A Hybrid Interval Type-2 Semi-supervised Possibilistic Fuzzy c-Means Clustering and Particle Swarm Optimization for Satellite Image Analysis

Dinh Sinh Mai[a,b,c], Long Thanh Ngo[a,*], Le Hung Trinh[b], and Hani Hagras [c]

[a] *Faculty of Information Technology, Le Quy Don Technical University, Hanoi, Vietnam*
[b] *Institute of Techniques for Special Engineering, Le Quy Don Technical University, Hanoi, Vietnam*
[c] *The Computational Intelligence Centre, School of Computer Science and Electronic Engineering, University of Essex, UK*
e-mail: maidinhsinh@lqdtu.edu.vn, ngotlong@gmail.com, trinhlehung@lqdtu.edu.vn and hani@essex.ac.uk

Corresponding Author*: Long Thanh Ngo, Mobile: +84-914 364 064.

Abstract

Although satellite images can provide more information about the Earth's surface in a relatively short time and over a large scale, they are affected by observation conditions (e.g., wind, sun, rain, and humidity) and the accuracy of the image acquisition equipment. The objects on the images are often unclear and uncertain, especially at their borders. The fuzzy clustering technique allows each data pattern to belong to many different clusters through membership function (MF) values that can handle data patterns with unclear and uncertain boundaries well. However, this technique is quite sensitive to noise, outliers, and limitations in handling uncertainties. Furthermore, the membership degrees of type-1 fuzzy sets (T1FSs) are crisp, and in many cases, it is difficult to precisely determine the T1FS parameters. To overcome these disadvantages, we propose a hybrid method encompassing interval type-2 semi-supervised possibilistic fuzzy c-means clustering (IT2SPFCM) and particle swarm optimization (PSO) to form the proposed IT2SPFCMPSO. We experimented on several satellite images (Landsat-5 TM, Landsat-7 ETM+, Landsat-8, Sentinel-2A) to prove the effectiveness of the proposed method. Experimental results show that the IT2SPFCM-PSO algorithm achieves accuracies from 98.8% to 99.39%, which are higher than those of other matching algorithms. An analysis of the results by indicators PC-I, CE-I, D-I, XB-I, $\tau - I$, and MSE also showed that the proposed method achieves better results in most experiments.

Keywords: Interval type-2 fuzzy sets, semi-supervised, possibilistic fuzzy c-means, particle swarm optimization, satellite image, landcover

1. Introduction

Satellite image data have been widely used in many fields, such as mapping, monitoring natural resources, urban planning, weather forecasting, and many other tasks. In satellite image processing, clustering and classification is an important task that is the basis for advanced image analysis problems. This is because the image data obtained from the satellite are influenced by natural conditions (such as wind, rain, sunshine, etc.). Furthermore, the image's objects are affected by high uncertainty levels, especially around the boundary areas of objects.

There are two commonly used clustering methods: hard clustering and soft (fuzzy) clustering. For hard clustering, data patterns can belong to (probability equal to 1) or not belong to (probability equal to 0) only a single cluster. This method makes it difficult to handle data where the patterns can simultaneously belong to many clusters. Fuzzy sets were introduced by Zadeh in 1965. The algorithm used as the underlying theory for fuzzy clustering problems is the fuzzy c-means clustering (FCM) algorithm [1]. According to the FCM algorithm, the membership value indicates the possibility that the data sample will belong to a particular cluster.

In FCM, the membership degrees are usually calculated based on the distance between a data pattern and the cluster center. There are many ways to determine the distance between the data pattern and cluster centers, where the most common distance is the Euclidean distance. This distance is suitable when the clusters are spherical, but it is not useful in the case of intricate shapes and overlapping data. For each data

sample, the sum of the membership degree is equal to 1, and the large membership degree represents the data sample closer to the cluster centroid. However, the FCM is shown to be sensitive to noise and outliers [1].

There are many different approaches to improving the FCM algorithm [2, 3], such as using complementary information [4, 5, 6], using the semi-supervised method [7, 8, 9], and hybridization with other algorithms [10, 11, 12]. Recently, type-2 fuzzy sets (T2FSs) and interval type-2 fuzzy sets (IT2FSs) [13] have shown an advantage when handling uncertainties and have been developed and applied to many different problems [14, 15, 16], including remote sensing image analysis [17, 18].

However, FCM has some disadvantages, which can be listed as follows:

- Most fuzzy clustering algorithms (type-1, type-2) are based on the traditional Euclidean distance to determine the distance between data patterns while ignoring complementary information, which is often used for advanced image analysis.

- The original fuzzy clustering algorithm is unsupervised; hence, the data description and representation influence the clustering results.

- There are difficulties in determining parameters for fuzzy clustering algorithms, since these parameters are not fixed and depend on each data set's characteristics.

Among the types of satellite imagery, optical imagery is the most widely used in studying the Earth's surface at a low cost. However, one disadvantage is that they are often affected by weather conditions, and image data usually include noise and outliers. FCM was widely used in the past [20], where it has shown drawbacks such as sensitivity to noise and outliers, and it did not work well on uncertain data collected from satellite imagery.

A variant of the fuzzy clustering based on the possibilistic approach was first proposed in [22]. This method has advantages when data clusters have different characteristics (shape, size, area, etc.), but it is challenging to separate similar clusters. Nikhil et al. proposed a possibilistic fuzzy c-means (PFCM) algorithm [23]. The PFCM is a hybridization algorithm of the PCM and FCM algorithms, which takes advantage of both the above algorithms. Although PFCM can overcome the coincident cluster problem of PCM and the outliers of FCM and PFCM, the algorithms still have the disadvantages associated with the use of T1FS, such as difficulty in selecting parameters and sensitivity to noise. A generalized entropy-based PFCM algorithm (GEPFCM) was proposed by Askari et al. [24] for clustering noisy data. The main objective of GEPFCM is to determine accurate cluster centers of noisy data by generalizing entropy c-means (ECM) combined with PFCM.

Fuzzy clustering algorithms often consider the role of feature components to be the same. Usually, in remote sensing image analysis, different feature components may have different roles depending on each task. Yang et al. [25] presented a novel method for improving the FCM algorithm, which can consider the influence of feature components through weight values. This method builds a learning schema for fuzzy parameters with feature-weighted entropy to reduce the influence of irrelevant feature components.

Furthermore, standard fuzzy clustering is also sensitive to initializing centroids and selecting parameters because of the limitations of the optimization process. This methodology can be easily trapped in a local minimum [10, 26, 27]. To address this, Zhang et al. [11] proposed using the PSO technique [19] to find the optimal parameters for unsupervised fuzzy clustering. Lilin et al. [12] proposed an improved fuzzy clustering method based on a self-adaptive cell genetic algorithm.

One of the other popular approaches for improving the accuracy of fuzzy clustering algorithms is to use the semi-supervised method. The semi-supervised algorithm introduced by Yasunori et al. [32] can be seen as a typical algorithm using additional information to improve the fuzzy clustering algorithm's accuracy. Yin et al. [28] developed a novel semi-supervised metric-based fuzzy clustering algorithm called SMUC by introducing metric learning and entropy regularization simultaneously into the conventional fuzzy clustering algorithm based on prior membership degrees.

Mai and Long introduced a semi-supervised fuzzy c-means clustering algorithm (SFCM) in [7]. The additional information from labeled pixels is added in the objective function to adjust cluster centroids and reduce the ability to fall into local optima. Another semi-supervised clustering approach for the kernel fuzzy c-means algorithm (SSKFCM) was introduced by Zhang et al. [29], with global optimization obtained by

repeatedly updating the fuzzy memberships and the optimized kernel parameter. However, in kernel-based methods, it is often difficult to choose the number of kernels, which affects both accuracy and computational complexity.

Although widely used, existing fuzzy clustering methods still face some of the following issues: (1) the Euclidean distance tends to work poorly if the importance of the features is different, (2) it is difficult to determine the optimal parameter for the objective function, and (3) fuzzy clustering based on T1FSs cannot easily handle uncertainties because their MFs are crisp.

Type-2 fuzzy sets (T2FSs) are expanded from T1FSs [30], where T2FSs have been developed and applied in many fields, including satellite image classification [15, 33]. In T1FSs, each pattern has a membership grade that is a crisp number in $[0,1]$. T2FSs are characterized by fuzzy MFs, meaning that the membership grade for each pattern of a T2FS is a fuzzy set in $[0,1]$ [34]. The MFs of T2FSs are three dimensional and include a footprint of uncertainty (FOU), which provides additional degrees of freedom that make it possible to directly model and handle uncertainties. The T2FSs are useful when it is difficult to determine the exact MF for a fuzzy set, which helps incorporate uncertainties [31].

In [15], Melin et al. reviewed some applications of T2FSs in classification and pattern recognition and pointed out that the general T2FS is limited by high computational complexity [31] and difficulty in installation. Therefore, in practical applications, interval type-2 fuzzy sets (IT2FSs) are more widely used [35]. One of the ways to apply IT2FSs in clustering is to use the interval type-2 fuzzy c-means clustering (IT2FCM) algorithm [36, 37]. Some studies applying the IT2FCM algorithm for remote sensing image classification problems can be found in [17].

Recently, some studies have improved the IT2FCM algorithm for satellite image classification. Accordingly, in [38, 40], a new distance was introduced to replace the traditional Euclidean distance in the IT2FCM algorithm using spectral and spatial information in multispectral remote sensing image clustering. In [39], the SIIT2FCM algorithm was expanded from IIT2FCM [38] to change detection on multispectral satellite images that used spatial information and the semi-supervised method to improve the accuracy of classification results. Some studies developed the IT2FCM algorithm mentioned in [33, 41] using the multiple kernel technique for data classification.

Ji et al. [42] proposed a novel possibilistic fuzzy c-means clustering algorithm based on an interval-valued fuzzy set to model the uncertainties. This algorithm can significantly improve accuracy when compared with the original PFCM algorithm. Wang et al. [43] proposed a supervised classification method for high-resolution remote sensing images based on IT2FS by analyzing the data characteristics and building an interval type-2 MF to model the pixels' uncertainty. However, this algorithm requires a large amount of labeled data to train. An improvement on the IT2FCM algorithm for landcover classification from hyperspectral image data was proposed by Huo et al. [44], in which the interval type-2 fuzzy MF is ranked by confidence level based on the uncertainty of the spectral information.

The common point of the studies [38, 39, 45, 46] is that they all use information about the spatial relationship between pixels to improve the objective function, while MFs also have a significant influence on which cluster the pixels will belong to. The work reported in [45, 46] used type-1 fuzzy MFs limited to describing uncertain data. The studies used spatial weights to calculate each pixel's influence in the clustering process. Meanwhile, [38, 39] developed FCM algorithms based on IT2FSs, which could overcome the disadvantage of T1FSs in describing uncertain data. However, both of these studies used only fuzzy MFs.

The FCM and PFCM algorithms have also stated that the parameters used in these papers are not the right guarantees for all problems. The studies [38, 39, 45, 46] use parameters from the original papers, and the improvement in the objective function focuses only on cluster centroids while taking care of the values of MFs. Therefore, there is a need to find the optimal parameters for each problem to improve the classification and clustering results.

The above reviews show that T2FSs have more advantages than T1FSs in data clustering, including the satellite image analysis problem. Therefore, the research and development of algorithms based on T2FSs for the remote sensing image analysis problem is a move in the right direction and has excellent potential.

In this paper, we propose a hybrid interval type-2 semi-supervised possibilistic fuzzy c-means clustering approach that is combined with the PSO technique [19] to optimize parameters and overcome some of the above-mentioned disadvantages. There are two main reasons for using this approach. First, the number of labeled samples in a dataset is usually limited and insufficient for supervised classification methods. Second, remote sensing image data often have a large capacity and many bands, so the PSO technique can converge

3

faster on the search space than using other optimization techniques such as the genetic algorithm. Accordingly, T2FSs can help to handle uncertainties from satellite image data, while the PSO technique is used to find optimal parameters for the IT2SPFCM algorithm, and the semi-supervised method is introduced to contend with noise and outliers and to avoid falling into local minima. Because PFCM uses both fuzzy MF and possibilistic MF, once the PFCM algorithm is based on IT2FSs, they can model data better than IT2FCM.

This algorithm's main idea is to generalize the semi-supervised method and overcome difficulties in selecting optimal parameters by using hybrid methods with PSO technique on a limited number of labeled data samples. There are two main contributions from the IT2SPFCM-PSO algorithm. First, we have proposed a new objective function with tighter constraints than some previous methods to handle uncertainty data, noise, and outliers. Second, we recommend a hybrid of IT2SPFCM and PSO. Unlike some previous methods, the PSO technique is used as a pretreatment step to find the optimal parameters. For the IT2SPFCM-PSO algorithm, PSO is performed in each iteration of the IT2SPFCM. Once the IT2SPFCM is finished, the parameters are optimal.

Experiments were performed on many types of remote sensing images in different regions (Landsat-5 TM, Landsat-7 ETM+, Landsat-8, Sentinel-2A) for landcover classification and change detection. The experimental results show that the IT2SPFCM-PSO algorithm achieves better classification results in most cases than some previous methods. Specifically, the IT2SPFCM-PSO algorithm provides better classification results than the IT2SPFCM algorithm, while the computational complexity of the IT2SPFCM-PSO algorithm is lower than that of the IT2SPFCM algorithm.

The paper is organized as follows. Section II briefly introduces some background about PFCM, T2FSs, and IT2FCM. Section III proposes a hybrid algorithm of IT2SPFCM and PSO for landcover classification and change detection. Section IV offers some experimental results, and Section V draws conclusions and suggests future research directions.

2. Background

2.1. Fuzzy c-means clustering

The FCM algorithm has been widely applied to many different problems because it allows each data pattern to belong to two or more clusters [1]. This algorithm considers MF values based on the distance from each data pattern to cluster centroids. The FCM algorithm optimizes the objective function:

$$\min\{J_m(U,V,X) = \sum_{i=1}^{c} \sum_{k=1}^{n} \mu_{ik}^m d_{ik}^2\} \tag{1}$$

where $U = [\mu_{ik}]_{cxn}$ is a fuzzy MF, $V = (v_1, v_2, ..., v_c)$ is a vector of (unknown) cluster center, $X = \{x_k, x_k \in \mathbf{R}^M, k = 1, ..., n\}$, and $d_{ik} = \|v_i - x_k\|$. with the following constraints:

$$m > 1; 0 \leq \mu_{ik} \leq 1; \sum_{i=1}^{c} \mu_{ik} = 1; 1 \leq i \leq c; 1 \leq k \leq n \tag{2}$$

The objective function $J_m(U,V,X)$ reaches the smallest value when and only if:

$$v_i = \sum_{k=1}^{n} \mu_{ik}^m x_i / \sum_{k=1}^{n} \mu_{ik}^m \tag{3}$$

$$\mu_{ik} = 1 / \sum_{j=1}^{c} \left(d_{ik}/d_{jk}\right)^{2/(m-1)} \tag{4}$$

Equations 3 and 4 can be obtained based on the Lagrange multiplier theorem with the constraints in 2. The FCM algorithm will perform iterations according to Equations 3 and 4 until the objective function $J_m(U,V,X)$ reaches the minimum value; the computational complexity of this algorithm with t loops is $O(3tMnC)$.

