step of state 1).

State 2. In this state, the input fuzzy numbers are not symmetric. On the other hand, we do not use from (41) to create fuzzy numbers. The procedure for producing fuzzy numbers in this state is as follows:

$$\tilde{x}_{k,j} = \left(x_{k,j} - 0.01, x_{k,j} + 0.03, 0.04x_{k,j}, 0.02x_{k,j} \right)_{LR} \quad (42)$$

In this state, we apply VFCSS clustering over LR-type fuzzy numbers with the same conditions as for state 1. Simulation results of the corresponding two steps of this state are demonstrated in Figures 4 and 5. The CM of these two steps is expressed in Table 10.

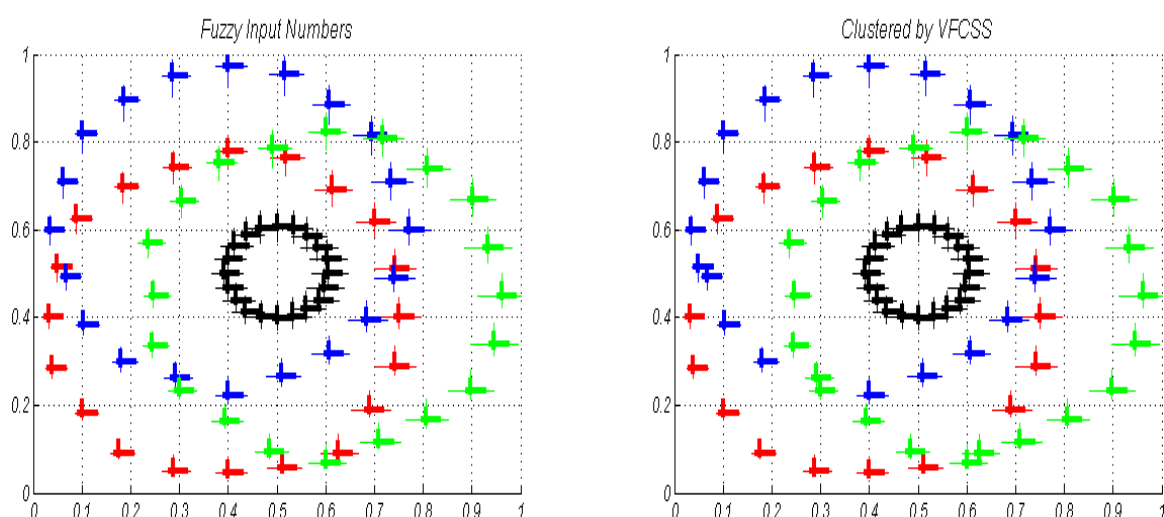


Figure 4. Proposed VFCSS clustering over input non-symmetric LR-type fuzzy numbers include four interference classes red, blue, black and green with (20,20,20,20) members, (first step of state 2).

