Engineering Applications of Artificial Intelligence 68 (2018) 205–213

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Multiple kernel approach to semi-supervised fuzzy clustering algorithm for land-cover classification

Sinh Dinh Mai, Long Thanh Ngo *

Department of Information Systems, Faculty of Information Technology, Le Quy Don Technical University, 236, Hoang Quoc Viet, Bac Tu Liem, Hanoi, Viet Nam

ARTICLE INFO

Keywords:
Semi-supervised clustering
Fuzzy clustering
Kernel fuzzy c-means
Multiple kernel
Land cover classification

ABSTRACT

Clustering is used to detect sound structures or patterns in a dataset in which objects positioned within the same cluster exhibit a substantial level of similarity. In numerous clustering problems, patterns is not easily separable due to the highly complex shaped data. In the previous studies, kernel-based methods have exhibited the effectiveness to partition such data. In this paper, we proposed a semi-supervised clustering method based fuzzy c-means algorithm using multiple kernel technique, called SMKFCM, in which the rudimentary centroids are directly used to the calculating process of centroids. The SMKFCM algorithm is on the basis of combining the labeled and unlabeled data together to improve performance. We used the labeled patterns to calculate the centrality of clusters considered as the rudimentary centroids which are added into the objective functions. The SMKFCM algorithm can be applied to both clustering and classification problems. The experimental results show that SMKFCM algorithm can improve significantly the classification accuracy which comes from comparison with a conventional classification or clustering algorithms such as semi-supervised kernel fuzzy c-means (S2KFCM), semi-supervised fuzzy c-means (SFCM) and Self-trained semi-supervised SVM algorithm (PS3VM).

1. Introduction

Recently, fuzzy clustering methods have been studied and widely used in various applications. One of the most popular fuzzy clustering methods is the Fuzzy C-Means (FCM) algorithm (Hoppner et al., 1999; Bezdek et al., 1984). In the FCM algorithm, a pattern may belong to more than one cluster with individual membership degrees. Although, the FCM clustering algorithm is limited to the discovery of spherical clusters. The limitation of the standard FCM algorithm is the usage of the Euclidian distance in the observation space on which the algorithm is only effective for spherical clusters, but it does not perform well for more general clusters (Shen et al., 2006). Besides, clustering results strongly depend on the characteristics of data-sets, the KFCM algorithm does not produce clusters with the desired accuracy in some datasets (Yu et al., 2011). The real-world clustering problems usually contains some useful features when combining with other ones. For example, image segmentation results could be produced better by combining pixel intensity feature and the local spatial information feature.

To overcome the drawbacks of the conventional FCM technique, kernel fuzzy c-means clustering (KFCM) algorithm was proposed Hathaway et al. (2005) to solve this problem by mapping input data into an appropriate space using a nonlinear function. This approach has received considerable attention because kernels make it possible to map data into a high-dimensional feature space in order to increase the representable capability of a linear clustering. Some proposed improvements KFCM algorithm by adapting a new kernel induced metric in the data space (Graves and Pedrycz, 2007) or transforming the original inputs into a much higher dimensional Hilbert space (Zhang and Chen, 2004, 2003b; Graves and Pedrycz, 2010).

On the basis of using kernel technique in clustering, some studied was done to improve the performance (Shawe-Taylor and Cristianini, 2004; Rakotomamonjy et al., 2007; Varma and Babu, 2009). Girolami (2002) proposed a kernel-based clustering for a wider variety of clusters and Tzortzis and Likas (2009) also introduce an algorithm based on kernel methods to overcome the cluster initialization problem. Later, Zhang and Chen (2003a) proposed the kernel-based fuzzy c-means (KFC) algorithm, which allows for incomplete data as well. Graves and Pedrycz (2010, 2007) launched a comprehensive comparative analysis of kernel-based fuzzy clustering and fuzzy clustering. The Fuzzy C-Means algorithm (FCM) algorithm and Gustafson–Kessel (GK) FCM was compared with two typical kernel-based fuzzy clustering algorithms: one with the prototypes located in the feature space (KFCM-F) and the other where the prototypes are distributed in the kernel space (KFCM-K). Both these kernel clustering algorithms are used the Gaussian kernel

* Corresponding author.
E-mail address: ngotlong@mta.edu.vn (L.T. Ngo).

https://doi.org/10.1016/j.engappai.2017.11.007
Received 1 November 2016; Received in revised form 21 September 2017; Accepted 12 November 2017

and KFCM-K deals with polynomial kernel. The analyzed datasets in this study show that the kernel-based FCM algorithms are better than the standard FCM and GKFCM.

Because of the advantages of clustering methods based on the kernel techniques, these algorithms have been applied in many different fields, particularly in image processing. Several studies and applications using KFCM in image segmentation include the kernelized fuzzy C-means (KFCM) algorithm (Zhang and Chen, 2004) which used a kernel-induced distance metric and a spatial penalty on the membership functions and a novel modified kernel fuzzy c-means(NMKFCM) (Yu et al., 2011) algorithm based on conventional KFCM which incorporates the neighbor term in its objective function.

Besides, Hathaway et al. (2005) presented a kernel expansion for clustering relational data by producing a kernelized algorithm of the non-Euclidean relational FCM. In addition, Chiang and Hao (2003) also proposed a multiple sphere support vector clustering algorithm based on the adaptive cell growing model which maps the input patterns to a high-dimensional feature space by using the desired kernel function.