4

2.2. Possibilistic c-means clustering

PCM algorithm proposed by Krishnapuram and Keller [22] was introduced to avoid the sensitivity of FCM algorithm. Instead of using fuzzy MFs such as FCM, PCM algorithm uses possibilistic MFs to represent typicality by τ_{ik}, the typicality matrix as $T = [\tau_{ik}]_{cxn}$, and $X = \{x_k, x_k \in R^M, k = 1,...,n\}$.

The PCM model is the constrained optimization problem:

$$\min\{J_\eta(T,V;X,\gamma) = \sum_{i=1}^{c}\sum_{k=1}^{n}\tau_{ik}^{\eta}d_{ik}^{2} + \sum_{i=1}^{c}\gamma_i\sum_{k=1}^{n}(1-\tau_{ik})^{\eta}\} \tag{5}$$

where $T = [\tau_{ik}]_{cxn}$ is a possibilistic MF, $V = (v_1, v_2,...,v_c)$ is a vector of cluster centers, $\gamma_i > 0$ is a user-defined constant. It has the following constraints:

$$\eta > 1; 0 \le \tau_{ik} \le 1; \sum_{k=1}^{n}\tau_{ik} = 1; 1 \le i \le c; 1 \le k \le n \tag{6}$$

The objective function $J_\eta(T,V;X,\gamma)$ reaches the smallest value when and only if:

$$v_i = \sum_{k=1}^{n}\tau_{ik}^{\eta}x_i / \sum_{k=1}^{n}\tau_{ik}^{\eta} \tag{7}$$

$$\tau_{ik} = 1 / \left(1 + (d_{ik}^2/\gamma_i)^{1/(\eta-1)}\right) \tag{8}$$

Krishnapuram and Keller also suggest using the results of the FCM algorithm as a good way to initialize the PCM algorithm, and the parameter γ_i should be calculated according to the following equation:

$$\gamma_i = K\sum_{k=1}^{n}\mu_{ik}^{\eta}d_{ik}^{2} / \sum_{k=1}^{n}\mu_{ik}^{\eta} \tag{9}$$

where μ_{ik} is the fuzzy membership from the results of the FCM algorithm, and K is a user-defined constant (usually selected by 1).

With constraint 6 and Equation 7 can be obtained by fixing T, then computing the derivative by v_i, and assigning zero to determine v_i; Equation 8 can be obtained by fixing V and finding the smallest value of the objective function $J_\eta(T,V;X,\gamma)$ by calculating the derivative γ_i, assigning zero to determine γ_i.

PCM algorithm will perform iterations according to Equations 7, 8 and 9 until the objective function $J_\eta(T,V;X,\gamma)$ reaches the minimum value. The computational complexity of this algorithm with t loops is $O(4tMnC)$.

2.3. Possibilistic fuzzy c-means clustering

FCM and PCM are the most popular approaches of fuzzy clustering and possibilistic clustering, respectively. However, there are disadvantages to these approaches, such as sensitivity to noise and difficulty working with overlapping data. The advantage of FCM is that it can work well with uncertain and approximate data, while its disadvantage is its sensitivity to noise and outliers. The PCM algorithm has the advantage in processing overlapping and noisy data, but it does not contend well with uncertain data.

The PFCM algorithm [23] is a hybrid algorithm between FCM and PCM, inheriting the advantages of both FCM and PCM. The PFCM algorithm has two types of MFs: the fuzzy MF in the FCM algorithm and the possibilistic MF in the PCM algorithm. PFCM model is the constrained optimization problem:

$$\min\{J_{m,\eta}(U,T,V,X,\gamma) = \sum_{i=1}^{c}\sum_{k=1}^{n}(a\mu_{ik}^{m} + b\tau_{ik}^{\eta})d_{ik}^{2} + \sum_{i=1}^{c}\gamma_i\sum_{k=1}^{n}(1-\tau_{ik})^{\eta}\} \tag{10}$$

where $X = \{x_k, x_k \in R^M, , k = 1,...,n\}$, $U = [\mu_{ik}]_{cxn}$ is a fuzzy partition matrix that contains the fuzzy membership degree; $T = [\tau_{ik}]_{cxn}$ is a typicality partition matrix that contains the possibilistic membership degree; $V = (v_1, v_2,...,v_c)$ is a vector of cluster centers, m is the weighting exponent for the fuzzy partition matrix and η is the weighting exponent for the typicality partition matrix, $\gamma_i > 0$ are constants given by the user.

5

The PFCM model is subject to the following constraints:

$$m, \eta > 1; a, b > 0; 0 \le \mu_{ik}, \tau_{ik} \le 1; \sum_{i=1}^{c} \mu_{ik} = 1; \sum_{k=1}^{n} \tau_{ik} = 1; 1 \le i \le c; 1 \le k \le n \tag{11}$$

The objective function $J_{m,\eta}(U,T,V,X)$ reaches the smallest value with constraints 11 when and only if

$$v_i = \frac{\sum_{k=1}^{n} (a\mu_{ik}^m + b\tau_{ik}^{\eta})x_i}{\sum_{k=1}^{n} (a\mu_{ik}^m + b\tau_{ik}^{\eta})} \tag{12}$$

$$\mu_{ik} = 1 / \sum_{j=1}^{c} \left(d_{ik}^2 / d_{jk}^2 \right)^{2/(m-1)} \tag{13}$$

$$\tau_{ik} = 1 / \left(1 + (bd_{ik}^2 / \gamma_i)^{1/(\eta-1)} \right) \tag{14}$$

in which, with constraint 11, Equations 12 and 13 are achieved in the same way as FCM algorithm, and Equation 14 is achieved in the same way as PCM algorithm.

The PFCM algorithm will perform iterations according to Equations 12, 13, 14 and 9 until the objective function $J_{m,\eta}(U,T,V,X)$ reaches the minimum value, and the computational complexity of this algorithm with t loops is $O(5tMnC)$. When n is large, the computational complexity of FCM, PCM, and PFCM algorithms is the same.

Despite the advantages of both FCM and PCM algorithms, PFCM uses only type-1 fuzzy MFs and type-1 possibilistic MFs, so it does not fully describe the characteristics of the data, especially with data that have high uncertainty. This is also the reason why many recent studies have developed fuzzy clustering algorithms based on type-2 fuzzy sets.

2.4. Type-2 fuzzy set

A type-2 fuzzy set in X is denoted $\tilde{A}$, and its membership grade of $x \in X$ is $\mu_{\tilde{A}}(x,u), u \in J_x \subseteq [0,1]$, which is a type-1 fuzzy set in $[0,1]$. The elements of the domain of $\mu_{\tilde{A}}(x,u)$ are called primary memberships of x in $\tilde{A}$ and memberships of primary memberships in $\mu_{\tilde{A}}(x,u)$ are called secondary memberships of x in $\tilde{A}$.

A type-2 fuzy set [31], denoted $\tilde{A}$, is characterized by a type-2 MF $\mu_{\tilde{A}}(x,u)$ where $x \in X$ and $u \in J_x \subseteq [0,1]$, i. e.

$$\tilde{A} = \{((x,u), \mu_{\tilde{A}}(x,u)) | \forall x \in X, \forall u \in J_x \subseteq [0,1]\} \tag{15}$$

or

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x,u)) / (x,u), J_x \subseteq [0,1] \tag{16}$$

in which $0 \le \mu_{\tilde{A}}(x,u) \le 1$.

At each value of x, say $x = x'$, the 2-D plane whose axes are u and $\mu_{\tilde{A}}(x',u)$ is called a vertical slice of $\mu_{\tilde{A}}(x,u)$. A secondary MF is a vertical slice of $\mu_{\tilde{A}}(x',u)$, it is $\mu_{\tilde{A}}(x = x', u)$ for $x \in X$ and $\forall u \in J_{x'} \subseteq [0,1]$, i. e.

$$\mu_{\tilde{A}}(x = x', u) \equiv \mu_{\tilde{A}}(x') = \int_{u \in J_{x'}} f_{x'}(u)/u, J_{x'} \subseteq [0,1] \tag{17}$$

in which $0 \le f_{x'}(u) \le 1$.

Type-2 fuzzy sets are called interval type-2 fuzzy sets if the secondary MF $f_{x'}(u) = 1 \, \forall u \in J_x$ i. e. a type-2 fuzzy set are defined as follows: An interval type-2 fuzzy set $\tilde{A}$ is characterized by an interval type-2 MF $\mu_{\tilde{A}}(x,u) = 1$ where $x \in X$ and $u \in J_x \subseteq [0,1]$, i. e. ,

$$\tilde{A} = \{((x,u), 1) | \forall x \in X, \forall u \in J_x \subseteq [0,1]\} \tag{18}$$

The FOU is union of primary functions i. e. $FOU(\tilde{A}) = \bigcup_{x \in X} J_x$. The upper/lower bounds of MF (UMF/LMF), denoted $\bar{\mu}_{\tilde{A}}(x)$ and $\underline{\mu}_{\tilde{A}}(x)$, of $\tilde{A}$ are two type-1 MFs and bounds of FOU.
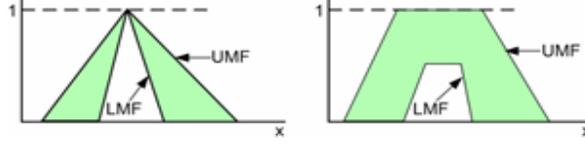
6

Figure 1: The MF of an interval type 2 fuzzy set [14]

The IT2FCM is extension of FCM clustering by using two fuzziness parameters, m_1, m_2, to make FOU, corresponding to upper and lower values of fuzzy clustering [36]. The use of fuzzifiers yields different objective functions to be minimized as follows:

$$J_{m_1}(U,V,X) = \sum_{k=1}^{N}\sum_{i=1}^{C} u_{ik}^{m_1} d_{ik}^2 \quad and \quad J_{m_2}(U,V,X) = \sum_{k=1}^{N}\sum_{i=1}^{C} u_{ik}^{m_2} d_{ik}^2 \tag{19}$$

in which $d_{ik} = |x_k - v_i|$ is the Euclidean distance between the pattern x_k and the centroid v_i, C is number of clusters and N is number of patterns. The upper/lower degrees of membership, $\bar{u}_{ik}$ and $\underline{u}_{ik}$ are determined as follows:

$$\bar{u}_{ik} = \begin{cases} 1/\sum_{j=1}^{C}\left(d_{ik}/d_{jk}\right)^{2/(m_1-1)} & if\, 1/\sum_{j=1}^{C}\left(d_{ik}/d_{jk}\right) < \frac{1}{C} \\ 1/\sum_{j=1}^{C}\left(d_{ik}/d_{jk}\right)^{2/(m_2-1)} & otherwise \end{cases} \tag{20}$$

$$\underline{u}_{ik} = \begin{cases} 1/\sum_{j=1}^{C}\left(d_{ik}/d_{jk}\right)^{2/(m_1-1)} & if\, 1/\sum_{j=1}^{C}\left(d_{ik}/d_{jk}\right) \geq \frac{1}{C} \\ 1/\sum_{j=1}^{C}\left(d_{ik}/d_{jk}\right)^{2/(m_2-1)} & otherwise \end{cases} \tag{21}$$

in which $i = \overline{1,C}$, $k = \overline{1,N}$.

Because each pattern has membership interval as the upper $\bar{u}_{ik}$ and the lower $\underline{u}_{ik}$, each cluster centroid is represented by the interval between v^R and v^L. The algorithm for finding cluster centroids is the enhanced iterative algorithm and stopping condition (EIASC) algorithm [36]. This algorithm has been shown to significantly reduce the time of cluster centroid determination compared to KM and EKM algorithms [36]. The algorithm for finding v_i^R is described as follows:

Algorithm 1: EIASC algorithm for finding v_i^R [36]

Input: Dataset $X = \{x_k, x_k \in R^M, k = 1,...,n\}$, the number of clusters $c(1 < c < n)$, fuzzifier parameters m_1, m_2, m.

Step 1: Without a loss of generality, assume that n patterns are sorted on each of M features in ascending order: $x_1 \leq x_2 \leq ... \leq x_n$, note is $\bar{u}_{ik}, \underline{u}_{ik}, u_{ik}$ will also change the order corresponding to $x_1 \leq x_2 \leq ... \leq x_n$.

Step 2: Compute $\bar{u}_{ik}, \underline{u}_{ik}$ by using Equations 20 and 21.

Step 3: Initialize $a_i = \sum_{k=1}^{n} \bar{u}_{ik} x_k$; $b_i = \sum_{k=1}^{n} \bar{u}_{ik}; t = n+1$

Step 4: $t = t - 1$ and compute

$$\begin{aligned} a_i &= a_i + x_t(\bar{u}_{it} - \underline{u}_{it}) \\ b_i &= b_i + (\bar{u}_{it} - \underline{u}_{it}) \\ v_i^R &= a_i/b_i \end{aligned} \tag{22}$$

Step 5: If $(v_i^R > x_t)$ stops, **then** go to **Step 4**.
Output: The centroid matrices v_i^R.

Similar to **Algorithm 1**, the algorithm for finding v_i^L is described as follows:

Algorithm 2: EIASC algorithm for finding v_i^L [36]

Input: Dataset $X = \{x_k, x_k \in R^M, k = 1,...,n\}$, the number of clusters $c(1 < c < n)$, fuzzifier parameters m_1, m_2, m.

Step 1: Without a loss of generality, assume that n patterns are sorted on each of M features in ascending order: $x_1 \le x_2 \le ... \le x_n$, note is $\bar{u}_{ik}, \underline{u}_{ik}, u_{ik}$ will also change the order corresponding to $x_1 \le x_2 \le ... \le x_n$.

Step 2: Compute $\bar{u}_{ik}, \underline{u}_{ik}$ by using Equations 20 and 21.

Step 3: Initialize $a_i = \sum\limits_{k=1}^{n} \underline{u}_{ik} x_k$; $b_i = \sum\limits_{k=1}^{n} \underline{u}_{ik}; t = 0$

Step 4: Compute $t = t + 1$

$$
\begin{aligned}
a_i &= a_i + x_t(\bar{u}_{it} - \underline{u}_{it}) \\
b_i &= b_i + (\bar{u}_{it} - \underline{u}_{it}) \\
v_i^L &= a_i / b_i
\end{aligned}
\tag{23}
$$

Step 5: If $(v_i^L \le x_{t+1})$ stops, **then** return **Step 4**.

Output: The centroid matrices v_i^L.