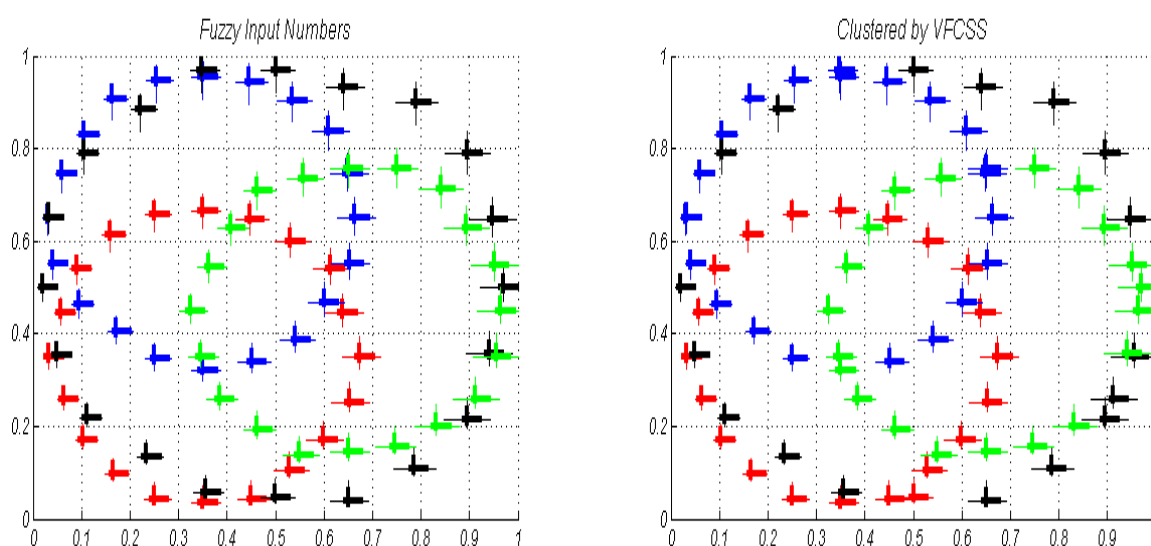


Figure 5. Proposed VFCSS clustering over input non-symmetric LR-type fuzzy numbers include four complexity interference classes red, blue, black and green with (20,20,20,20) members (second step of state 2).

Table 10. CM of VFCSS applied over LR-type fuzzy numbers of Figures 4 and 5 (first and second steps of state 2).

Output Clusters	Step	
	First Step	Second Step
Confusion Matrix	$\begin{bmatrix} 18, 1, 0, 1 \\ 0, 19, 0, 1 \\ 0, 0, 20, 0 \\ 0, 0, 0, 20 \end{bmatrix}$	$\begin{bmatrix} 20, 0, 0, 0 \\ 0, 19, 0, 1 \\ 1, 2, 15, 2 \\ 0, 1, 2, 17 \end{bmatrix}$

State 3. In the third state, as we can represent an LR-type fuzzy number with its mean $((m_1 + m_2)/2$ from 41) as a crisp number, we apply a similar condition as for the two previous states for crisp numbers. Therefore, conventional FCSS is applied over input crisp numbers. Simulation results for the corresponding two steps of this state are demonstrated in Figures 6 and 7. The CM of these two steps is expressed in Table 11.