Land-cover classification has widely applied to various fields (Torahi and Rai, 2011) and approached according to some methods of soft computing (Stavrakoudis et al., 2010; Stavrakoudis et al., 2011; Gordo et al., 2013). These studies are mainly based on methods of supervised and unsupervised classification (Genitha and Vani, 2013). Further, an approach using an ensemble of semisupervised classifiers were proposed for change detection in remotely sensed images (Roy et al., 2014) by using a multiple classifier system in semi-supervised (learning) framework instead of a single weak classifier. In the manner of semi-supervised change detection, Yuan et al. (2015) proposed a new distance metric learning framework for change detection by abundant spectral information of hyper-spectral image in noisy condition; Liu et al. (2013) proposed a novel semi-supervised SVM (PS3VM) model using selftraining approach to address the problem of remote sensing land cover classification.

Karvelis et al. (2000) presented a semi-unsupervised method for the M-FISH chromosome image classification. Firstly, the separation of foreground and background is performed by using an automated thresholding procedure. Then, these features are normalized. Secondly, the K-Means algorithm was applied to cluster the chromosome pixels into the 24 chromosome classes. Although this algorithm does not require a training, it produces on average, higher accuracy. However, the tests of the algorithm only used a small number of images. The usage of fuzzy C-means method with the labeled information (pre-determined clusters) Mai and Ngo (2015) introduced an approach which exploits local spatial information between the pixel and its neighbors to compute the membership degree. After that, Ngo et al. (2015) has developed algorithms for interval type-2 fuzzy clustering algorithm applying to the problems of satellite image analysis consisting of land cover classification and change detection.

Frigui et al. (2013) provided an overview of several fuzzy kernels clustering algorithms by using partial supervision information to guide the optimization process and avoid local minima. Zhang et al. presented a semi-supervised kernel-based FCM algorithm (Zhang et al., 2004) by introducing semi-supervised learning technique and the kernel method simultaneously into conventional fuzzy clustering algorithm in which the labeled and unlabeled data are used together. A semi-supervised kernel-based FCM algorithm with Pairwise Constraints was proposed by Wang et al. (2008), which incorporates both semi-supervised learning technique and the kernel method into fuzzy clustering algorithm.

Note that kernel methods depend on the usage of a suitable kernel function. If the kernel method only selects a single kernel from a predefined group then sometime it is not insufficient to represent all datasets. In addition, individual features of the selected input data can result in different clusters corresponding to individual kernels. Therefore, combining multiple kernels from a set of basis kernels has been proposed to refine better clusters rather than using single kernel method. The most important key in the kernel method is how to use the formulation of suitable kernel function (Zhao et al., 2009; Ganesh and Palanisamy 2012; Huang et al. 2012; Yugander et al., 2012).

Thus, kernel fuzzy clustering algorithms are necessary to be extended with the aggregation of kernel functions from different sources. The rudimentary information of centroids was also added to the objective function to adjust the centroids through the iterative computing process. The paper introduces new clustering method to show how to apply multiple kernel technique in semi-supervised clustering. There are two problems mentioned in this paper. Firstly, we extracted the characteristics from datasets to estimate the rudimentary centroid of clusters which is one of components of the objective function. Besides, the single kernel and multiple kernel techniques are used in the semi-supervised fuzzy clustering. Secondly, two the proposed algorithms are applied to classification on various datasets and satellite image datasets. Experiments are compared with previous algorithms like semi-supervised kernel fuzzy c-means (S2KFCM), semi-supervised fuzzy c-means (SFCM) and Self-trained semi-supervised SVM algorithm (PS3VM); and validity indices are analyzed in comparison with the survey data.

The remaining parts of this paper are organized as follows: Section 2 provides background on the kernel technique and kernel fuzzy c-mean, Section 3 presents two proposed algorithms i.e semi-supervised kernel or multiple kernel fuzzy C-Means clustering; Section 4 demonstrates experiments on satellite image classification; Section 5 is a conclusion and future works.

2. Prerequisites

2.1. The kernel technique

In machine learning, kernel methods are a class of algorithms for pattern analysis which is to find the general types of relations (for example clusters, rankings, principal components, correlations, classifications) in datasets. In several algorithms solving these tasks, the raw-presented datasets have to be explicitly transformed into the feature vector — presented ones via a user-specified feature map.

The key idea of kernel technique is to invert the series of arguments, i.e. choose a kernel k rather than a mapping before applying a learning algorithm. Note that it is not any symmetric function k is considered as a kernel. Suppose the input space X has a limited number of elements, i.e. $X = \{x_1, x_2, \dots, x_r\}$. Then, the $r \times r$ kernel matrix K with $K_{ij} = k(x_i, x_j)$ is definitely a symmetric matrix and $k(x_i, x_j)$ is a kernel value between x_i and x_j in X. The necessary and sufficient conditions of $k : \chi \times \chi \to \mathbb{R}$ are a kernel, which are given by Mercers theorem (Girolami, 2002).

Theorem 2.1. *The function $k : \chi \times \chi \to \mathbb{R}$ is a Mercer kernel if, and only if, for each $r \in \mathbb{N}$ and $x = (x_1, x_2 \dots, x_r) \in \chi^r$ the $r \times r$ matrix $K = \left(k\left(x_i, x_j\right)\right)_{i,j=1}^r$ is positive semi definite.*

There exist simple rules for designing kernels on the basis of given kernel functions.