After obtaining v_i^R, v_i^L, a type-reduction operator is applied to obtain the centroid of the ith cluster. We defuzzify the interval set by using average of v_i^R and v_i^L as follows:

$$
v_i = (v_i^R + v_i^L)/2
\tag{24}
$$

For membership grades:

$$
u_i(x_k) = (u_i^R(x_k) + u_i^L(x_k))/2, i = 1,...,C
\tag{25}
$$

in which

$$
u_i^L = \sum_{l=1}^{M} u_{il}/M, u_{il} = \begin{cases} \bar{u}_i(x_k) & \text{if} & x_{il} \quad \text{uses } \bar{u}_i(x_k) \text{for} v_i^L \\ \underline{u}_i(x_k) & & \text{otherwise} \end{cases}
\tag{26}
$$

$$
u_i^R = \sum_{l=1}^{M} u_{il}/M, u_{il} = \begin{cases} \bar{u}_i(x_k) & \text{if} & x_{il} \quad \text{uses } \bar{u}_i(x_k) \text{for} v_i^R \\ \underline{u}_i(x_k) & & \text{otherwise} \end{cases}
\tag{27}
$$

Cluster centroids are computed in the same way as the FCM as follows:

$$
v_i = \sum_{k=1}^{N} u_{ik}^m x_k / \sum_{k=1}^{N} u_{ik}^m
\tag{28}
$$

in which $i = \overline{1,C}$.

Next, defuzzification for IT2FCM is performed as if $u_i(x_k) > u_j(x_k)$ for $j = 1,...,C$ and $i \neq j$; then, x_k is assigned to cluster i. Detailed implementation steps of IT2FCM algorithm are described in **Algorithm 3**.

Algorithm 3: IT2FCM algorithm

Input: Dataset $X = \{x_k, x_k \in R^M, k = 1,...,n\}$, the number of clusters $C(1 < C < n)$, fuzzifier parameters m_1, m_2, m, $t = 0$;

Step 1: Initialize the centroid matrix $V^{(t)} = [v_i^{(t)}], V^{(t)} \in R^{MxC}$ by choosing randomly from the input dataset X.

Step 2: Compute $U^{(t)}$ by using Equations 20, 21, 25, 26 and 27.

Step 3: Repeat

    3.1 $t = t + 1$

    3.2 Update the centroid matrix $V^{(t)} = [v_i^{(t)}]$ by using **Algorithms 1&2**.

    3.3 Compute $U^{(t)}$ by using Equations 20, 21, 25, 26 and 27.

    3.4 Assign data pattern x_k to the ith cluster if $u_{ik} \ge u_{jk}, j = 1,...,C; j \neq i$.

    3.5 Check condition $max(\left\| U^{(t+1)} - U^{(t)} \right\|) \le \varepsilon$. **If** yes **then** stop; **otherwise** return **Step 3**.

Output: The membership matrix U and the centroid matrix V.

Defuzzification: Assign data pattern x_k to the ith cluster if $u_{ik} \ge u_{jk}, j = 1,...,C; j \neq i$.

Computational complexity: In *Step 2* the computational complexity is $O(5nMC)$. In each iteration we will include *Step 3.1* the computational complexity is $O(1)$. In *Step 3.2*, the centroid matrix $V^{(t)} = [v_1^{(t)}, v_2^{(t)}, ..., v_C^{(t)}]$ is updated, and the right and left centroid are calculated according to Algorithms 1&2. This algorithm sorts n patterns on each of M features in ascending order: $x_1 \leq x_2 \leq ... \leq x_n$ and then executes the M loop to find v^L and v^R. Therefore, using the quick-sort algorithm, the computational complexity of Algorithms 1&2 is $O(MCnlogn + MCn)$. In *Step 3.3*, $U^{(t)}$ is computed by using equations 20, 21, 26, 27 and 28; thus, the computational complexity is $O(5nMC)$. In *Step 3.4*, the computational complexity is $O(nC)$. In *Step 3.5*, the computational complexity is $O(1)$. In each loop, there is computational complexity $O(MCnlogn + 6nMC + nC + 2)$. When *n* is large and the number of iterations of the algorithm is t, the computational complexity of IT2FCM algorithm is $O(tMCnlogn + 6tMCn + tCn + 5MCn)$.

3. Hybrid Interval Type-2 Semi-supervised Possibilistic Fuzzy c-Means Clustering and Particle Swarm Optimization

3.1. Interval type-2 Semi-supervised PFCM

In this section, the paper presents a hybrid algorithm of semi-supervised possibilistic fuzzy c-means clustering and PSO technique based on IT2FS. Typically, supervised methods require a large amount of labeled data for training, but this method can be applied in cases where there is very little labeled data.

Dataset $X = \{x_k, x_k \in R^M, k = 1,...,n\}$ with some labeled data in clusters. Let *c* be the number of clusters; the calculate of *c* centroids $v_1^*, v_2^*, ..., v_c^*$ is from the labeled pixel dataset, and $V^* = [v_1^*, v_2^*, ..., v_c^*]$ is the set of additional cluster centroids, which is averaged from the labeled data as follows:

$$v_i^* = \sum_{s=1}^{m_i} P_{is}/N_i \tag{29}$$

where P_{is} is the s_{th} labeled pixel on the *i* cluster, and N_i is the number of labeled pixels on the *i* cluster, $s = 1, 2, ..., N_i; i = 1, 2, ..., c$. The additional fuzzy MF is calculated based on a set of additional centroids V^* by the FCM algorithm.

$$\mu_{ik}^* = 1/\sum_{z=1}^{c} \left(\frac{x_k - v_i^*}{x_k - v_z^*}\right)^{2/(m-1)} \tag{30}$$

where γ_i is calculated according to Equation 9. The additional possibilistic MF is calculated based on a set of additional centroids V^* by PCM algorithm.

$$\tau_{ik}^* = 1/\left(1 + (b\|v_i^* - x_k\|)^{1/(\eta-1)}/\gamma_i\right) \tag{31}$$

We propose a new objective function by adding additional information including the MFs μ_{ik}^*, τ_{ik}^* and cluster centroids v_i^*. This new objective function $J_{m,\eta}(U, T, V, X, \gamma)$ is

$$\min \left\{ \begin{array}{l} J_{m,\eta} = \sum_{i=1}^{c} \sum_{k=1}^{n} (a\|\mu_{ik} - \mu_{ik}^*\|^m + b\|\tau_{ik} - \tau_{ik}^*\|^\eta)(\|v_i - x_k\|^2 + \delta\|v_i - v_i^*\|^2) \\ + \sum_{i=1}^{c} \gamma_i \sum_{k=1}^{n} (1 - \tau_{ik})^\eta \end{array} \right\} \tag{32}$$

and is subject to the constraint

$$0 \leq \mu_{ik}, \tau_{ik} \leq 1; \sum_{i=1}^{c} \mu_{ik} = 1; \sum_{k=1}^{n} \tau_{ik} = 1; 1 \leq i \leq c; 1 \leq k \leq n \tag{33}$$

and

$$D_{ik}^2 = \|v_i - x_k\|^2 + \delta\|v_i - v_i^*\|^2 \tag{34}$$

where $\delta \geq 0$ is a user-defined constant that represents the role of the additional centroid value in the objective function, and $\delta = 0$ when v_i^* does not exist, $m, \eta > 1; a, b > 0$.

Expanding Equation 32 by using two fuzziness parameters, m_1, m_2, and two possibilistic parameters, η_1, η_2, to make FOU correspond to upper and lower values of fuzzy clustering and possibilistic clustering. We propose a new algorithm called interval type-2 semi-supervised possibilistic fuzzy c-means clustering (IT2SPFCM).

9

The use of m_1, m_2 and η_1, η_2 yields different objective functions $J_{m_1,\eta_1}(U,T,V,X,\gamma)$ and $J_{m_2,\eta_2}(U,T,V,X,\gamma)$ to be minimized as follows:

$$J_{m_1,\eta_1} = \sum_{i=1}^{c} \sum_{k=1}^{n} (a\|\mu_{ik}-\mu_{ik}^*\|^{m_1} + b\|\tau_{ik}-\tau_{ik}^*\|^{\eta_1})D_{ik}^2 + \sum_{i=1}^{c} \gamma_i \sum_{k=1}^{n} (1-\tau_{ik})^{\eta_1}$$
$$J_{m_2,\eta_2} = \sum_{i=1}^{c} \sum_{k=1}^{n} (a\|\mu_{ik}-\mu_{ik}^*\|^{m_2} + b\|\tau_{ik}-\tau_{ik}^*\|^{\eta_2})D_{ik}^2 + \sum_{i=1}^{c} \gamma_i \sum_{k=1}^{n} (1-\tau_{ik})^{\eta_2}$$
(35)

and is subject to the constraints

$$m_1, \eta_1, m_2, \eta_2 > 1; a, b > 0; \delta \geq 0; 0 \leq \mu_{ik}, \tau_{ik} \leq 1;$$
$$\sum_{i=1}^{c} \mu_{ik} = 1; \sum_{k=1}^{n} \tau_{ik} = 1; 1 \leq i \leq c; 1 \leq k \leq n$$
(36)

Theorem 3.1. *Theorem IT2SPFCM*

For $X = \{x_k, x_k \in \mathbb{R}^M, k = 1,...,n\}$, $m, \eta > 1; c \geq 1, \delta \geq 0$ and X contains at least c distinct data points. With constraints 36 and Equation 33 then $J_{m_1,\eta_1}(U,T,V,X,\gamma)$ and $J_{m_2,\eta_2}(U,T,V,X,\gamma)$ can minimize only if:

$$\mu_{ik}^{(1)} = \begin{cases} \mu_{ik}^* + \dfrac{(1-\sum_{i=1}^{c}\mu_{ik}^*)[1/D_{ik}^2]^{1/(m_1-1)}}{\sum_{i=1}^{c}[1/D_{ik}^2]^{1/(m_1-1)}} & if \dfrac{1}{\sum_{j=1}^{C}(D_{ik}/D_{jk})} < \dfrac{1}{c} \\[4mm] \mu_{ik}^* + \dfrac{(1-\sum_{i=1}^{c}\mu_{ik}^*)[1/D_{ik}^2]^{1/(m_2-1)}}{\sum_{i=1}^{c}[1/D_{ik}^2]^{1/(m_2-1)}} & otherwise \end{cases}$$
(37)

$$\mu_{ik}^{(2)} = \begin{cases} \mu_{ik}^* + \dfrac{(1-\sum_{i=1}^{c}\mu_{ik}^*)[1/D_{ik}^2]^{1/(m_1-1)}}{\sum_{i=1}^{c}[1/D_{ik}^2]^{1/(m_1-1)}} & if \dfrac{1}{\sum_{j=1}^{C}(D_{ik}/D_{jk})} \geq \dfrac{1}{c} \\[4mm] \mu_{ik}^* + \dfrac{(1-\sum_{i=1}^{c}\mu_{ik}^*)[1/D_{ik}^2]^{1/(m_2-1)}}{\sum_{i=1}^{c}[1/D_{ik}^2]^{1/(m_2-1)}} & otherwise \end{cases}$$
(38)

where

$$\bar{\mu}_i(x_k) = max\{\mu_{ik}^{(1)}, \mu_{ik}^{(2)}\} \quad and \quad \underline{\mu}_i(x_k) = min\{\mu_{ik}^{(1)}, \mu_{ik}^{(2)}\}$$
(39)

$$\tau_{ik}^{(1)} = \begin{cases} \left(\tau_{ik}^* + [\gamma_i/bD_{ik}^2]^{1/(\eta_1-1)}\right) / \left(1 + [\gamma_i/bD_{ik}^2]^{1/(\eta_1-1)}\right) & \tau_{ik} \geq \tau_{ik}^* \\[3mm] \left(\tau_{ik}^* - [\gamma_i/bD_{ik}^2]^{1/(\eta_1-1)}\right) / \left(1 - [\gamma_i/bD_{ik}^2]^{1/(\eta_1-1)}\right) & else \end{cases}$$
(40)

$$\tau_{ik}^{(2)} = \begin{cases} \left(\tau_{ik}^* + [\gamma_i/bD_{ik}^2]^{1/(\eta_2-1)}\right) / \left(1 + [\gamma_i/bD_{ik}^2]^{1/(\eta_2-1)}\right) & \tau_{ik} \geq \tau_{ik}^* \\[3mm] \left(\tau_{ik}^* - [\gamma_i/bD_{ik}^2]^{1/(\eta_2-1)}\right) / \left(1 - [\gamma_i/bD_{ik}^2]^{1/(\eta_2-1)}\right) & else \end{cases}$$
(41)

where

$$\bar{\tau}_i(x_k) = max\{\tau_{ik}^{(1)}, \tau_{ik}^{(2)}\} \quad and \quad \underline{\tau}_i(x_k) = min\{\tau_{ik}^{(1)}, \tau_{ik}^{(2)}\}$$
(42)

Because each pattern has a membership interval as the upper $\bar{\mu}, \bar{\tau}$ and the lower $\underline{\mu}, \underline{\tau}$, each cluster centroid is represented by the interval between v^L and v^R.

For possibilistic membership grades:

$$\tau_i(x_k) = (\bar{\tau}_i(x_k) + \underline{\tau}_i(x_k))/2; i = 1,...,C; k = 1,...,n$$
(43)

Proof:

Equations 37 and 38 are similar to Equations 20 and 21 in the IT2FCM algorithm achieved by using the Lagrange multiplier and adding additional information. When $\mu_{ik}^* = 0(\mu_{ik}^*$ does not exist or is not used), considering that the distance D_{ik} is similar to the distance d_{ik}, Equations 37 and 38 degenerate to Equations 20 and 21 in the IT2FCM algorithm.

Equations 40 and 41 are derived by evaluating the minimum problem for the objective function 35, with V and U fixed by minimum problem

$$min\left\{J_{m_1,\eta_1}(T) = (a\|\mu_{ik} - \mu_{ik}^*\|^{m_1} + b\|\tau_{ik} - \tau_{ik}^*\|^{\eta_1})D_{ik}^2 + \gamma_i(1 - \tau_{ik})^{\eta_1}\right\} \tag{44}$$

and

$$min\left\{J_{m_2,\eta_2}(T) = (a\|\mu_{ik} - \mu_{ik}^*\|^{m_2} + b\|\tau_{ik} - \tau_{ik}^*\|^{\eta_2})D_{ik}^2 + \gamma_i(1 - \tau_{ik})^{\eta_2}\right\} \tag{45}$$

When $\tau_{ik}^* = 0$ (τ_{ik}^* does not exist or is not used), considering that the distance D_{ik} is similar to the distance d_{ik} in Equation 8 or Equation 14 with the exception of using two possibilistic parameters η_1, η_2.