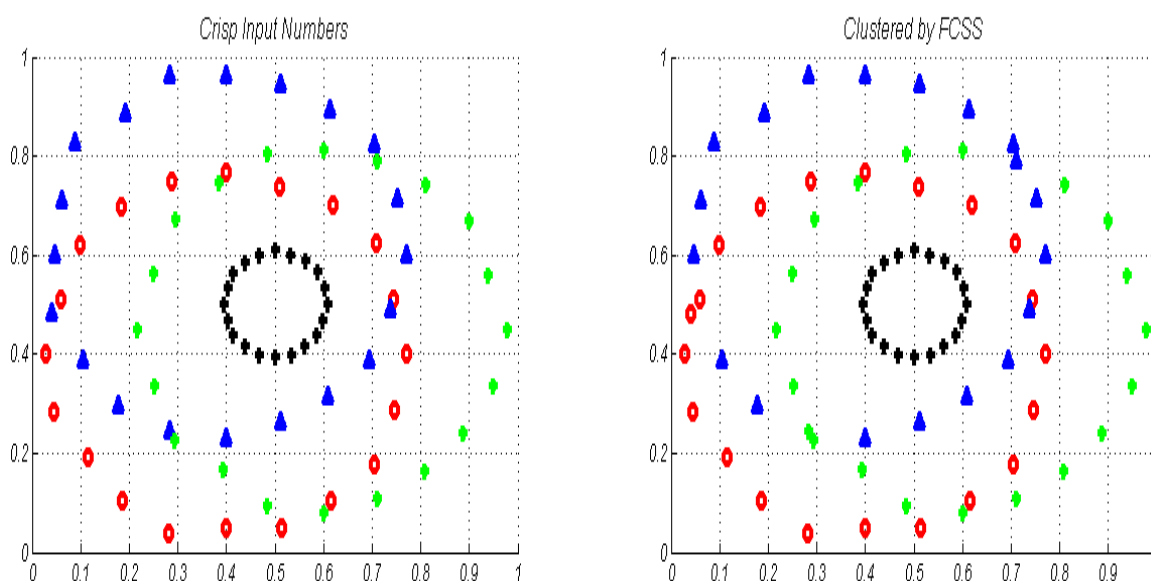


Figure 6. Conventional FCSS clustering over crisp input numbers include four interference classes red, blue, black, and green with (20,20,20,20) members, (first step of state 3).

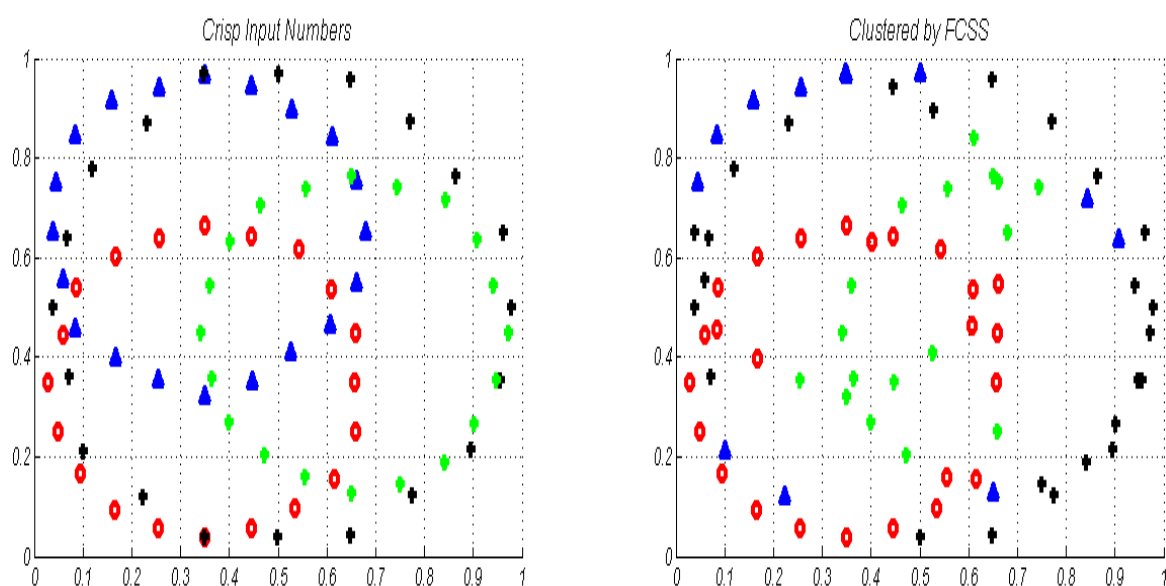


Figure 7. Conventional FCSS clustering over crisp input numbers include four complexity interference classes red, blue, black and green with (20,20,20,20) members (second step of state 3).

Table 11. CM of VFCSS applied over crisp numbers of Figures 6 and 7 (first and second steps of state 3).

Step		First Step	Second Step
Output Clusters			
Confusion Matrix		20,0,0,0	8,7,4,1
		1,18,0,1	7,8,3,2
		0,0,20,0	1,0,19,0
		0,1,0,19	0,1,1,18

In this state it is observed, in the first step, the performance of the proposed VFCSS (first step of state 3) and FCSS (first step of state 5) clustering methods are similar. While in the second step, the proposed VFCSS can cluster input numbers well, while the conventional FCSS cannot do this as well. Although in the simulation results the procedure of producing fuzzy numbers is very simple and primary. Performance of the proposed VFCSS can be improved further by a change in the procedure of producing fuzzy numbers (41), as well.