Theorem 2.2. *Kernel functions. Let $k_1 : \chi \times \chi \to \mathbb{R}$ and $k_2 : \chi \times \chi \to \mathbb{R}$ be any two Mercer kernels. Then, the functions $k : \chi \times \chi \to \mathbb{R}$ given by*

1. $k(x, \tilde{x}) = k_1(x, \tilde{x}) + k_2(x, \tilde{x})$
2. $k(x, \tilde{x}) = c.k_1(x, \tilde{x}), \forall c \in \mathbb{R}^+$
3. $k(x, \tilde{x}) = k_1(x, \tilde{x}) + c, \forall c \in \mathbb{R}^+$
4. $k(x, \tilde{x}) = k_1(x, \tilde{x}) \cdot k_2(x, \tilde{x})$
5. $k(x, \tilde{x}) = f(x) \cdot f(\tilde{x}) \forall f : \chi \to \mathbb{R}$.

are also Mercer kernels.

Theorem 2.3. *Let $k_1 : \chi \times \chi \to \mathbb{R}$ be any Mercer kernel. Then, the functions $k : \chi \times \chi \to \mathbb{R}$ given by*

1. $k(x, \tilde{x}) = \left(k_1(x, \tilde{x}) + \theta_1\right)^{\theta_2} \forall \theta_1 \in \mathbb{R}^+, \forall \theta_2 \in \mathbb{N}$.
2. $k(x, \tilde{x}) = \exp\left(\frac{k_1(x, \tilde{x})}{\sigma^2}\right), \forall \sigma \in \mathbb{R}^+$.

3. $k(x,\tilde{x}) = \exp\left(-\frac{k_1(x,x)-2k_1(x,\tilde{x})+k_1(\tilde{x},\tilde{x})}{2\sigma^2}\right), \forall\sigma \in \mathbb{R}^+$.
4. $k(x,\tilde{x}) = \frac{k_1(x,\tilde{x})}{\sqrt{k_1(x,x)\cdot k_1(\tilde{x},\tilde{x})}}$.

are also Mercer kernels.

2.2. Kernel fuzzy c-means clustering

As an enhancement of conventional FCM, the Kernel Fuzzy C-mean Clustering (KFCM-F) uses a nonlinear map defined as follows:

$\phi : x \to \phi(x) \in H, x \in X \in R^d$

where X denotes the dataset and H is a Hilbert space (called kernel space).

Generally, the transform function ϕ is not given out explicitly, but the kernel function is given and defined as $k : \theta x\theta \to R$

$$k(x,y) = \phi(x)\phi(y)^T \ \forall x, y \in R. \tag{1}$$

Here $\phi(x)\phi(y)^T$ is an inner product of the kernel function. Such kernel functions are usually called Mercer kernels or kernel. Given a Mercer kernel k, we know that there is always a transform function $\phi : x \to \phi(x) \in H, x \in X \in R^d$ satisfies $k(x,y) = \phi(x)\phi(y)^T$, although we do not know the specific form of ϕ.

The widely used Mercer kernels consist of the Gaussian kernel $k(x,y) = \exp\left(-\|x-y\|^2/r^2\right)$ and the polynomial kernel $k(x,y) = \left(x^T.y+d\right)^2$.

Similar to FCM, fuzzifier m is used (m is usually 2) to handle the uncertainty, it minimized the following objective function as

$$J_m(U,V) = \sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}{}^m \|\phi(x_j) - \phi(v_i)\|^2 \tag{2}$$

where c is the number of clusters, n is the number of patterns and $\|\phi(x_j) - \phi(v_i)\|$ is the Euclidean distance between the pattern x_j and the prototype v_i in the kernel space.

By using the Euclidean distance $\|\phi(x_k)-\phi(v_i)\|$, the squared distance is computed in the kernel space using a kernel function

$$\left\|\phi(x_j) - \phi(v_i)\right\|^2 = k(x_j,x_j) + k(v_i,v_i) - 2k(x_j,v_i). \tag{3}$$

The degree of membership u_{ij} is determined as follows:

$$u_{ij} = \frac{1}{\sum_{l=1}^{c}\left(\frac{d_{\phi ij}}{d_{\phi lj}}\right)^{2/(m-1)}} \tag{4}$$

in which $i = 1,\dots,c$, $j = 1,\dots,n$, $d_{\phi ij} = \left\|\phi(x_j) - \phi(v_i)\right\|$.

If we use the Gaussian kernel then $k(x,x) = 1$ and $\left\|\phi(x_j) - \phi(v_i)\right\|^2 = 2(1 - k(x_j,v_i))$. The derivation of prototypes depends on the specific selection of the kernel function. The calculation of prototypes v_i, $i = 1,\dots,c$, with the Gaussian kernel is reached as follows

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^m k(x_j,v_i)\, x_j}{\sum_{j=1}^{N} u_{ij}^m k(x_j,v_i)}, \tag{5}$$

Next, defuzzification for KFCM-F is made as if $u_i(x_k) > u_j(x_k)$ for $j = 1,\dots,c$ and $i \neq j$ then x_k is assigned to cluster i.