Because each pattern has both membership interval values as the upper $\bar{\mu}, \bar{\tau}$ and the lower $\underline{\mu}, \underline{\tau}$; each cluster centroid is represented by an interval between v^R and v^L. We use Algorithms 1&2 to find the centroids v^R and v^L, where $V = [v_i]$ is computed in the same way as in the IT2FCM algorithm in Equation 25.

In Equations 37, 38, 40, 41 and 35, we can use both MFs μ_{ik}^*, τ_{ik}^* and the centroid v_i^*, or use one of them depending on the additional information obtained. It is easy to see that, when $\mu_{ik}^*, \tau_{ik}^*, v_i^*$ are not used and $\mu_{ik}^* = 0, \tau_{ik}^* = 0, v_i^* = 0$. The correctness of the theorem IT2SPFCM follows exactly as it does in IT2FCM algorithm. The implementation steps of IT2SPFCM algorithm are similar to those in the IT2FCM, details of the steps are as follows:

Algorithm 4: IT2SPFCM algorithm

Input: Dataset $X = \{x_k, x_k \in R^M, k = 1,...,n\}$; the labeled data set $X^* = \{P_{is}, P_{is} \in R^M, s << n; i = 1,...,C\}$; the number of clusters $C(1 < C < n)$, fuzzifier parameters $m_1, m_2, m, \eta_1, \eta_2, \eta, T_{max}$; and $t = 0$.

Step 1: Compute the additional cluster centroids $V^* = [v_i^*]$ by using Equation 29, the additional fuzzy MF $U^* = [\mu_{ik}^*]$ by using Equation 30, and the additional possibilistic MF $T^* = [\tau_{ik}^*]$ by using Equations 9 and 31.

Step 2: Initialize the centroid matrix $V^{(t)} = [v_i^{(t)}], V^{(t)} \in R^{MxC}$ by choosing randomly from the input dataset X.

Step 3: Compute $U^{(t)}$ by using Equations 37, 38, 39, 26,27, and 28.

Step 4: Compute $T^{(t)}$ by using Equations 9, 40, 41,42, and 43.

Step 5: Repeat

    5.1 $t = t + 1$

    5.2 Compute the centroids v^R and v^L by using Algorithms 1&2.

    5.3 Update the centroid matrix $V^{(t)} = [v_i^{(t)}]$ by using Equation 24.

    5.4 Update $U^{(t)}$ by using Equations 37, 38, 39, 26,27, and 28.

    5.5 Update $T^{(t)}$ by using Equations 9, 40, 41,42, and 43.

    5.6 Assign data pattern x_k to the ith cluster if $u_{ik} \geq u_{jk}, j = 1,...,C; j \neq i$.

    5.7 Check **if** $max(\|U^{(t+1)} - U^{(t)}\| + \|T^{(t+1)} - T^{(t)}\|) \leq \varepsilon$ or $t \geq T_{max}$. **If** yes **then** stop and **goto** Output **otherwise** return Step 5.1.

Output: The membership matrix U, T and the centroid matrix V.

Defuzzification: Assign the data pattern x_k to the ith cluster if $u_{ik} \geq u_{jk}, j = 1,...,C; j \neq i$.

Similar to PFCM algorithm, IT2SPFCM algorithm will execute a conditional loop, when either of the conditions $max(\|U^{(t+1)} - U^{(t)}\| + \|T^{(t+1)} - T^{(t)}\|) \leq \varepsilon$ or $t \geq T_{max}$ comes first, the algorithm will stop and give the classification result.

Computational complexity: When each iteration includes *Step 5.1*, the computational complexity is $O(1)$. In *Step 5.2*, the right and the left centroids are calculated according to Algorithms 1&2. This algorithm sorts n patterns on each of the M features in ascending order: $x_1 \leq x_2 \leq ... \leq x_n$. Then the M loop is executed to find v^L and v^R. Using the quick-sort algorithm, the computational complexity of Algorithms 1&2 is $O(MCnlogn + MCn)$. In *Step 5.3*, the centroid matrix is updated as $V^{(t)} = [v_1^{(t)}, v_2^{(t)}, ..., v_C^{(t)}]$ by using Equation 25, and the computational complexity is $O(MC)$.

In *Step 5.4*, Equations 37 and 38 have a computational complexity of $O(2nMC)$, and Equations 39, 26,27 and 28 have a computational complexity of $O(nC)$. In *Step 5.5*, Equations 9, 40, and 41 are similar to the PFCM algorithm, and the complexity is $O(3nMC)$, while Equations 42 and 43 have a complexity of $O(nC)$.

In *Step 5.6*, the computational complexity is $O(C)$. In *Step 5.7*, the computational complexity is $O(1)$. In each loop, there is a computational complexity of $O(MCnlogn + 6nMC + MC + nC + C + 2)$.

When n is large and the number of iterations of the algorithm is t, the computational complexity of IT2SPFCM algorithm is $O(tMCnlogn + 6tMCn + Cn)$ and equivalent to that of the IT2FCM algorithm.

3.2. The hybrid algorithm between IT2SPFCM and PSO

In real applications, the above-mentioned algorithms often have difficulty in initializing the parameters. These are usually not fixed but change according to the characteristics of each data set. This means that the parameters may be suitable on this data set, but it is uncertain how well they can work on other data sets. In this study, we propose a method to find parameters using the PSO technique [19]. This algorithm has the advantage of simple installation and is suitable for large data sets.

An important point at which to start the PSO algorithm is with the initialization of particles. Usually, in satellite image classification, the number of clusters is determined by the user based on the number of landcovers in the image. The parameters of the IT2SPFCM algorithm need to find the optimal value, including the centroid of clusters and the parameters $m, m_1, m_2, \eta, \eta_1, \eta_2, a, b$. For satellite images that have has M spectrum bands ($M = 3$ in an *RGB* color image) with C number of clusters, the total number of particles to be initialized is $M * C + 8$ (see Equation 46).

$$\underbrace{v_{11}, v_{12}, ...v_{1M}}_{V_1} \quad \underbrace{v_{21}, v_{22}, ...v_{2M}}_{V_2} \quad ... \quad \underbrace{v_{C1}, v_{C2}, ...v_{CM}}_{V_C} \quad \underbrace{m, m_1, m_2, \eta, \eta_1, \eta_2, a, b}_{\text{parameters}} \tag{46}$$

where $v_i = [v_{ij}]$ represents cluster centroids ($i = 1, ..., c; j = 1, ..., M$); and $m, m_1, m_2, \eta, \eta_1, \eta_2$ are fuzzy and possibilistic parameters; and a, b are user-defined parameters.

Let $P = (p_1, p_2, ..., p_{C*M}, p_{C*M+1}, ..., p_{C*M+8})$ be the set of all particle positions; $p_1, p_2, ..., p_{C*M}$ represent cluster centroids; $p_{C*M+1}, p_{C*M+2}, p_{C*M+3}$ represent fuzzy parameters; $p_{C*M+4}, p_{C*M+5}, p_{C*M+6}$ represent possibilistic parameters; p_{C*M+7}, p_{C*M+8} represent parameters a, b, respectively. With each particle, there will be position and movement velocity. The position of a particle is usually randomly generated in the search space. Each particle will include the following information: p_i, the current position of the ith particle; vel_i the current velocity of the ith particle; and $pBest_i$, the personal best position of the ith particle.

With the objective function F, the personal best position of a particle at time t is updated as:

$$pBest_i^{(t+1)} = \begin{cases} pBest_i^{(t)} & \text{if} \quad F(p_i^{(t+1)}) \geq F(pBest_i^{(t)}) \\ p_i^{(t+1)} & \text{if} \quad F(p_i^{(t+1)}) < F(pBest_i^{(t)}) \end{cases} \tag{47}$$

With the entire population, the best position of the population is denoted by *gBest*:

$$gBest^{(t)} = \{pBest_i^{(t)} | F(pBest_i^{(t)}) = \min\{F(pBest_1^{(t)}), F(pBest_2^{(t)}), ..., F(pBest_{C*M+4}^{(t)})\}\} \tag{48}$$

For each iteration of the PSO algorithm, p_i and vel_i are updated as follows:

$$vel_i^{(t+1)} = \omega * vel_i^{(t)} + c_1 * r_1 * (pBest_i^{(t)} - p_i^{(t)}) + c_2 * r_2 * (gBest^{(t)} - p_i^{(t)})$$
$$p_i^{(t+1)} = p_i^{(t)} + vel_i^{(t+1)} \tag{49}$$

An important issue in the PSO algorithm is the selection of parameters. Parameters c_1 and c_2 represent the influence of the best particle solution and the best global solution. These two parameters are normally set to 2.05 as suggested in the original paper [19]. ω is the inertia parameter, which indicates the rate of change in velocity of the particle during moving, common values range from zero to one. r_1, r_2 are random numbers in the range $(0, 1)$.

In the PSO algorithm loops, every particle must always be in the search space with conditions:

- $p_{\min} \leq p_i \leq p_{\max}, i = 1, ..., C * M$ and $p_{C*M+1}, ..., p_{C*M+6} > 1$ and $p_{C*M+7}, p_{C*M+8} > 0$

- If $p_i < p_{\min}$ then $p_i = p_{\min}$ and if $p_i > p_{\max}$ then $p_i = p_{\max}$

12

in which $i = 1, \ldots, C * M + 8$.

Similar to the velocity of particles, $vel_{\min} \leq vel_i \leq vel_{\max}, \forall i$ is set to be the velocity limits of the particles, in which $vel_{\min}, vel_{\max}$ are selected by user experience. A constraint is given, if $vel_i < vel_{\min}$ then $vel_i = vel_{\min}$ and if $vel_i > vel_{\max}$ then $vel_i = vel_{\max}$.

Similar to IT2SPFCM algorithm, it is necessary to define an objective function for IT2PFCM-PSO algorithm. The hybrid algorithm between IT2SPFCM and PSO is considered to be the following objective function:

$$F_{m_1,\eta_1,m_2,\eta_2}(U,T,V,X,\gamma) = \frac{F_{m_1,\eta_1}(U,T,V,X,\gamma) + F_{m_2,\eta_2}(U,T,V,X,\gamma)}{\min\limits_{i,j=1,\ldots,C; i \neq j} \left\| v_i - v_j \right\|^2} \tag{50}$$

The steps to implement hybrid algorithm between IT2SPFCM and PSO are as follows:

Algorithm 5: IT2SPFCM-PSO algorithm

Input: Satellite image data $X = \{x_k, x_k \in R^M, k = 1,\ldots,n\}$, labeled dataset $X^* = \{P_{is}, P_{is} \in R^M, s << n;$ the number of clusters $C(1 < C < n)$, i $= 1,\ldots,$C; $\varepsilon, T_{max}, t = 0, c_1, c_2, r_1, r_2, \omega$.

Step 1: Compute the additional cluster centroids $V^* = [v_i^*]$ by using Equation 29, the additional fuzzy MF $U^* = [\mu_{ik}^*]$ by using Equation 30, and the additional possibilistic MF $T^* = [\tau_{ik}^*]$ by using Equations 9 and 31.

Step 2: Initialization

2.1 Initialize the centroid matrix $V^{(0)} = [v_i^{(0)}], V^{(0)} \in R^{MxC}$ by using the FCM algorithm.

2.2 Initialize the location particles $P^{(0)} = (p_1^{(0)}, p_2^{(0)}, \ldots, p_{C*M}^{(0)}, p_{C*M+1}^{(0)}, \ldots, p_{C*M+8}^{(0)})$ by using $V^{(0)} = [v_i^{(0)}]$ and the random values $m, m_1, m_2, \eta, \eta_1, \eta_2, a, b$ within the limits of the search space.

2.3 Create the random velocity of particles: $vel_1^{(0)}, vel_2^{(0)}, \ldots, vel_{C*M}^{(0)}, vel_{C*M+1}^{(0)}, \ldots, vel_{C*M+8}^{(0)}$ within limits of the velocity.

2.4 Compute $U^{(0)}$ by using Equations 37, 38, 39, 26, 27, and 28.

2.5 Compute $T^{(0)}$ by using Equations 9, 40, 41, 42, and 43.

2.6 Compute $F_{m_1,\eta_1,m_2,\eta_2}^{(0)}$ by using Equation 50.

2.7 Let $pBest_i^{(0)} = p_i^{(0)}$, $gBest^{(0)}$ by using Equation 48.

Step 3: Hybrid algorithm of IT2SPFCM and PSO

3.1 $t = t + 1$

3.2 **For each** particle i

+ Compute the velocity of particles
$$vel_i^{(t+1)} = \omega * vel_i^{(t)} + c_1 * r_1 * (pBest_i^{(t)} - p_i^{(t)}) + c_2 * r_2 * (gBest^{(t)} - p_i^{(t)})$$

+ Compute the location of particles $p_i^{(t+1)} = p_i^{(t)} + vel_i^{(t+1)}$

+ Compute the objective function $F_{m_1,\eta_1,m_2,\eta_2}^{(t)}$ by using Equation 50.

+ Update $pBest_i^{(t)}$ by using Equation 47.

+ Update the cluster centroids $V^{(t)} = [v_i^{(t)}]$.

+ Update the parameters $m, m_1, m_2, \eta, \eta_1, \eta_2, a, b$ (if change).

3.3 Find the global best solution $gBest^{(t)}$ by using Equation 48.

3.4 Update $U^{(t)}$ by using Equations 37, 38, 39, 26, 27, and 28.

3.5 Update $T^{(t)}$ by using Equations 9, 40, 41, 42, and 43.

3.6 Check **if** $(t > T_{max})$ **then** go to Output **otherwise** return **Step 3.1**.

Output: $V^{(t)}, U^{(t)}, T^{(t)}, m, m_1, m_2, \eta, \eta_1, \eta_2, a, b$.

Defuzzification: Assign data x_k to the ith cluster if $u_{ik} \geq u_{jk}, j = 1,\ldots,C; j \neq i$.

Computational complexity: In *Step 1* the computational complexity by Equations 29, 30, 31 and 9 is $O(4nMC)$. In *Step 2*, the computational complexity of FCM algorithm is $O(nMC)$, and the computational complexity by Equations 37, 38, 39, 26, 27, 28, 9, 40, 41, 42, 43, 50 and 48 is $O(12nMC + MC)$. In *Step 3*, each iteration includes *step 3.2*, and Equation 50 causes a computational complexity of $O(nMC)(MC + 8)$, while that of *Step 3.3* is $O(CM + 8)$. For *step 3.4* and *step 3.5*, the computational complexity due to Equations 37, 38, 39, 26, 27, 28, 9, 40, 41, 42, and 43 is $O(11nMC)$. Therefore, in each loop in *step 3* there is a computational

complexity of $O(nM2C2 + 19nMC + CM + 8)$. When n is large and the number of iterations of is T_{max}, the computational complexity of IT2SPFCM-PSO algorithm is $O(nM2C2Tmax + 19nMCTmax + 17nMCT)$.