State 4. In the last state, the application of the proposed model is illustrated in real life. Geomorphology science and remotely sensed images are selected for simulation in this state. Geomorphology is the study of landforms dealing with the terrain relief. It is the morphology of the earth surface that works as the basic element separating geomorphology into an autonomous science in the Earth sciences. By analyzing the geometry of landforms, it is possible to study the relief and its properties from a validated perspective using mathematical apparatus [34]. Furthermore, remotely sensed imagery has been used widely in geomorphology due to the availability of satellite data, with its value measurable by the point to which it can meet the investigative needs of geomorphologists. Geomorphologists are therefore concerned with the Earth surface geometry and composition of the terrain relief. They use this information to determine presently operating processes, as well as predictive prior landforms and the events. They try to model geomorphological processes and use a wide range of techniques to predict future land surface change [35]. Therefore, in this paper, three georeferenced satellite images (obtained using Google maps) illustrated in Figure 8 are utilized, where they are related to the US with coordinates (Upper Left: 37°24′10″N, 105°35′12″W and Lower Right: 37°22′30″N, 105°32′35″W). These images are

used to estimate the area of the agricultural fields (by estimating the circle radius values). For this purpose, using Arc GIS 10.5 software, the circles are extracted with the clip tool.



(a)



(b)



(c)

Figure 8. Three georeferenced satellite images (obtained using Google maps) obtained over three different dates (a) 23-10-2011 (b) 09-09-2016 (c) 30-12-2018 (used for state 4).

If we demonstrate the extracted circles from images (a), (b) and (c) of Figure 8 by red, green and blue color respectively, the resulting circles are as in Figure 9. It can be observed, some of the resulting circles are non-overlapping while in some areas, as in the worst case, all 3 circles are in contact and are specified by red, green, and blue colors. In some area, two colors from RGB overlap and combined and cyan, yellow, and pink colors result. Finally, in rare cases, by complete overlapping of the circles, white color results. For this case, the resulting radius by conventional FCSS is reported in Table 12. We can observe various values for each circle are obtained by conventional FCSS (in an un-regular manner, it means the resulting radius from the first image in some cases is the min value, in other cases is the max value, etc.). This ambiguity will be increased, by increasing the number of available images. Furthermore, computational complexity will be increased by growing the number of available images as well. For this reason, FCSS must be applied on image circles separately.

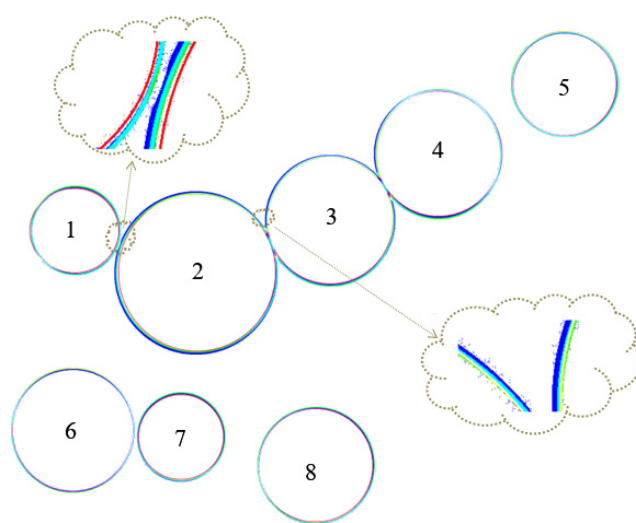


Figure 9. The extracted circles from images (a), (b) and (c) Figure 8 represented by red, green and blue colors, respectively.

Table 12. Simulation results for state 4 and comparison of resulting radii with FCSS and VFCSS algorithms.