The literal KFCM-F algorithm has a memory requirement of $O(nCMT)$,where n is the number objects in the dataset, M is the dimensionality of the patterns and T is the number of iterations.

3. Semi-supervised fuzzy c-means using kernel techniques

3.1. Semi-supervised kernel fuzzy c-means clustering

Normally, fuzzy clustering algorithms will determine prototype of clusters depending on the structure of patterns. When a small amount of patterns in the entire datasets could be labeled, the semi-supervised clustering algorithms are implemented on the combination of the labeled and unlabeled data to improve performance.

In the proposed method, the labeled data was used to calculate the rudimentary centroid of clusters, denoted V^*. The idea of the approach is to use the rudimentary centroids V^* to adjust centroids to move closer to V^* by extending to Semi-supervised Kernel Fuzzy C-Means in feature space (SKFCM-F).

The measure of the difference between the rudimentary clusters and the final clusters is determined as follows:

$$\left\|\phi(v_i^*) - \phi(v_i)\right\|^2 = k(v_i^*,v_i^*) + k(v_i,v_i) - 2k(v_i^*,v_i). \tag{6}$$

The distance d_{ij} between the pattern x_j and the prototype v_i in the kernel space is computed as follows:

$$\left\|\phi(x_j) - \phi(v_i)\right\|^2 = k(x_j,x_j) + k(v_i,v_i) - 2k(x_j,v_i). \tag{7}$$

The prototype v_i are constructed in the kernel space so we obtain the objective function as follow:

$$\begin{aligned} J_m(U,v) &= \sum_{j=1}^{n}\sum_{i=1}^{c}(u_{ij})^m[\|\phi(x_j) - \phi(v_i)\|^2 \\ &\quad + \|\phi(v_i^*) - \phi(v_i)\|^2]. \end{aligned} \tag{8}$$

In which u_{ij} are satisfied the constrain $\sum_{i=1}^{C} u_{ij} = 1$, n is the number of patterns, c is the number of clusters. When minimizing the objective function, Lagrange multiplier is used to find the solution by the following function:

$$\begin{aligned} L(u_{ij},\lambda_j) &= \sum_{j=1}^{n}\sum_{i=1}^{c}(u_{ij})^m[\left\|\phi(x_j) - \phi(v_i)\right\|^2 \\ &+ \left\|\phi(v_i^*) - \phi(v_i)\right\|^2] + \sum_{j=1}^{n}\lambda_j(1 - \sum_{i=1}^{c} u_{ij}). \end{aligned} \tag{9}$$

Calculate the first derivative of function $L(u_{ij},\lambda_j)$ follow u_{ij} and v_i

$$\begin{aligned} &\nabla_{u_{ij}} L(u_{ij},\lambda_j) = 0 \\ &m.u_{ij}^{m-1}[\left\|\phi(x_j) - \phi(v_i)\right\|^2 + \left\|\phi(v_i^*) - \phi(v_i)\right\|^2] - \lambda_j = 0 \\ &m.u_{ij}^{m-1}[2(1-k(x_j,v_i)) + 2(1-k(v_i^*,v_i))] - \lambda_j = 0 \\ &u_{ij} = \left(\frac{\lambda_j}{2m[2-k(x_j,v_i)-k(v_i^*,v_i)]}\right)^{1/(m-1)} \end{aligned} \tag{10}$$

with constrain $\sum_{i=1}^{c} u_{ij} = 1$ we have:

$$u_{ij} = \left(\frac{\frac{1}{2[2-k(x_j,v_i)-k(v_i^*,v_i)]}}{\sum_{j=1}^{c}\left[\frac{1}{2[2-k(x_j,v_i)-k(v_i^*,v_i)]}\right]^{\frac{1}{m-1}}}\right)^{\frac{1}{m-1}} \tag{11}$$

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^m[k(x_j,v_i)\,x_j + k(v_i^*,v_i)\,v_i^*]}{\sum_{j=1}^{N} u_{ij}^m[k(x_j,v_i) + k(v_i^*,v_i)]}. \tag{12}$$

With Gaussian kernel $k(x,y) = \exp\left(-\|x-y\|^2/r^2\right)$, we obtain the formulas of membership function (11) and prototypes (12). The following is the detailed steps of the SKFCM-F algorithm.

Algorithm 1 (*Semi-supervised Kernel Fuzzy C-Means (SKFCM-F)*).

Input: Given a set of n patterns $X = \{x_i\}_{i=1}^{n}$ and the desired number of clusters c.

Output: Membership matrix $U = \{U_{ij}\}_{i,j=1}^{n,c}$.

Step 1: Estimate the rudimentary centroids from labeled data.

1.1 Extract labeled patterns from dataset.

1.2. Calculating centroid $V^* = [v_i^*], v_i^* \in R^n$ from the labeled patterns.

Step 2: Initialization

2.1 Choose fuzzifier m, $(1 < m)$, error e.

2.2 Initialization membership matrix $U^{(0)}{}_{ij}$.

Step 3: Repeat

3.1. Update centroids $V^j = [v_1^j, v_2^j, \dots, v_c^j]$ by using the formula (12).

3.2. Compute the membership matrix $U_{ij}^{(t)}$ by the formulas (11).

3.3 Verify if the termination condition is satisfied: If $Max(|U^{(t)} - U^{(t-1)}|) < \varepsilon$, go to step 4.

Step 4: Report results clustering.