From the computational complexity analysis of the PFCM, IT2FCM, IT2SPFCM and IT2SPFCM-PSO algorithms, the IT2FCM and IT2SPFCM algorithms reveal similar computational complexities that are the largest compared to other algorithms. Therefore, when n is large, the computational complexities of the PFCM and IT2SPFCM-PSO algorithms are similar and smaller than the computational complexities of the IT2FCM and IT2SPFCM algorithms.

4. Experiment and Results

4.1. Parameter Initialization and Evaluation methods

We selected some datasets at different locations including cities, deltas and mountain forests for testing. Multispectral satellite images are used, including Landsat-5 TM, Landsat-7 ETM+, Landsat-8, and Sentinel-2A.

For a multispectral image with M bands, each pixel is characterized by M components on M gray bands, which described as follows: $X = [x_1, x_2, ...x_n]$ with $x_i = (b_{i1}, b_{i2}, ..., b_{iM})$.

Experimental algorithms include SFCM [7], PFCM [23], SPFCM-W [45], SPFCM-SS [46], MKSFCM [21], SIIT2FCM [39], IT2SPFCM, and IT2SPFCM-PSO. Additional information in semi-supervised algorithms is calculated from the labeled data. The algorithms are executed for a maximum of 1000 iterations, and $\varepsilon = 10^{-6}$. For all algorithms, we first ran FCM algorithm ($m = 2$) to determine the initial centroids. With the algorithms PFCM, IT2SPFCM and IT2SPFCM-PSO, $K = 1$ was selected to calculate the value γ_i by using Equation 9.

The parameters of the SFCM, SPFCM-W, SPFCM-SS, MKSFCM and SIIT2FCM algorithms were selected from the following papers [7, 21, 39, 45, 46]. The parameters of the PFCM algorithm were selected from [23]. The parameters of the IT2SPFCM algorithm were selected from [23, 36] specifically as follows: $m = \eta = 2$, $m_1 = \eta_1 = 1.5$, $m_2 = \eta_2 = 3.5$, and $a = b = \delta = 1$.

In each loop of the IT2SPFCM-PSO algorithm, PSO will search for the optimal parameters in their space. The individuals in the population include the centroids and parameters for the IT2SPFCM-PSO algorithm.

The set of centroids: $\underbrace{v_{11}, v_{12}, ...v_{1M}}_{V_1} \quad \underbrace{v_{21}, v_{22}, ...v_{2M}}_{V_2} \quad ... \quad \underbrace{v_{C1}, v_{C2}, ...v_{CM}}_{V_C}$

For 8-bit images, the numerical values of the pixels will range from 0 to 255, while those of 16-bit images range from 0 to 65536. Therefore, the search space for the set of centroids will be $(p_{min}; p_{max}) = (0; 255)$ or $(p_{min}; p_{max}) = (0; 65536)$.

The parameters for IT2SPFCM-PSO algorithm is $\underbrace{m, m_1, m_2, \eta, \eta_1, \eta_2, a, b}_{parameters}$. The remaining individuals representing the parameters of IT2SPFCM-PSO algorithm are limited to the range of 1 to 5, $(p_{min}; p_{max}) = (1; 5)$.

From the search space identified above, the paper also suggests a limit on the velocity of individuals that limits individuals from jumping out of the search space as $(vel_{min}, vel_{max}) = (-\frac{p_{max} - p_{min}}{2}; \frac{p_{max} - p_{min}}{2})$ for individuals who represent the centroids and $(vel_{min}, vel_{max}) = (-2.5; 2.5)$ for individuals who represent the parameters of IT2SPFCM-PSO algorithm.

The parameters $c_1 = c_1 = 2.05$ as suggested in the paper [19]. Initially, $\omega = 0.9$ is used and then decreased to 0.1 when the maximum number of loops (the generation number) is reached.

The classification results are compared with the statistical data of the Vietnam National Remote Sensing Center (VNRSC). The labeled data are taken directly from the satellite image data according to six landcover classes. In contrast, the clustering results are evaluated by several validity indexes including Bezdek partition coefficient index (PC-I) [48], Dunn separation index (D-I), the classification entropy index (CE-I) [49], the Xie-Beni index (XB-I) [47], the τ index ($\tau - I$), and the mean squared error index (MSE) [50], which are introduced to assess the degree of characteristic separation between pixels and cluster centroids. Large values for indexes PC-I and D-I are good for clustering results, while small values for indexes CE-I, XB-I, $\tau - I$ and MSE are good for clustering results. Semi-supervised algorithms use the distance D_{ik}; otherwise the distance is d_{ik}. However, to match the proposed method, we modify the way to calculate some indicators accordingly as follows:

- Partition Coefficient index:

$$PC = \frac{1}{n}\sum_{i=1}^{C}\sum_{k=1}^{n}\left(\mu_{ik}^2 + \tau_{ik}^2\right) \tag{51}$$

- Classification Entropy index:

$$CE = -\frac{1}{n}\sum_{i=1}^{C}\sum_{k=1}^{n}\left(\mu_{ik}\log\mu_{ik} + \tau_{ik}\log\tau_{ik}\right) \tag{52}$$

- Dunn index is defined as follows:

$$D = \min_{i=1,\ldots,C}\left\{\min_{j=1,\ldots,C;j\neq i}\left\{\delta(A_i,A_j)/\max_{t=1,\ldots,C}\{\Delta(A_t)\}\right\}\right\} \tag{53}$$

in which $\delta(A_i,A_j) = \min\left\{d(x_i,x_j)|x_i \in A_i, x_j \in A_j\right\}$ and $\Delta(A_t) = \max\left\{d(x_i,x_j)|x_i,x_j \in A_t\right\}$.

- The Xie and Beni index (XB) aims to quantify the ratio of the total variation within clusters and the separation of clusters:

$$XB = \frac{1}{n}\sum_{i=1}^{C}\sum_{k=1}^{n}\mu_{ik}^m D_{ik}^2 / \min_{i,j=1,\ldots,C;i\neq j}\left\|v_i - v_j\right\|^2 \tag{54}$$

- The index $\tau - I$ is defined as follows:

$$\tau = \frac{1}{n}\sum_{i=1}^{C}\sum_{k=1}^{n}\tau_{ik}^\eta D_{ik}^2 / \min_{i=1,\ldots,C;\forall x_k \notin v_i}\left\|v_i - x_k\right\|^2 \tag{55}$$

- The MSE index:

$$MSE(x,v) = \frac{1}{n}\sum_{i=1}^{c}\sum_{k=1}^{n}\left(x_{ik} - v_i\right)^2 \tag{56}$$

with $X = \{x_i\} = \{x_1, x_2, \ldots, x_n\}$ and $V = \{v_i\} = \{v_1, v_2, \ldots, v_c\}$ respectively the initial pixels and the centroid of the clusters.

- Furthermore, we also compared the correct classification rate of the number of labeled pixels according to the landcover. It is calculated by the following equations:

$$\begin{aligned} Percentage_{class_i} &= N_i^{true}/N_i \\ Percentage_{total} &= N^{true}/N \end{aligned} \tag{57}$$

where $Percentage_{class_i}, Percentage_{total}$ represent the correct classification rate according to the landcover and the correct classification rate for the entire area, respectively; N, N_i are the number of labeled pixels on the entire area and the ith cluster, respectively; N^{true}, N_i^{true} are the number of labeled pixels correctly classified on the entire area and on the ith cluster, $i = 1,2,\ldots,c$,respectively;

4.2. Landcover classification

We tested the landcover classification on three types of satellite images, including Landsat-7 ETM+, Landsat-8, and Sentinel-2A images.

Landsat-7 ETM+ image data are from the Hanoi capital central area in Vietnam ($105^0 26'47.1714"E$, $21^0 10'15.7519"N$, and $106^0 12'38.5927"E$, $20^0 52'33.2849"N$) acquired on September 30, 2009, with seven spectrum bands and a spatial resolution of $30m$ (Figure 2.a).

Landsat-8 image data are from the Quy Hop district area of Nghe An Province in Vietnam ($104^0 44'11.4162"E$, $19^0 29'1.3803"N$, and $105^0 34'33.0556"E$, $19^0 09'32.1210"N$) acquired in 2016. These data have eight spectrum bands and a spatial resolution of $30m$ (Figure 2.b).

Sentinel-2A image data are from the Vinh Phuc Province's northern area acquired on December 20, 2017, with four spectrum bands and a spatial resolution of $10m$ (Figure 2.c).
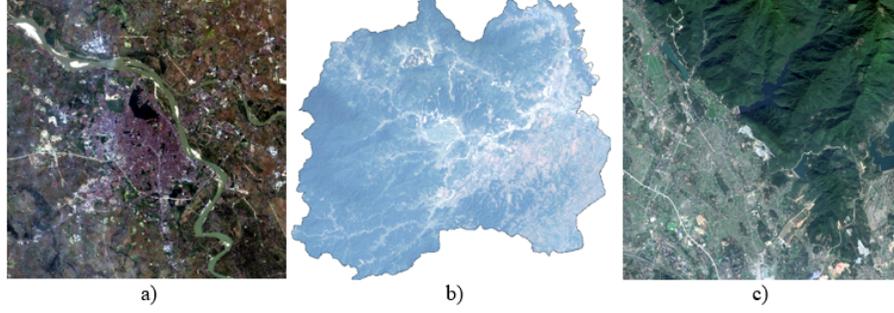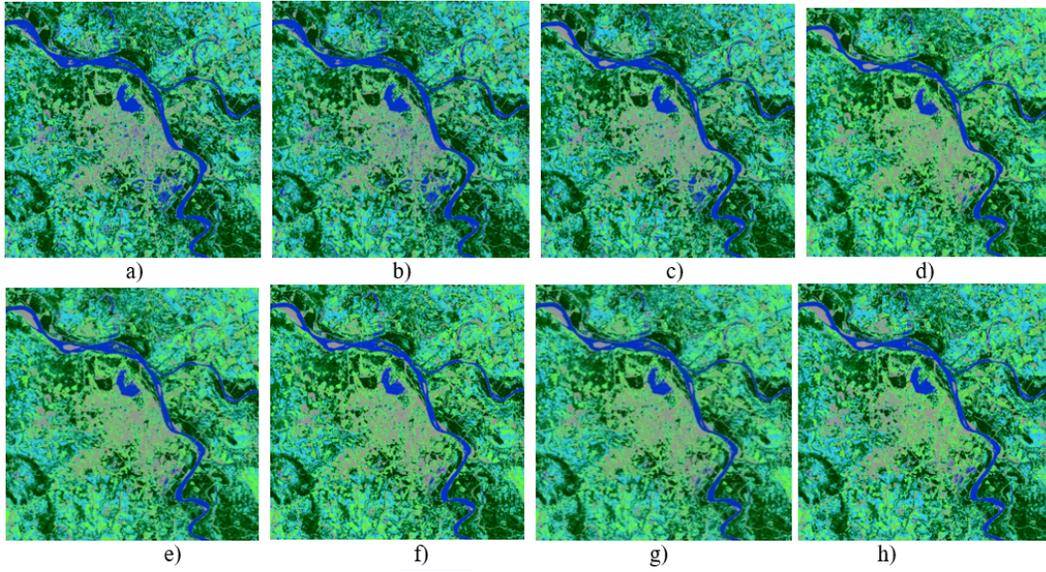
Figure 2: RGB color images of the three test areas

Legend: Class 1 (Rivers, ponds, and lakes ▬); Class 2 (Rocks, bare soil, and construction land ▬);
Class 3 (Fields and grass ▬); Class 4 (Planted forests and low woods ▬); Class 5 (Perennial forest
▬); Class 6 (Jungles forest ▬)

Figure 3: Landcover classification results of Hanoi central area: a) PFCM; b) SFCM; c) SPFCM-W; d) SPFCM-SS;
e) SMKFCM; f) SIIT2FCM; g) IT2SPFCM; h) IT2SPFCM-PSO

Table 1: Optimal parameters for the IT2SPFCM-PSO algorithm for the Hanoi central area

m	m_1	m_2	η	η_1	η_2	a	b
2.2136	1.3653	3.2651	2.1987	1.4763	3.0736	0.5275	0.5246

4.2.1. Experiment 1: Landsat-7 ETM+ image

In Table 1, the parameter values are achieved by the IT2SPFCM-PSO algorithm implementation. Figure 3 shows the classification results according to six landcovers by algorithms a) PFCM, b) SFCM, c) SPFCM-W, d) SPFCM-SS, e) SMKFCM, f) SIIT2FCM, g) IT2SPFCM, and h) IT2SPFCM-PSO.

Table 2 shows the correct classification rate (%) according to each landcover based on the number of labeled pixels. The IT2SPFCM-PSO algorithm achieves the best classification results, with the correct classification rate reaching over 99% for labeled pixels, while the lowest rate is 87.77% by the PFCM algorithm. This correct classification rate significantly increases when using the semi-supervised method. The IT2SPFCM algorithm has a correct classification rate of 98.21%, which is lower than that of the IT2SPFCM-PSO algorithm, while the computational complexity is higher than that of the IT2SPFCM-PSO algorithm. Furthermore, the

16

Table 2: Accuracy according to labeled data for the Hanoi central area(%)

Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Accuracy
SFCM	93.21	91.23	88.65	87.37	91.25	86.28	89.67
SMKFCM	98.45	97.18	93.23	95.55	95.99	92.65	95.51
SIIT2FCM	98.43	97.09	95.88	93.23	96.28	94.33	95.87
PFCM	89.72	88.23	91.46	87.38	84.61	85.19	87.77
SPFCM-W	95.34	93.76	93.22	91.26	92.65	91.89	93.02
SPFCM-SS	94.46	94.10	93.71	91.76	93.13	92.46	93.27
IT2SPFCM	99.11	**98.33**	98.37	97.42	**98.48**	97.56	98.21
IT2SPFCM-PSO	**100**	98.11	**99.15**	**99.08**	98.43	**99.32**	**99.02**

IT2SPFCM-PSO algorithm also provides the highest accuracy in 4 out of 6 landcovers.

Table 3 shows the area statistics data from VNRSC by six landcovers and the difference when classified by the algorithms SFCM, SMKFCM, SIIT2FCM, PFCM, SPFCM-W, SPFCM-SS, IT2SPFCM, and IT2SPFCM-PSO. The average difference in area is shown in the last column of Table 3.

It shows that the IT2SPFCM-PSO algorithm achieves the classification results with the lowest deviation of 0.114% compared to VNRSC statistics data; in other words, the classification accuracy by IT2SPFCM-PSO is 99.886%, while the deviation is 1.107% with the IT2SPFCM algorithm and 0.831% with the SIIT2FCM algorithm. The largest value comes from the PFCM algorithm, 2.196%, which is an unsupervised algorithm among the algorithms used for testing. The IT2SPFCM-PSO algorithm also achieves the highest accuracy in 5 out of 6 landcovers.