Circle No Radius	1	2	3	4	5	6	7	8
Result by applying FCSS on image (a) Figure 8	205	377	301	310	250	285	201	268
Result by applying FCSS on image (b) Figure 8	212	380	307	314	258	297	211	279
Result by applying FCSS on image (c) Figure 8	210	385	315	318	257	291	209	274
Result by VFCSS	209	381	306	315	255	293	208	274

However, on the other side, the proposed VFCSS method is applied on the resulting fuzzy numbers and circles for three images of Figure 9 (considering the fuzzification process of [36] and associated interval-symbolic numbers for each circle's pixels). Simulation results (reported in Table 12) show only one acceptable and moderate value for each circle; even by increasing the available images that leads to more exact and nearer to the true radius value (see [36]). Furthermore, the computation complexity in the proposed method is very low, fixed, and equivalent with one time applying the FCSS, even by increasing the available number of images.

For more remote sensing applications, Figure 10 shows the proposed model results on a fish farm satellite image which is captured as Aquaculture farms off the coast of Greece

by Bernhard Lang [37]. The extracted circles and simulation results for circles radii with VFCSS algorithm are shown. It should be noted that the fuzzy-related works can be utilized without accessing a big dataset and just by a single image. These models can be applied directly without any training and validation processes. The proposed model is a new fuzzy clustering model which can be utilized for remote sensing applications. This fuzzy model can be helpful for designing smart image processing systems. For lower-quality images, it just is needed to add some simple preprocessing (noise removal, adjustment, etc.) to have a high-quality image for the proposed model.



Figure 10. A fish farm satellite image (Left) and the extracted circles from the image and radii results (Right) using VFCSS algorithm.

4. Conclusions

In this paper, a vector form of fuzzy c-spherical shells clustering (VFCSS) has been proposed. We can cluster non-crisp numbers that have spherical shells form using the proposed VFCSS. These non-crisp numbers can be available, or we can produce them from crisp number. It has been shown that we can improve the performance of conventional fuzzy c-spherical shells clustering over crisp numbers using the proposed VFCSS. This fact can be obtained by choosing a suitable procedure for producing non-crisp (fuzzy) numbers from crisp numbers. Furthermore, simulation results show better performance and low-cost computational complexity of the proposed VFCSS method versus the conventional FCSS in real life application in morphology science. We implement our clustering model over various types of geomorphology images for extracting the radii of circular agricultural fields using remotely sensed images and, we applied our model on these types of images to show that this clustering model can be utilized for remote sensing applications. To apply new deep learning models on an application we need a big and proper dataset to train, valid and test the models. The main uniqueness of fuzzy-related works is that we can utilize them without accessing a big dataset. We can use these models directly without any training and validation processes. Our model is a new fuzzy clustering model which can be utilized for remote sensing wapplications.

Author Contributions: Methodology, H.M.; software, I.A.K. and H.M.; validation, I.A.K., H.M., G.D. and D.T.; investigation, I.A.K.; writing—original draft preparation, H.M.; writing—review and editing, I.A.K., G.D. and D.T.; visualization, I.A.K.; supervision, I.A.K., G.D. and D.T.; project administration, I.A.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work has received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement, SMART 4.0, No. 847577; and a research grant from Science Foundation Ireland (SFI) under Grant Number 16/RC/3918 (Ireland’s European Structural and Investment Funds Programmes and the European Regional Development Fund 2014–2020).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