4.1. Return $U^{(t)}$ and assign a pattern to a cluster.

4.2. Report results of clustering.

3.2. Semi-supervised multiple kernel fuzzy c-means clustering

The kernel-based methods deals with the difficulty in the combination or selection of the best kernels among the extensive possibilities. This combination is often influenced strongly by prior knowledge about the data and by the patterns which are discovered. Besides, many real-world clustering problems often contain many useful features when combining them together. Therefore, it is necessary to use multiple kernels with their weights to aggregate for features from different sources into a final kernel function. A semi-supervised multiple kernel fuzzy c-means (SMKFCM) algorithm is extended from SKFCM-F by combining different kernels to obtain better results.

SMKFCM maps the data from the feature space into kernel space H by using transform functions: $\psi = \{\psi_1, \psi_2, \dots, \psi_M\}$ where $\psi_k(x_i)^T\psi_k(x_j) = K_k(x_i, x_j)$ and $\psi_k(x_i)^T\psi_{k'}(x_j) = 0 | k \neq k'$.

The prototypes v_i is constructed in the kernel space, the general framework of SMKFCM aims to minimize the objective function like the SKFCM-F function:

$$J_m(U, v) = \sum_{j=1}^{n}\sum_{i=1}^{c}(u_{ij})^m[\left\|\psi\left(x_j\right) - v_i\right\|^2 + \|v_i^* - v_i\|^2]. \tag{13}$$

In which, $\sum_{i=1}^{c} u_{ij} = 1$, n is the number of patterns, c is the number of clusters, $\psi(x) = \omega_1\psi_1(x) + \omega_2\psi_2(x) + \dots + \omega_M\psi_M(x)$.

Subject to:$\omega_1 + \omega_2 + \dots + \omega_M = 1$ and $\omega_k \geq 0, \forall k$. where v_i is the centroid of the ith cluster in the kernel space, $(\omega_1, \omega_2, \dots, \omega_M)$ is a vector of weights for features, respectively. The distance d_{ij} concerns the jth data (pattern) and the ith prototype:

$$d_{ij}^2 = (\psi(x_j) - v_i)^2 = (\psi(x_j) - v_i)^T(\psi(x_j) - v_i).$$

Some learning algorithms could automatically adjust the weights ω_k on a typical kernel learning method like multiple-kernel regression and classification which have been studied (Sahoo et al., 2014; Gonen and Alpaydn, 2010). Here, we propose a similar algorithm for SMKFCM using linearly combined kernels on the typical kernels such as Gaussian kernel and polynomial kernel. By introducing the Lagrange term of the constraint of weights into the objective function, define as follows:

$$\begin{aligned} L(v_i, u_{ij}, \omega_k) = &\sum_{j=1}^{n}\sum_{i=1}^{c}(u_{ij})^m(\left\|\psi\left(x_j\right) - v_i\right\|^2 + \|v_i^* - v_i\|^2) \\ &+ \sum_{j=1}^{n}\lambda_j(1 - \sum_{i=1}^{c}u_{ij}) + \sum_{j=1}^{n}\beta_j(1 - \sum_{k=1}^{M}\omega_k) \end{aligned} \tag{14}$$

with λ_j, β_j are constants. Optimizing the objective function (14) is expressed as:

$$\frac{\partial L}{\partial v_i} = 0, \frac{\partial L}{\partial u_{ij}} = 0, \frac{\partial L}{\partial \omega_k} = 0$$

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^m(\psi(x_j) + v_i^*)}{2\sum_{j=1}^{n} u_{ij}^m} \tag{15}$$

$$u_{ij} = \frac{\left(\frac{1}{m((\psi(x_j)-v_i)^2+(v_i^*-v_i)^2)}\right)^{\frac{1}{m-1}}}{\sum_{i=1}^{c}\left(\frac{1}{m((\psi(x_j)-v_i)^2+(v_i^*-v_i)^2)}\right)^{\frac{1}{m-1}}} \tag{16}$$

$$\omega_k = \frac{\beta_j + 2\sum_{i=1}^{c} u_{ij}^m v_i \psi_k(x_j)}{2\sum_{i=1}^{c} u_{ij}^m \psi_k^T(x_j)\psi_k(x_j)}. \tag{17}$$

With $\omega_1 + \omega_2 + \dots + \omega_M = 1$ and after some mathematical transformations we have:

$$\beta_j = 2\sum_{k=1}^{M}\sum_{i=1}^{c} u_{ij}^m \psi_k^T(x_j)\psi_k(x_j)\left(1 - \sum_{k=1}^{M}\frac{\sum_{i=1}^{c} u_{ij}^m v_i \psi_k(x_j)}{\sum_{i=1}^{c} u_{ij}^m \psi_k^T(x_j)\psi_k(x_j)}\right).$$