Table 3: The difference in landcover area compared to data from VNRSC (%)

Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Difference
VNRSC	4.646	13.211	18.512	26.552	23.308	13.772	0.000
SFCM	1.422	0.889	4.611	0.184	0.305	2.420	1.639
SMKFCM	0.790	0.395	1.166	2.134	0.277	2.391	1.192
SIIT2FCM	0.362	0.362	0.030	2.115	0.908	1.207	0.831
PFCM	2.872	0.561	4.592	0.165	1.435	3.550	2.196
SPFCM-W	0.158	**0.052**	2.280	1.902	1.875	2.097	1.394
SPFCM-SS	0.280	0.176	1.212	2.927	0.774	2.911	1.380
IT2SPFCM	0.180	0.220	2.070	2.071	1.090	1.010	1.107
IT2SPFCM-PSO	**0.121**	0.122	**0.010**	**0.020**	**0.197**	**0.216**	**0.114**

The experimental results also show that the tested algorithms have an accuracy of over 95% in the landcovers. Moreover, when using interval type-2 fuzzy sets and the PSO technique, the accuracy can increase to over 98%. The hybrid algorithms with PSO can help find the appropriate parameters, resulting in a significant increase in efficiency compared to the same algorithm without using the PSO technique.
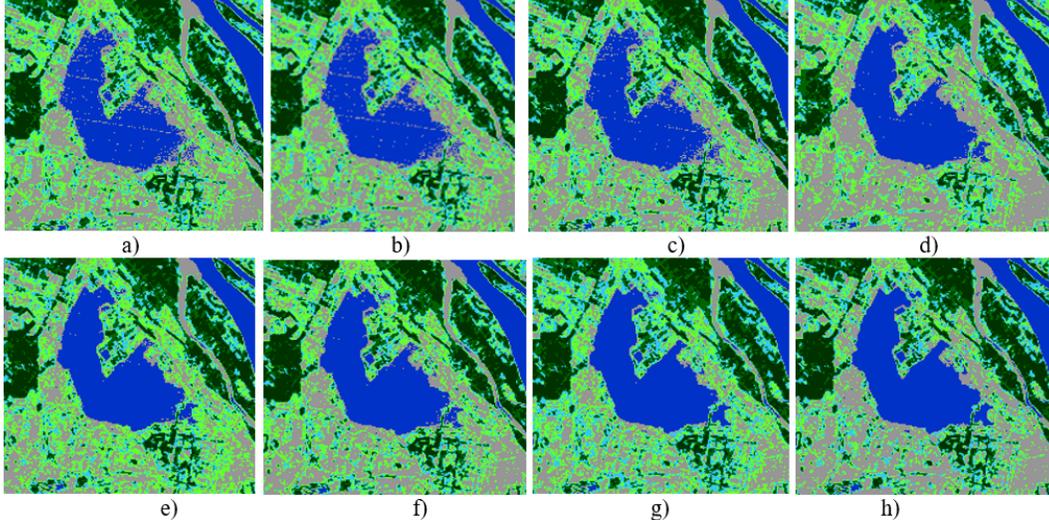
Table 4: The various validity indexes for the Hanoi central area

Algorithm	PC-I	CE-I	D-I	XB-I	$\tau - I$	MSE
SFCM	0.5821	0.5973	0.2654	2.1734	NaN	17.3674
SMKFCM	0.8776	0.4098	0.4798	0.6912	NaN	14.7862
SIIT2FCM	0.8893	0.2317	0.4938	0.1986	NaN	14.1893
PFCM	0.5765	0.5623	0.2589	1.9852	0.1762	18.8762
SPFCM-W	0.6873	0.4742	0.2985	1.0986	0.1438	16.9823
SPFCM-SS	0.7084	0.3587	0.3492	0.8764	0.1499	13.8751
IT2SPFCM	0.9165	0.2985	**0.5981**	0.1783	0.1078	10.1621
IT2SPFCM-PSO	**0.9243**	**0.2279**	0.5979	**0.1697**	**0.0921**	**9.8276**

In Table 4, the IT2SPFCM-PSO algorithm achieves the best results in 5 out of 6 indicators, followed

by the IT2SPFCM algorithm. However, in a few cases, the PFCM algorithm achieves better results than some semi-supervised algorithms. Specifically, the CE-I index obtained by the PFCM algorithm is 0.5623, compared to the 0.5973 achieved by the SFCM algorithms. The XB-I index obtained by the PFCM algorithm is 1.9852, compared to the 2.1734 achieved by the SFCM algorithm. It can be seen that by using unsupervised algorithms with initialization of suitable parameters, good clustering results can be achieved.

Legend: Class 1 (Rivers, ponds, lakes ▮); Class 2 (Rocks, bare soil, construction land ▮); Class 3 (Fields, grass ▮); Class 4 (Planted forests, low woods ▮); Class 5 (Perennial forests ▮); Class 6 (Jungle forests ▮)

Figure 4: Landcover classification results of West Lake area (Hanoi): a) PFCM; b) SFCM; c) SPFCM-W; d) SPFCM-SS; e) SMKFCM; f) SIIT2FCM; g) IT2SPFCM; and h) IT2SPFCM-PSO

Figure 4 shows the landcover classification results for the West Lake area in the center of Hanoi. In Figure 4 (a, b, and c), it can be seen that there is still much noise on the resulting image. This noise is visible in the West Lake area. In Figure 4.d, the noise has been significantly reduced, and there is little noise in Figure 4 (e and f). Especially in Figure 4 (g and h), it is challenging to detect noise and outliers with the naked eye.

In this experiment, the results of landcover classification by PFCM, SFCM, SPFCM-W, and SPFCM-SS algorithms yield the worst results (Figure 4 (a, b, c, and d). The algorithms SMKFCM, SIIT2FCM, IT2SPFCM, IT2SPFCM-PSO obtain significantly better results than the algorithms PFCM, SFCM, SPFCM-W, SPFCM-SS. It is challenging to identify noise and outliers on the resulting image, especially in the image for the IT2SPFCM-PSO algorithm.
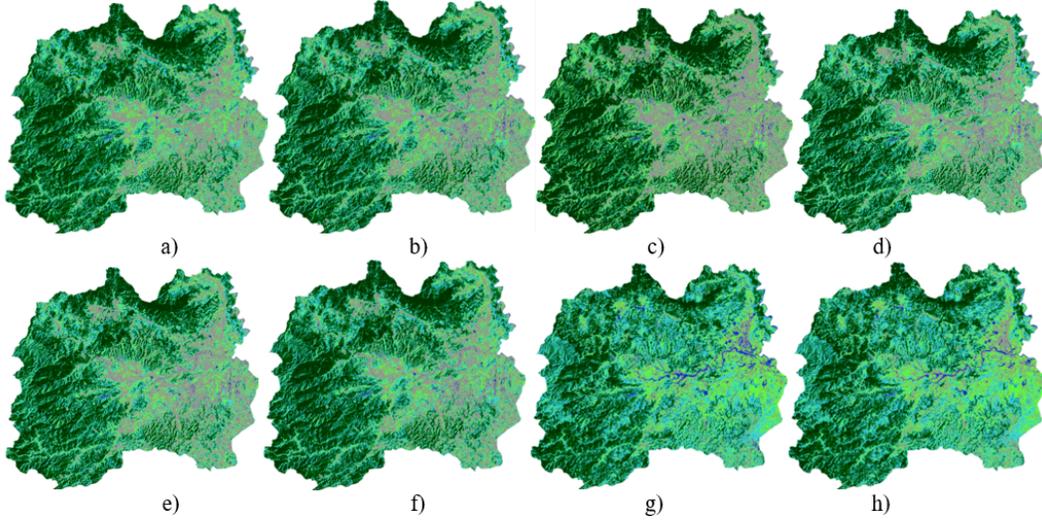
4.2.2. Experiment 2: Landsat-8 image

Table 5: Optimal parameters for the IT2SPFCM-PSO algorithm for the Quy Hop area

m	m_1	m_2	η	η_1	η_2	a	b
2.2876	1.4764	3.4565	2.1876	1.3768	3.3764	0.3764	0.3798

Table 5 shows the parameters achieved by the IT2SPFCM-PSO algorithm. Table 6 shows the correct classification rate according to the labeled pixels. The IT2SPFCM-PSO algorithm achieves the best classification results with the accuracy reaching 98.87%, while the lowest accuracy is 88.16% by the SFCM algorithm. The IT2SPFCM algorithm has an accuracy of 97.67%, while the SIIT2FCM algorithm has an accuracy of 95.52%. If considering each landcover, the IT2SPFCM-PSO algorithm achieves accuracies from 97.64% to 99.84% and is the highest on all six landcovers.

Table 7 shows the classification results according in landcover area (%) and VNRSC data. In this experiment, the IT2SPFCM-PSO algorithm has the smallest average deviation of 0.586%; followed by the

Legend: Class 1 (Rivers, ponds, and lakes ▇▇▇); Class 2 (Rocks, bare soil, and construction land ▇▇▇);
Class 3 (Fields and grass ▇▇▇); Class 4 (Planted forests and low woods ▇▇▇); Class 5 (Perennial forests
▇▇▇); Class 6 (Jungle forests ▇▇▇)

Figure 5: Landcover classification results of Quy Hop area: a) PFCM; b) SFCM; c) SPFCM-W; d) SPFCM-SS; e) SMKFCM; f) SIIT2FCM; g) IT2SPFCM; h) IT2SPFCM-PSO

Table 6: Accuracy according to labeled data for the Quy Hop area (%)

Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Accuracy
SFCM	92.14	89.49	86.32	89.34	85.38	86.28	88.16
SMKFCM	98.54	96.82	94.22	92.88	94.56	93.65	95.11
SIIT2FCM	98.28	95.89	96.46	94.82	96.32	91.33	95.52
PFCM	90.21	89.57	84.98	97.45	82.19	86.49	88.48
SPFCM-W	92.48	92.34	91.77	90.33	89.99	86.76	90.61
SPFCM-SS	93.89	92.11	89.46	90.89	90.45	88.69	90.92
IT2SPFCM	99.23	97.39	95.87	96.65	**98.79**	98.07	97.67
IT2SPFCM-PSO	**99.84**	**99.12**	**98.87**	**97.64**	**98.79**	**98.93**	**98.87**

IT2SPFCM and SIIT2FCM algorithms with 0.879% and 0.997%, respectively. The two algorithms SFCM and PFCM have average differences of over 2.3%. Although the IT2SPFCM-PSO algorithm achieved the highest accuracy in only two out of six landcovers, the overall accuracy was the highest at 99.414%.

Table 7: The difference in landcover area compared to data from VNRSC (%)

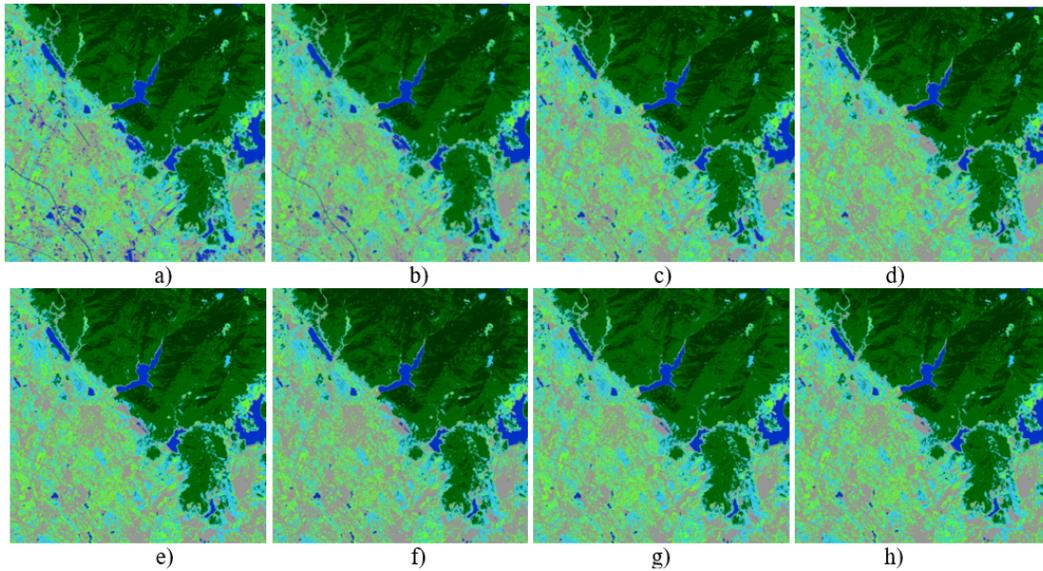| Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Difference
VNRSC	0.281	18.452	14.613	20.845	24.239	21.57	0
SFCM	0.374	1.828	3.090	4.338	2.583	1.628	2.307
SMKFCM	0.241	2.59	0.448	1.541	0.783	0.955	1.093
SIIT2FCM	0.184	2.67	**0.122**	**0.475**	**0.014**	2.515	0.997
PFCM	0.336	3.774	2.136	6.246	1.915	1.915	2.720
SPFCM-W	0.279	1.828	1.182	3.288	0.961	0.960	1.416
SPFCM-SS	0.269	1.552	2.479	2.43	1.250	**0.621**	1.434
IT2SPFCM	0.057	2.281	0.247	0.694	0.109	1.886	0.879
IT2SPFCM-PSO	**0.009**	**1.018**	0.695	0.768	0.034	0.989	**0.586**

Table 8 shows some indicators obtained from experimentation; as seen, the IT2SPFCM-PSO algorithm

achieves the best clustering results in all indicators, followed by the algorithms SIIT2FCM and IT2SPFCM. The two algorithms SFCM and PFCM obtain the poorest clustering quality compared to other algorithms.

Table 8: The various validity indexes for the Quy Hop area

Algorithm	PC-I	CE-I	D-I	XB-I	$\tau - I$	MSE
SFCM	0.4986	0.8753	0.3652	1.7824	NaN	12.9876
SMKFCM	0.6173	0.7863	0.4762	0.3278	NaN	9.5737
SIIT2FCM	**0.8763**	0.7382	0.4918	0.3148	NaN	8.5642
PFCM	0.4652	0.8864	0.3287	1.8712	0.1387	13.8743
SPFCM-W	0.5437	0.8593	0.3968	1.2874	0.1322	11.7842
SPFCM-SS	0.5519	0.8468	0.3981	0.1789	0.1299	10.9682
IT2SPFCM	0.8651	0.6529	0.5289	0.1328	0.0899	8.5601
IT2SPFCM-PSO	**0.8763**	**0.6281**	**0.5473**	**0.1147**	**0.0668**	**7.3654**

4.2.3. Experiment 3: Sentinel-2A image

Legend: Class 1 (Rivers, ponds, and lakes [blue]); Class 2 (Rocks, bare soil, and construction land [gray]); Class 3 (Fields and grass [light green]); Class 4 (Planted forests and low woods [cyan]); Class 5 (Perennial forests [green]); Class 6 (Jungle forests [dark green])

Figure 6: Landcover classification results for the Vinh Phuc area: a) PFCM; b) SFCM; c) SPFCM-W; d) SPFCM-SS; e) SMKFCM; f) SIIT2FCM; g) IT2SPFCM; h) IT2SPFCM-PSO

Table 9: Optimal parameters by the IT2SPFCM-PSO algorithm for the Vinh Phuc area

m	m_1	m_2	η	η_1	η_2	a	b
2.2653	1.4762	3.0984	2.1987	1.6872	2.9875	0.7653	0.7759

Table 9 shows the parameters that were achieved when the IT2SPFCM-PSO algorithm was completed. Table 10 shows the correct classification rate for each landcover based on labeled data. The IT2SPFCM-PSO algorithm achieves the best classification results, with the correct classification rate of the labeled pixels reaching 99.39%, while the lowest rate is 90.56% by the PFCM algorithm. Moreover, the algorithms SMKFCM, SIIT2FCM, IT2SPFCM, and IT2SPFCM-PSO achieve accuracies over 95%.