- Bellman, R.; Kalaba, R.; Zadeh, L.A. Abstraction and Pattern Classification. *J. Math. Anal. Appl.* **1966**, *13*, 1–7. [\[CrossRef\]](#)
- Bezdek, J.C. *Pattern Recognition with Fuzzy Objective Function Algorithms*; Kluwer Academic Publishers: New York, NY, USA, 1981.
- Yang, M.-S. A survey of fuzzy clustering. *Math. Comput. Model.* **1993**, *18*, 1–16. [\[CrossRef\]](#)
- MacQueen, J. Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, CA, USA, 21 June–18 July 1965; Volume 1, pp. 281–297.
- Zadeh, L.A. Fuzzy sets. *Inf. Control.* **1965**, *8*, 338–353. [\[CrossRef\]](#)
- Wu, K.-L.; Yang, M.-S. Alternative c-means clustering algorithms. *Pattern Recognit.* **2002**, *35*, 2267–2278. [\[CrossRef\]](#)
- Hadi, M.; Morteza, K.; Hadi, S.Y. Vector fuzzy C-means. *J. Intell. Fuzzy Syst.* **2013**, *24*, 363–381. [\[CrossRef\]](#)
- Hossein-Abad, H.M.; Shabanian, M.; Kazerouni, I.A. Fuzzy c-means clustering method with the fuzzy distance definition applied on symmetric triangular fuzzy numbers. *J. Intell. Fuzzy Syst.* **2020**, *38*, 2891–2905. [\[CrossRef\]](#)
- Hossein-Abad, H.M.; Shabanian, M.; Kazerouni, I.A. Vectorized Kernel-Based Fuzzy C-Means: A Method to Apply KFCM on Crisp and Non-Crisp Numbers. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* **2020**, *28*, 635–659. [\[CrossRef\]](#)
- Saha, A.; Das, S. On the unification of possibilistic fuzzy clustering: Axiomatic development and convergence analysis. *Fuzzy Sets Syst.* **2018**, *340*, 73–90. [\[CrossRef\]](#)
- Yu, H.; Fan, J.; Lan, R.; La, R. Suppressed possibilistic c-means clustering algorithm. *Appl. Soft Comput.* **2019**, *80*, 845–872. [\[CrossRef\]](#)
- Saha, A.; Das, S. Axiomatic generalization of the membership degree weighting function for fuzzy C means clustering: Theoretical development and convergence analysis. *Inf. Sci.* **2017**, *408*, 129–145. [\[CrossRef\]](#)
- Dave, R. Fuzzy shell-clustering and applications to circle detection in digital images. *Int. J. Gen. Syst.* **1990**, *16*, 343–355. [\[CrossRef\]](#)
- Dave, R.; Bhaswan, K. Adaptive fuzzy c-shells clustering and detection of ellipses. *IEEE Trans. Neural Netw.* **1992**, *3*, 643–662. [\[CrossRef\]](#) [\[PubMed\]](#)
- Krishnapuram, R.; Nasraoui, O.; Frigui, H. The fuzzy c spherical shells algorithm: A new approach. *IEEE Trans. Neural Netw.* **1992**, *3*, 663–671. [\[CrossRef\]](#)
- Man, Y.; Gath, I. Detection and separation of ring-shaped clusters using fuzzy clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **1994**, *16*, 855–861. [\[CrossRef\]](#)
- Gath, I.; Hoory, D. Fuzzy clustering of elliptic ring-shaped clusters. *Pattern Recognit. Lett.* **1995**, *16*, 727–741. [\[CrossRef\]](#)
- Krishnapuram, R.; Frigui, H.; Nasraoui, O. Fuzzy and possibilistic shell clustering algorithms and their application to boundary detection and surface approximation. *IEEE Trans. Fuzzy Syst.* **1995**, *3*, 29–43. [\[CrossRef\]](#)
- Bezdek, J.C.; Hathaway, R.J.; Pal, N.R. Norm-induced shell-prototypes (NISP) clustering. *Neural Parallel Sci. Comput.* **1995**, *3*, 431–449.
- Hoepfner, F. Fuzzy shell clustering algorithms in image processing: Fuzzy C-rectangular and 2-rectangular shells. *IEEE Trans. Fuzzy Syst.* **1997**, *5*, 599–613. [\[CrossRef\]](#)
- Wang, T. Possibilistic Shell Clustering of Template-Based Shapes. *IEEE Trans. Fuzzy Syst.* **2008**, *17*, 777–793. [\[CrossRef\]](#)
- Song, Q.; Yang, X.; Soh, Y.C.; Wang, Z.M. An information-theoretic fuzzy C-spherical shells clustering algorithm. *Fuzzy Sets Syst.* **2010**, *161*, 1755–1773. [\[CrossRef\]](#)
- Mahdipour, H.H.-A.; Khademi, M.; Sadoghi, H.Y. Model-based fuzzy c-shells clustering. *Neural Comput. Appl.* **2011**, *21*, 29–41. [\[CrossRef\]](#)
- D’Urso, P.; Giordani, P. A weighted fuzzy c-means clustering model for fuzzy data. *Comput. Stat. Data Anal.* **2006**, *50*, 1496–1523. [\[CrossRef\]](#)
- De Carvalho, F. Fuzzy clustering algorithms for symbolic interval data based on adaptive and non-adaptive Euclidean distances. In Proceedings of the 2006 Ninth Brazilian Symposium on Neural Networks (SBRN’06), Ribeiro Preto, Brazil, 23–27 October 2006; pp. 60–65.
- de Carvalho, F. Fuzzy c-means clustering methods for symbolic interval data. *Pattern Recognit. Lett.* **2007**, *28*, 423–437. [\[CrossRef\]](#)
- Chuang, C.-C.; Jeng, J.-T.; Li, C.-W. Fuzzy c-means clustering algorithm with unknown number of clusters for symbolic interval data. In Proceedings of the 2008 SICE Annual Conference, Tokyo, Japan, 20–22 August 2008; pp. 358–363.
- Hathaway, R.; Bezdek, J.; Pedrycz, W. A parametric model for fusing heterogeneous fuzzy data. *IEEE Trans. Fuzzy Syst.* **1996**, *4*, 270–281. [\[CrossRef\]](#)
- Yang, M.-S.; Ko, C.-H. On a class of fuzzy c-numbers clustering procedures for fuzzy data. *Fuzzy Sets Syst.* **1996**, *84*, 49–60. [\[CrossRef\]](#)
- Yang, M.-S.; Hwang, P.-Y.; Chen, D.-H. Fuzzy clustering algorithms for mixed feature variables. *Fuzzy Sets Syst.* **2004**, *141*, 301–317. [\[CrossRef\]](#)
- Hung, W.-L.; Yang, M.-S. Fuzzy clustering on LR-type fuzzy numbers with an application in Taiwanese tea evaluation. *Fuzzy Sets Syst.* **2005**, *150*, 561–577. [\[CrossRef\]](#)