Now we can calculate the distance d_{ik} concerns the jth data and the ith prototype as:

$$d_{ij}^2 = \psi(x_j)^T * \psi(x_j) - 2\psi(x_j)^T * v_i + (v_i)^T * v_i.$$

By replacing the v_i in (15) and $\psi_k(x_i)^T\psi_k(x_j) = K_k(x_i, x_j)$ to the above equations, we have:

$$\begin{aligned} d_{ij}^2 = &\sum_{k=1}^{M}\omega_k^2 K_k(x_j, x_j) - \frac{\sum_{k=1}^{M}\sum_{j=1}^{n} u_{ij}^m \omega_k^2(K_k(x_j, x_j) + K_k(x_j, v_i^*))}{\sum_{j=1}^{n} u_{ij}^m} \\ &+ \frac{\sum_{k=1}^{M}\sum_{j=1}^{n} u_{ij}^{2m}\omega_k^2(K_k(x_j, x_j) + 2K_k(x_j, v_i^*) + K_k(v_i^*, v_i^*))}{\left(\sum_{j=1}^{n} u_{ij}^m\right)^2} \end{aligned} \tag{18}$$

$$\begin{aligned} \beta_j = &\ 2\sum_{k=1}^{M}\sum_{i=1}^{c} u_{ij}^m K_k(x_j, x_j) \\ &\times \left(1 - \sum_{k=1}^{M}\frac{\sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^m(K_k(x_j, x_j) + K_k(x_j, v_i^*))}{2\sum_{j=1}^{n}\sum_{i=1}^{c} u_{ij}^m K_k(x_j, x_j)}\right) \end{aligned} \tag{19}$$

$$\omega_k = \frac{\beta_j + \sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^m(K_k(x_j, x_j) + K_k(x_j, v_i^*))}{2\sum_{j=1}^{n}\sum_{i=1}^{c} u_{ij}^m K_k(x_j, x_j)}. \tag{20}$$

To construct multi-kernel, we consider Gaussian kernel as K_1 and Polynomial kernel as K_2:

$$K_1(x, y) = \exp\left(-\|x - y\|^2/r^2\right) \tag{21}$$

$$K_2(x, y) = \left(x^T.y + d\right)^p \tag{22}$$

where $r, d \in R^+, p \in N^+$.

The following is the detailed steps of the SMKFCM algorithm.

Algorithm 2 (*Semi-supervised Multiple Kernel Fuzzy C-Means (SMKFCM)*).

Input: Given a set of n patterns $X = \{x_i\}_{i=1}^n$, a set of kernel functions $\{K_k\}_{k=1}^M$, and the desired number of clusters c.

Output: Membership matrix $U = \{u_{ij}\}_{i,j=1}^{n,c}$ and weights $\{\omega_k\}_{k=1}^M$ for the kernels. To construct multiple kernels, we use Gaussian kernel as K_1 and Polynomial Kernel as K_2.

Step 1: Estimating centroids from the labeled data

1.1 Extracting the labeled patterns from dataset.

1.2. Calculating the rudimentary centroids $V^* = [v_i^*], v_i^* \in R^n$ from labeled patterns.

Step 2: Initialization

2.1 Choose fuzzifier m, $(1 < m)$, error e.

2.2 Initialize membership matrix $U^{(0)}$

Step 3: repeat:

3.1 Calculate constants β_j by using (19).

3.2 Update weights ω_k by using (20).

3.2 Calculate distance in kernel space d_{ij} by using (18).

3.3 Update memberships $u_{ij}^{(t)}$ by using (16).

3.4 Verify if the termination condition is satisfied:

If $Max(|u^{(t)} - u^{(t-1)}|) < \varepsilon$, go to step 4, otherwise go to step 3.

Step 4: Report results clustering.

4.1. Return $U^{(t)}$ and ω_k with $k = 1, 2, \dots, M$.

4.2. Assign a pattern to a cluster and report results of clustering.

4. Experiments

4.1. Experiments 1

The first experiment was implemented on the well-known datasets consisting of Urban Land Cover (ULC) with 168 instances and 148 attributes, Wine Quality (WQ) with 4898 instances and 12 attributes, Forest Type Mapping (FTM) with 326 instances and 27 attributes, Iris with 150 instances and 4 attributes.[1] To evaluate the classification results, we deployed the previous algorithms such as SFCM (Mai and Ngo, 2015), S2KFCM (Zhang et al., 2004) and PS3VM (Karvelis et al., 2000) to compare with the proposed algorithms (SMKFCM and SKFCM-F).

The aim of the classification, the ULC data is to distinguish between three classes of water, ponds, lakes; plants and buildings, roads. The WQ data is to distinguish quality between good wine or bad wine. The FTM data is to distinguish three types of forest and the Iris dataset contains three classes, where each class refers to a type of the Iris plants.

The parameters and terminal conditions: The number of iterations $L = 30$ and the error $\sigma < 0.00001$, the fuzzy parameter m is set to 2. Set $\delta^2 = 4$ in kernel K_1 and $\theta = 10$ and $p = 2$ in kernel K_2. The classifiers was performed 30 times to take the averages, the labeled data accounted for 5%, 10%, 15% and 30% of the dataset ULC, WQ, FTM and Iris, respectively.

There are several terms that are commonly used along with the description of accuracy. We should distinguish between misclassification of positive samples (FP) and negative samples (FN) and correct classification of positive samples (TP) and negative samples (TN). Thus, TP + FN is number of all positive assessment, FP + TN is number of all negative assessment. To assess the accuracy of the classification results, the performance of the classification was evaluated with the True Positive Rate (TPR) and False Positive Rate (FPR) which are defined by the following equations:

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{23}$$

where TPR is the ratio between the number of true positive assessment and the number of all positive assessment, TP is the number of correctly classified data and FN is the number of incorrectly misclassified data.

$$\mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \tag{24}$$

with TPR is the ratio between the number of true negative assessment) and the number of all negative assessment, FP is the number of incorrectly classified data and TN is the number of correctly misclassified data. According to the formula (23), the value of TPR is as large as possible, while the value of FPR is as small as possible (formula (24)). The results of the clustering or the quality of classification is shown in the indicators TPR and FPR. The efficient algorithms have larger TPR value and smaller FTR value.