20

Table 10: Accuracy according to labeled data for the Vinh Phuc area (%)

Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Accuracy
SFCM	94.65	92.42	89.31	88.08	90.27	89.46	90.70
SMKFCM	98.39	95.37	96.37	93.44	92.54	94.36	95.08
SIIT2FCM	98.22	97.28	95.99	96.43	94.77	93.10	95.97
PFCM	92.19	91.21	90.43	88.88	90.90	89.73	90.56
SPFCM-W	96.65	94.22	90.56	91.95	89.81	92.95	92.69
SPFCM-SS	96.73	95.29	93.64	89.81	90.18	91.87	92.92
IT2SPFCM	99.28	99.34	98.26	98.49	98.21	**99.43**	98.84
IT2SPFCM-PSO	**100.00**	**99.67**	**99.11**	**99.24**	**99.55**	98.78	**99.39**

Table 11: The difference in landcover area compared to data from VNRSC (%)

Algorithm	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Difference
VNRSC	1.504	22.319	18.223	16.444	30.842	10.667	0.000
SFCM	0.314	2.241	1.351	1.244	1.426	1.466	1.340
SMKFCM	0.081	1.913	**0.204**	1.288	0.708	0.510	0.784
SIIT2FCM	0.101	0.644	0.640	0.777	1.375	0.970	0.751
PFCM	0.304	2.251	1.341	1.213	2.235	2.235	1.597
SPFCM-W	0.203	2.223	1.191	1.233	1.526	1.526	1.317
SPFCM-SS	0.303	2.661	0.264	1.283	1.956	1.146	1.269
IT2SPFCM	0.041	0.040	0.452	**0.451**	0.405	0.506	0.316
IT2SPFCM-PSO	**0.070**	**0.010**	0.755	0.752	**0.101**	**0.101**	**0.299**

Table 11 shows the difference in landcover area percentages (%) compared to VNRSC data. The classification results by the IT2SPFCM-PSO algorithm have the highest accuracy, with an average difference of only 0.299%, followed by the IT2SPFCM algorithm, with 0.316%. The two algorithms SMKFCM and SIIT2FCM have average differences of 0.784% and 0.751%, respectively. The remaining algorithms all have an average difference of over 1.2%, and the largest is 1.579% with the PFCM algorithm.

Table 12: The various validity indexes for Vinh Phuc area

Algorithm	PC-I	CE-I	D-I	XB-I	$\tau - I$	MSE
SFCM	0.6986	0.3167	0.3269	0.4389	NaN	7.7836
SMKFCM	0.8276	0.1388	0.5781	0.2517	NaN	6.8937
SIIT2FCM	0.8457	0.1287	0.5987	0.2074	NaN	5.4872
PFCM	0.4985	0.4278	0.3963	0.5286	0.1897	7.6812
SPFCM-W	0.7659	0.1687	0.2098	0.2587	0.1689	7.3519
SPFCM-SS	0.7981	0.1631	0.2911	0.2684	0.1573	7.1673
IT2SPFCM	0.8635	0.0938	**0.8524**	0.1983	0.0653	5.0982
IT2SPFCM-PSO	**0.8902**	**0.0842**	**0.8524**	**0.1729**	**0.0376**	**4.8278**

Table 12 shows the values of the indicators PC-I, CE-I, D-I, XB-I, MSE, and $\tau - I$. The proposed algorithm has better results than previous algorithms in all indexes. Moreover, when considering the computational complexity, the IT2SPFCM-PSO algorithm is lower than the IT2SPFCM algorithm.

Tables 1, 5, and 9 show that the parameter values are different from the above experiment. The Sentinel-2A image experiment has the highest accuracy, which is due to the Sentinel-2A image having a higher resolution ($10m$) compared to the images Landsat-7 ETM+ and Landsat-8 ($30m$).

Figure 7 shows the change in the objective function's value according to the number of loops. Specifically, Figure 7 (a, b, and c) shows the value of the F function for the central Hanoi capital area, the Quy Hop district, and the Vinh Phuc area, respectively. The process of finding the optimal parameter is also the process of minimizing the F objective function. It can be seen that the F function value decreases very quickly in
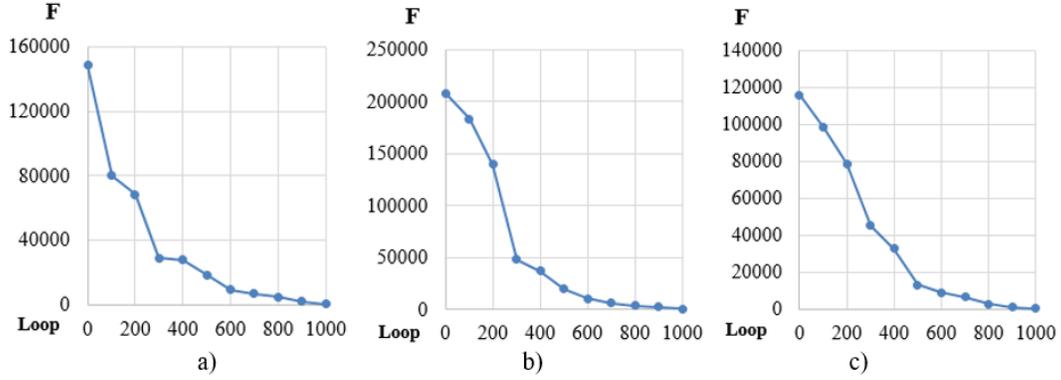
Figure 7: Objective function value of the IT2SPFCM-PSO algorithm according to the number of iterations

the first 300 iterations. Then the speed decreases and reaches the smallest value when the maximum number of iterations is reached.

In general, through the above three experiments, we can see that the semi-supervised algorithms based on the interval type-2 fuzzy sets (SIIT2FCM, T2SPFCM, IT2SPFCM-PSO) have higher accuracy than the other algorithms. From the above experiments, the proposed method has reduced computational complexity and achieves higher accuracy than the remaining algorithms. The results also show that the choice of parameters and the use of labeled data can significantly improve the classification algorithm's efficiency.

4.3. Landcover change detection

In this section, satellite image data are used to assess the landcover change including Landsat-5 TM, Landsat-7 ETM+ and Landsat-8. The test area is the Bac Binh district in Binh Thuan Province, Vietnam, from 1988 to 2017 ($107^0 39' 22.5529"E$, $11^0 33' 10.9214"N$, and $109^0 03' 54.7794"E$, $10^0 58' 25.7780"N$). This is one of the two districts most severely affected by drought and desertification in the South Central Coast region. Figure 8 shows the RGB color image of the Bac Binh district at six different times and the detailed information about satellite image data by year is described in Table 13. In this paper, we use six bands with a spatial resolution of $30m$ in the bands of each image.
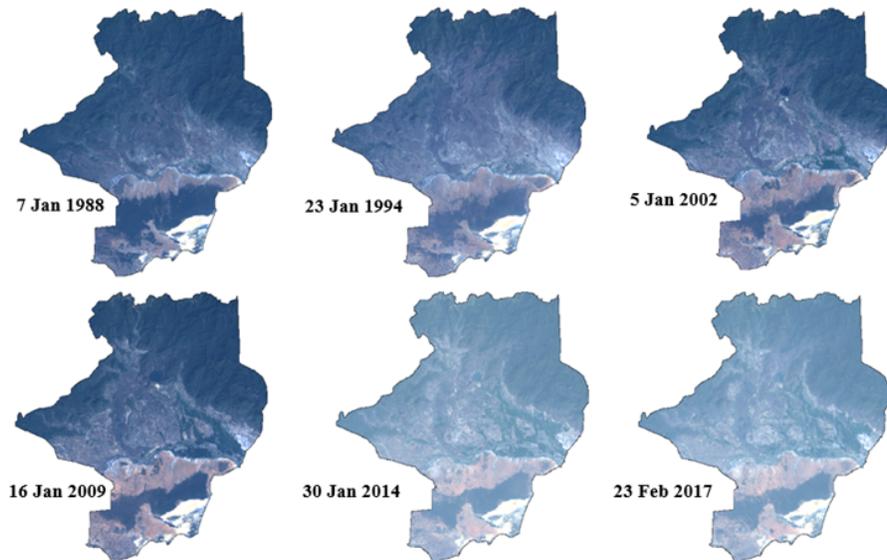

Figure 8: RGB color images: Bac Binh district, Binh Thuan Province, Vietnam

22

Table 13: Satellite image data of Bac Binh district, Binh Thuan Province, Vietnam

No.	Image type	Year	Number of bands	Number of bands used	Number of bits per pixel
1	Landsat-5 TM	07 Jan 1988	7	6	8 bits
2	Landsat-5 TM	23 Jan 1994	7	6	8 bits
3	Landsat-7 ETM+	05 Jan 2002	8	6	8 bits
4	Landsat-7 ETM+	16 Jan 2009	8	6	8 bits
5	Landsat-8	30 Jan 2014	11	6	16 bits
6	Landsat-8	23 Feb 2017	11	6	16 bits

The landcover classification using the IT2SPFCM-PSO algorithm results in six classes by percentage (%), as shown in Table 14 and Figure 9. In general, while most landcover areas are reduced or unchanged, the area of the "soil, rock, and construction land" class (Class 2) has a substantial increase from 1988 to 2017.

First, the "rivers, ponds, and lakes" class, occupying the smallest area, has increased from 0.06% to 0.325%; this is due to the lake's appearance near the district center. The "soil, rock, and construction land" class area saw the most robust increase, from 23.943% in 1988 to 37.518% in 2017. The process of urbanization is almost continuous over the years and is extended to the north of the Bac Binh district.

Second, the "field and grass" class in the area has hardly changed, although there is a change in distribution, which happens because people here still live mainly on agriculture. Similarly, the "jungle forest" class area also changed very little, from 16.444% in 1988 to 15.616%, in 2017. The area of jungle forests in the north is well preserved.

Table 14: Results of landcover classification by percentage using IT2SPFCM-PSO

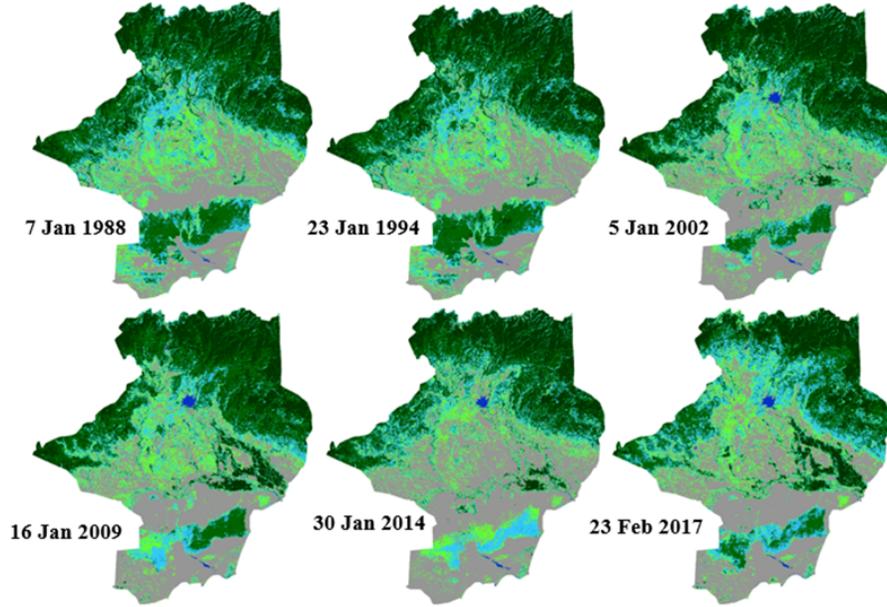| Class/ Year	1988(%)	1994(%)	2002(%)	2009(%)	2014(%)	2017(%)
Class 1	0.060	0.090	0.336	0.428	0.383	0.325
Class 2	23.943	24.697	28.531	28.227	32.19	37.518
Class 3	14.000	13.217	14.348	14.024	16.003	14.106
Class 4	17.150	15.422	14.236	13.729	16.252	14.806
Class 5	28.402	26.981	21.605	24.406	21.926	17.628
Class 6	16.444	19.593	20.944	19.186	13.806	15.616

Third, two classes of "planted forests and low woods" and "perennial forests" were all significantly reduced in the area. Mainly, the "perennial forest" class decreased significantly from 28.402% in 1988 to 17.628% in 2017, while the "planted forests and low woods" class decreased from 17.150% in 1988 to 14.806% in 2017. The areas of the "planted forests, low woods" and "perennial forest" classes decreased due to urbanization and the exploitation of forest products by humans.

Figure 10 shows the classification results according to six landcovers by year. A significant change can be seen in the landcover distribution. The difference in the landcover area here is mainly the loss of forest cover to give way to agricultural land, residential land, and desertification from 1988 to 2017, which can be seen in the "rock, bare soil and construction land" class.

An analysis of the results showed that the general trend in landcover changes in the Bac Binh district (Binh Thuan Province) was an increase in the area of "rocks, bare soil, and construction land" with a corresponding decline in forest area, including "planted forest", "perennial forests", and "jungle forests". Cover types such as surface water, fields, and grass did not change significantly during the period $1988 - 2017$.

The area of the "rocks, bare soil, and construction land" class has the most significant fluctuation, from 23.943% in 1988 to 37.518% in 2017, equivalent to 0.468% of the area of the district per year. The growth rate in the period from 2009 to 2017 (1.161% per year) is much higher than that in the period $1988 - 2009$ (0.204% per year). The increase in the area of "rocks, bare soil, and construction land" in the Bac Binh district can be explained by the expansion of residential areas as well as the impact of desertification here.