32. Yang, M.-S.; Hung, W.-L.; Cheng, F.-C. Mixed-variable fuzzy clustering approach to part family and machine cell formation for GT applications. *Int. J. Prod. Econ.* **2006**, *103*, 185–198. [[CrossRef](#)]
33. Rong, L.; Jiu-lun, F. A Fuzzy C-means Type Clustering Algorithm on Triangular Fuzzy Numbers. In Proceedings of the Sixth International Conference on Fuzzy Systems and Knowledge Discovery, Tianjin, China, 14–16 August 2009.
34. Lopatin, D.V.; Zhironov, A.I. Geomorphology in the system of Earth sciences. *Geogr. Nat. Resour.* **2017**, *38*, 313–318. [[CrossRef](#)]
35. Smith, M.; Pain, C. Applications of remote sensing in geomorphology. *Prog. Phys. Geogr. Earth Environ.* **2009**, *33*, 568–582. [[CrossRef](#)]
36. Hadi, S.Y.; Morteza, K. Efficient Land-cover segmentation using Meta fusion. *J. Inf. Syst. Telecommun. (JIST)* **2016**, *4*, 152–166.
37. Lang, B. Aquaculture Farms off the Coast of Greece. Available online: <https://www.wired.com/story/the-fantastic-geometry-of-greeces-fish-farms-2000-feet-up/> (accessed on 20 May 2017).