Table 1 shows that the correctly classifying ratios obtain the best values coming from SMKFCM algorithm on datasets ULC, FTM and IRIS, followed by PS3VM algorithm on dataset WQ. While SFCM algorithm produces the classes with the lowest rate.

On the datasets ULC, FTM and IRIS, the SMKFCM algorithm obtain the largest TPR of 96.32%, 96.48% and 97.06%, respectively, and higher than remaining algorithms. While the FPR produced by SMKFCM algorithm obtain the smallest values of 1.15%, 1.09% and 1.26%, respectively.

On dataset WQ, PS3VM algorithm obtains the best TPR of 96.92% in comparison with 96.88% from SMKFCM and the FPR of 1.07% respecting to 1.04% from SMKFCM. That means PS3VM and SMKFCM produce the approximate ratios on dataset WQ.

In summary, the experiment exhibits that SMKFCM obtains the better ratios than algorithms like PS3VM, SKFCM-F, S2KFCM and SFCM in the most cases on the considered datasets (ULC, WQ, FTM and IRIS).

4.2. Experiments 2: Land cover classification

The second experiment mentions the problem of classification on satellite imagery. In fact, several types of data normally have the fixture centroid of clusters, for example the centroid of clusters in classification of remote sensing images because of the physical characteristics of electromagnetic spectrum when reflecting from the land cover surface. Hence, fuzzy clustering algorithms could give the incorrect results of clusters, especially large different amount between clusters. Besides, clustering on complex shaped data, classification in kernel space potentially get the better results than Euclidean space.

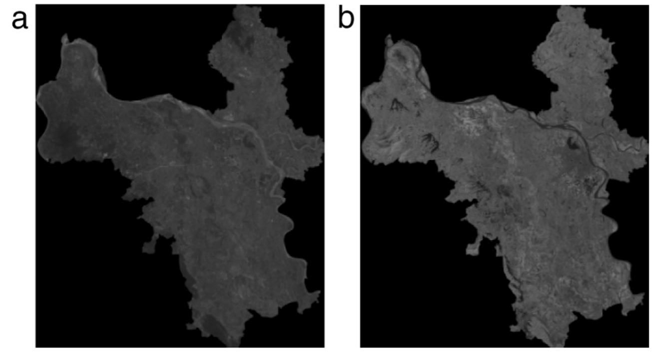

Fig. 1. Landsat-7 satellite imagery: (a) Band 3; (b) Band 4.

Data is divided into two sets consisting of the labeled and unlabeled parts. The labeled data is used to estimate the centroid of clusters. These centroids are called the rudimentary centroids, denote $V^* = [v_1^*, v_2^*, \ldots, v_c^*]$, which are used into processing the satellite image segmentation.

The detail of the land-cover classification algorithm consists of the following three main steps:

Algorithm 3 (*Land-cover Classification Algorithm Using SMKFCM Algorithm*).

Step 1: Preprocessing step for multi-spectral satellite imagery.

Step 2: Multi-spectral satellite imagery dataset will be classified into six classes representing six types of land covers:

1. Class1: Rivers, ponds, lakes.
2. Class2: Rocks, bare soil.
3. Class3: Fields, grass.
4. Class4: Planted forests, low woods.
5. Class5: Perennial tree crops.
6. Class6: Jungles.

Step 3: Compute the percentage of the individual regions:

$$S_i = \frac{n_i}{N} \tag{25}$$

where S_i is area of the ith region, n_i is the number of pixels of the $i$thregion, N is the total samples of n-bands imagery.

4.2.1. Study data-sets

Study area (dataset) no. 1 is Hanoi area located in (11°24′02.32″ N, 107°36′26.74″ E to 10°50′24.61″ N, 108°09′50.57″ E) from the LANDSAT-7 imagery source which two bands No. 3 and No. 4 are displayed in Fig. 1. Resolution of the imagery is 30 m, the area is 3161.304 km^2 with capacity of 187.59 MB.

Study area (dataset) no. 2 is Bao Loc city, Lam Dong province, Vietnam (11°18′29.13″ N, 108°18′10.57″ E to 11°58′29.63″ N, 107°01′44.93″ E) with the area of 1707.31 km^2 and total of pixels is 1,897,008.

Study area (dataset) no. 3 comes from Landsat7 satellite images of Thai Nguyen city, Vietnam (105°37′16.0190″ E, 21°37′39.8284″ N and 105°59′49.7296″ E, 21°28′58.9896″ N) with the area of 614.43 km^2 in 2014 and total of pixels is 411 045.

The second series of experiments concern multi-spectral remote sensing images. The pixel information is composed from different bands as a feature vector. The different kernels for pixel intensities are defined by

[1] http://archive.ics.uci.edu/ml/datasets.html.

Table 1
Classification results of SFCM, S2KFCM, PS3VM, SKFCM-F and SMKFCM.