The sharpest decline was recorded by the "perennial forests" class, with a decline rate of approximately 0.513% per year. The rate of decline of the "perennial forests" was also low during the period $1988 - 2009$ (0.190% per year) and increased from 2009 to 2017 (0.753% of the total area of the district per year). From

Legend: Class 1 (Rivers, ponds, and lakes ■); Class 2 (Rocks, bare soil, and construction land ■);
Class 3 (Fields and grass ■); Class 4 (Planted forests and low woods ■); Class 5 (Perennial forests
■); Class 6 (Jungle forests ■)

Figure 9: Classification results: Bac Binh district, Binh Thuan Province, Vietnam

the classification results, it is possible to show the landcover change over the years. This result can help state management agencies in urban planning, forest resource management, socioeconomic development, etc.

Figure 10: Diagram of the landcover change by year from 1988 to 2017

The classification results on satellite images (Landsat-5 TM, Landsat-7 ETM+, Landsat-8, and Sentinel-2A) show that the proposed method can achieve higher accuracy than other algorithms. According to the classification results, when using some indicators to assess cluster quality, IT2SPFCM-PSO algorithm attains the best results in most cases. The accuracy of the classification results by the IT2SPFCM-PSO algorithm is above 95% for all experiments. The above results can be explained as follows:

- Using an interval type-2 fuzzy set can address uncertain data on satellite images better than using a

24

type-1 fuzzy set.

- The semi-supervised method used in the proposed algorithm can help improve accuracy compared to unsupervised algorithms.

- The PSO technique is used to optimize the centroid of clusters and fuzzy parameters. These values can help the proposed method improve stability and avoid falling into local minima. Experimental results indicate that the appropriateness of the parameters in clustering algorithms is very important.

Moreover, a higher resolution image yields higher accuracy on the same algorithm. The proposed method is also designed to be used for many different types of remote sensing image data, with many different resolutions.

The above-mentioned experiments have also shown that the proposed method can overcome some disadvantages and give higher accuracy in most cases than some other methods. The proposed method still has the following limitations:

- In principle, the proposed method can work with any dimensional image data, but it has not been applied to hyperspectral image data. Applications for hyperspectral image often require a massive number of calculations. This is feasible only when a parallel computing model or high-performance computing based on graphics processing units (GPUs) is established.

- The parameters of the algorithm, after being found, may not be useful on other data sets. This can occur because objects on the surface are continually changing in shape, size, and color. The same object may have different image data at different times.

5. Conclusions and Future work

The paper has proposed a hybrid algorithm between the semi-supervised PFCM and PSO based on IT2FS. The classification results on some satellite images (Landsat-5 TM, Landsat-7 ETM+, Landsat-8, and Sentinel-2A) show that the proposed method can achieve higher accuracy than some previous algorithms. According to the classification results, when using some indicators to assess cluster quality, the IT2SPFCM-PSO algorithm achieves the best results in most cases. The proposed method also achieves the highest accuracy when compared to labeled data, while the computational complexity of the IT2SPFCM-PSO algorithm is lower than that of the IT2SPFCM algorithm. Experiments also show that higher resolution image data yields higher accuracy on the same algorithm. Moreover, the semi-supervised method used in the proposed algorithm can help to improve accuracy, improve stability and avoid falling into local minima, while the PSO technique can determine the parameters of the proposed algorithm instead of utilizing random initialization. The accuracy of the classification results by the IT2SPFCM-PSO algorithm is above 95% for all experiments, which indicates that the appropriateness of the parameters in clustering algorithms is very important.

In the future, we wish to develop parallel models to accelerate computation and solve big data problems for hyperspectral image data. The development of models for dynamic parameter adaptation in optimization has also been suggested as one of the potential research directions to address the change in the shape and characteristics of objects on the ground surface.

References

[1] J.C. Bezdek, E.R William, Full. FCM: The fuzzy c-meams clustering algorithm. *Computers and Geosciences*, Vol.10 (2–3), pp. 191–203, 1984.

[2] X. Yang, F. Yu and W. Pedrycz, Typical Characteristics-based Type-2 Fuzzy C-Means Algorithm, *IEEE Transactions on Fuzzy Systems*, doi: 10.1109/TFUZZ.2020.2969907.

[3] Y. Shen, W. Pedrycz, Y. Chen, X. Wang and A. Gacek, Hyperplane Division in Fuzzy C-Means: Clustering Big Data, *IEEE Transactions on Fuzzy Systems*, doi: 10.1109/TFUZZ.2019.2947231.

[4] T.L. Hung, D.S. Mai. Classification of Remote Sensing Imagery Based on Density and Fuzzy c-Means Algorithm, *International Journal of Fuzzy System Applications*, Vol.8(2), pp. 1-15, 2019.

[5] Y. Zhong, A. Ma, and L. Zhang, An Adaptive Memetic Fuzzy Clustering Algorithm With Spatial Information for Remote Sensing Imagery, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Vol. 7(4), pp. 1235-1248, 2014.

[6] H. Zhang, Q. Wang, W. Shi, and M. Hao, A Novel Adaptive Fuzzy Local Information C-Means Clustering Algorithm for Remotely Sensed Imagery Classification, *IEEE Transactions on Geoscience and Remote Sensing*, Vol.55(9), pp. 5057 – 5068, 2017.

[7] D.S Mai and L.T Ngo, Semi-Supervised Fuzzy C-Means Clustering for Change Detection from Multispectral Satellite Image, *FUZZ-IEEE*, pp. 1-8, 2015.

[8] Z. Chen and B. Wang, Semi-supervised Spectral–Spatial Classification of Hyperspectral Imagery with Affinity Scoring, *IEEE Geoscience and Remote Sensing Letters*, Vol. 12(8), pp. 1710 – 1714, 2015.

[9] Y. Yuan, L. Haobo, X. Lu. Semi-supervised change detection method for multi-temporal hyperspectral images. *Neurocomputing*, Vol. 148, pp. 363-375, 2015.

[10] H. Li, S. Zhang, X. Ding, C. Zhang & R. Cropp. A novel unsupervised bee colony optimization (UBCO) method for remote-sensing image classification: a case study in a heterogeneous marsh area, *International Journal of Remote Sensing*, Vol. 37(24), 5726-5748, 2016.

[11] M. Zhang, J. Ma, M. Gong. Unsupervised Hyperspectral Band Selection by Fuzzy Clustering with Particle Swarm Optimization, *IEEE Geoscience and Remote Sensing Letters*, 14(5), pp. 773 – 777, 2017.

[12] J. Lilin, L. Weidong, S. Zheng, T. Shasha. Hybrid fuzzy clustering methods based on improved self-adaptive cellular genetic algorithm and optimal-selection-based fuzzy c-means, *Neurocomputing*, Vol. 249, pp. 140-156, 2017.

[13] H. Hagras, C. Wagner, Towards the Widespread Use of Type-2 Fuzzy Logic Systems in Real World Applications. *IEEE Computational Intelligence Magazine*, pp.14-24, 2012.

[14] J. Mendel, Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Second Edition, *Springer*, 2017.

[15] Melin, P., & Castillo, O. A review on the applications of type-2 fuzzy logic in classification and pattern recognition. *Expert Systems with Applications*, 40(13), pp. 5413–5423, 2013.

[16] J. Andreu-Perez, F.Cao, H. Hagras, G.Yang, A self-adaptive online brain–machine interface of a humanoid robot through a general type-2 fuzzy inference system, *IEEE Transactions on Fuzzy Systems*, Vol. 26(1), pp. 101-116, 2018.

[17] Yu, X., Zhou, W., & He, H. A method of remote sensing image auto classification based on interval type-2 fuzzy c-means. *FUZZ-IEEE*, pp. 223-228, 2014.

[18] D.S. Mai, L.T. Long, L.H. Trinh. Approach the Interval Type-2 Fuzzy System and PSO Technique in Landcover Classification. *ACIIDS, Lecture Notes in Computer Science*, Vol 12034. Springer, 2020.

[19] J. Kennedy, R. Eberhart. Particle Swarm Optimization. *IEEE International Conference on Neural Networks*, pp. 1942 – 1948, 1995.

[20] C. Wang, W. Pedrycz, J. Yang, M. Zhou and Z. Li, Wavelet Frame-Based Fuzzy C-Means Clustering for Segmenting Images on Graphs, *IEEE Transactions on Cybernetics*, Vol. 50(9), pp. 3938-3949, 2020.

[21] D.S Mai and L.T Ngo, Multiple Kernel Approach to Semi-Supervised Fuzzy Clustering Algorithm for Land-Cover Classification, *Engineering Applications of Artificial Intelligence*, Vol. 68, pp. 205–213, 2018.

[22] R. Krishnapuram and J. Keller, The possibilistic c-Means algorithm: Insights and recommendations, *IEEE Transactions on Fuzzy Systems*, Vol. 4(3), pp. 385–393, 1996.

[23] N. R. Pal, K. Pal, J. M. Keller, and J. C. Bezdek. A Possibilistic Fuzzy c-Means Clustering Algorithm, *IEEE Transactions on Fuzzy Systems*, Vol. 13(4), pp. 517-530, 2005.

[24] S. Askari, N. Montazerin, M.H. Fazel Zarandi and E. Hakimi, Generalized entropy based possibilistic fuzzy C-means for clustering noisy data and its convergence proof, *Neurocomputing*, Vol.219, pp. 186-202, 2017.

[25] M.S. Yang and Y. Nataliani, A feature-Reduction Fuzzy Clustering Algorithm Based on Feature-Weighted Entropy, *IEEE Transactions on Fuzzy Systems*, Vol.99, 2017.

[26] Y. Wu, Q. Miao, W. Ma, M. Gong, S. Wang. Particle Swarm Optimization Sample Consensus Algorithm for Remote Sensing Image Registration, *IEEE Geoscience and Remote Sensing Letters*, Vol.15(2), pp. 242 – 246, 2018.

[27] M. Maboudi, J. Amini, M. Hahn & M. Saati. Objectbased road extraction from satellite images using ant colony optimization, *International Journal of Remote Sensing*, Vol. 38(1), 179-198, 2016.

[28] X. Yin, T. Shu, Q. Huang, Semi-supervised fuzzy clustering with metric learning and entropy regularization, *Knowledge-Based Systems*, Vol.35, 304–311, 2012.

[29] H. Zhang, J. Lu. Semi-supervised fuzzy clustering: A kernel-based approach, *Knowledge-Based Systems*, Vol.22, 477–481, 2009.

[30] C. Lynch. H. Hagras, V. Callaghan, Embedded Interval Type-2 Neuro-Fuzzy Speed Controller for Marine Diesel Engines, *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2006)*, pp. 1340-1347, 2006.

[31] J. Mendel and R. John, Type-2 fuzzy sets made simple, *IEEE Transactions Fuzzy Systems*, Vol. 10, pp. 117–127, 2002.

[32] E. Yasunori, H. Yukihiro, Y. Makito, M. Sadaaki. On Semi-Supervised Fuzzy c-Means Clustering, *FUZZ-IEEE*, pp. 1119-1124, 2009.

[33] D.D. Nguyen, L.T. Ngo, L.T. Pham, W. Pedrycz, Towards Hybrid Clustering Approach to Data Classification: Multiple Kernel based-Interval-valued Fuzzy C-Means Algorithms, *Fuzzy Sets and Systems*, 2015.

[34] G. Ruiz-García, H. Hagras, H. Pomares and I. R. Ruiz, Toward a Fuzzy Logic System Based on General Forms of Interval Type-2 Fuzzy Sets, *IEEE Transactions on Fuzzy Systems*, vol. 27(12), pp. 2381-2395, 2019.

[35] H. B. Sola, J. Fernandez, H. Hagras, F. Herrera, M. Pagola and E. Barrenechea, Interval Type-2 Fuzzy Sets are Generalization of Interval-Valued Fuzzy Sets: Toward a Wider View on Their Relationship, *IEEE Transactions on Fuzzy Systems*, vol. 23(5), pp. 1876-1882, 2015.

[36] C. Hwang and F. C. H. Rhee, Uncertain Fuzzy clustering: Interval Type-2 Fuzzy Approach to C-Means, *IEEE Transactions on Fuzzy Systems*, Vol. 15(1), pp. 107-120, 2007.

[37] O. Linda and M. Manic, General Type-2 Fuzzy C-Means Algorithm for Uncertain Fuzzy Clustering, *IEEE Transactions on Fuzzy Systems*, Vol. 20(5), pp. 883-897, 2012.

[38] D.S Mai, L.T Ngo, Interval Type-2 Fuzzy C-Means Clustering with Spatial Information for Land-Cover Classification, *ACIIDS, Springer LNAI 9011*, pp.387-397, 2015.

[39] L.T. Ngo, D.S Mai, W. Pedrycz. Semi-supervising Interval Type-2 Fuzzy C-Means clustering with spatial information for multi-spectral satellite image classification and change detection, *Computers and Geosciences*, 83, pp. 1-16, 2015.

[40] Guo, J., & Huo, H. An Enhanced IT2FCM* Algorithm Integrating Spectral Indices and Spatial Information for Multi-Spectral Remote Sensing Image Clustering. *Remote Sensing*, 9(9), 960, 2017.

[41] D.D. Nguyen, L.T. Ngo, L.T. Pham, Multiple Kernel Interval Type-2 Fuzzy C-Means Clustering, *FUZZ-IEEE*, 2013.

[42] Ji, Z., Xia, Y., Sun, Q., Cao, G., Interval-valued possibilistic fuzzy C-means clustering algorithm. *Fuzzy Sets and Systems*, Vol.253, pp. 138-156, 2014.

[43] C. Wang, A. Xu and X. Li. Supervised Classification High-Resolution Remote-Sensing Image Based on Interval Type-2 Fuzzy Membership Function, *Remote Sensing*, 10, 710, 2018.

[44] H. Huo, J. Guo and Z.L. Li, Hyperspectral Image Classification for Land Cover Based on an Improved Interval Type-II Fuzzy C-Means Approach, *Sensors*, 18, 363, 2018.

[45] D.S Mai, L.T Ngo. Semi-supervised method with spatial weights based Possibilistic fuzzy C-means clustering for Land-cover Classification, *IEEE-NICS*, pp.406-411, 2018.

[46] D.S Mai, L.T Ngo, H.L Trinh. Advanced Semi-supervised Possibilistic Fuzzy C-means Clustering using Spatial-Spectral distance for Land-cover Classification, *IEEE Conference on SMC*, pp.4375-4380, 2018.

[47] U. Maulik, S. Bandyopadhyay, Performance Evaluation of Some Clustering Algorithms and Validity Indices, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(12), pp.1650-1654, 2002.

[48] J.C. Bezdek, N. Pal. Some new indexes of cluster validity. *IEEE Transactions on Systems, Man and Cybernetics*, 28(3), pp.301-315, 1998.

[49] C.H. Chou, M.C, Su, E. Lai. A new cluster validity measure and its application to image compression. *Pattern Analysis and Applications*, 7(2), pp 205-220, 2004.

[50] Wang Z. and Bovik A. C. Mean squared error: love it or leave it? A new look at signal fidelity measures. *IEEE signal processing magazine*, 98-117, 2009.