Datasets	Rate	SFCM	S2KFCM	PS3VM	SKFCM-F	SMKFCM
ULC	TPR(%)	88.21 ± 3.12	92.86 ± 2.98	95.56 ± 1.56	93.16 ± 2.42	**96.32 ± 1.32**
c = 3	FPR(%)	4.31 ± 1.14	3.84 ± 1.85	1.61 ± 0.92	1.38 ± 0.63	1.15 ± 0.46
WQ	TPR(%)	90.83 ± 3.95	92.06 ± 3.01	**96.92 ± 0.98**	93.96 ± 2.69	96.88 ± 1.01
c = 2	FPR(%)	3.98 ± 1.46	4.11 ± 1.19	1.07 ± 0.41	1.24 ± 0.65	1.04 ± 0.28
FTM	TPR(%)	89.05 ± 2.92	93.27 ± 2.91	96.32 ± 1.02	93.18 ± 1.02	**96.48 ± 0.97**
c = 3	FPR(%)	3.19 ± 1.68	3.02 ± 1.14	1.23 ± 0.87	1.09 ± 0.68	1.09 ± 0.68
IRIS	TPR(%)	90.87 ± 3.98	91.58 ± 3.88	95.89 ± 1.21	94.98 ± 1.02	**97.06 ± 0.82**
c = 3	FPR(%)	4.01 ± 2.08	3.13 ± 1.22	1.97 ± 0.99	1.86 ± 0.83	1.26 ± 0.63

applying the combined kernel in a multiple-kernel learning algorithm. The kernel algorithm uses Gaussian kernel K_1 for pixel intensities and the multiple kernel algorithm uses Gaussian kernel K_1 and polynomial kernel K_2 for pixel intensities. Thus, u_{ij} and V values can be calculated according to the formulas (11) and (12) in the SKFCM-F and u_{ij}, β_j, ω_k, d_{ij} according to the formulas (16), (18), (19) and (20) in Algorithm SMKFCM.

The number of iterations $L = 30$ and the error $\sigma < 0.00001$, the fuzzy parameter $m = 2$ are used in this experiment. Set $\delta^2 = 4$ in kernel K_1 and $\theta = 10$ and $p = 2$ in kernel K_2. The labeled data corresponds to around 5% to 10% for each class.

We use two bands No. 3 and No. 4 to compute the NDVI index (Normalized Difference Vegetation Index) which is the most common measurement to assess the growth and distribution of the vegetation on the earth's surface.

$$\mathrm{NVDI} = \frac{\mathrm{NIR} - \mathrm{VR}}{\mathrm{NIR} + \mathrm{VR}}. \tag{26}$$

In which, NIR (Near-Infrared) and VR (Visible Red) corresponding to band No. 3 and band No. 4 in the 7-bands LANDSAT imagery.

The value of NDVI index of a pixel takes in the range [−1, 1]. The no-vegetable pixel takes a value around zero. The value of almost 1.0 means the highest density of vegetable. Low values of NDVI (around 0.1) correspond to barren areas of rock, sand, or snow. Moderate values represent shrub and grassland (0.2 to 0.3), while high values indicate temperate and tropical rainforests (0.6 to 0.8). To process conveniently NDVI data, pixel based image is converted to NDVI based image by the following formula:

$$Pixel_{value} = (\mathrm{NDVI} + 1) * 127. \tag{27}$$

4.2.2. Validity measures

To assess the performance of algorithms on the experimental images, we considered several validity indexes such as the Bezdek's partition coefficient(PC-I) (Chen and Philip Chen, 2012), Dunn's separation index (DI), the Separation index (SI), Xie and Beni's index (XB-I), and Classification Entropy index (CE-I) (Bezdek and Pal, 1998; Wang and Zhang, 2007; Zhao et al., 2009). Note that the validity indexes are proposed to evaluate the quality of clustering. Algorithms producing better results come with smaller values of D-I, S-I, CE-I, XB-I and the larger value of PC-I.

– Partition coefficient: Bezdek designed the partition coefficient (PC-I) to measure the "overlapped" amount between clusters which is calculated as follows:

$$PC - I = \frac{1}{N}\sum_{i=1}^{C}\sum_{j=1}^{N} u_{ij}^2$$

where u_{ij} is the membership of data point j in cluster i.

– Dunn's separation index: The Dunn's index is defined as

$$D - I = \min_{i\in c}\left\{\min_{j\in c, j\neq i}\left\{\frac{\delta(A_i, A_j)}{\max_{k\in c}\{\Delta(A_k)\}}\right\}\right\}.$$

In which,

$$\delta(A_i, A_j) = \min\{d(x_i, x_j) | x_i \in A_i, x_j \in A_j\}$$
$$\Delta(A_k) = \max\{d(x_i, x_j) | x_i, x_j \in A\}$$

and d is a distance function and A_j is the set whose elements are assigned to the ith cluster.

– Separation index: The validity index S is based on the objective function J by determining the average number of data and the square of the minimum distances of the cluster centers.

$$S - I = \frac{1}{N}\frac{\sum_{i=1}^{C}\sum_{j=1}^{N} u_{\mathrm{ij}}^2 \|x_j - v_i\|^2}{\min_{m,n=1,\ldots,c;\, n\neq m} \|v_m - v_n\|^2}$$

where d_{min} is the minimum Euclidean distance between cluster centroids.

– Classification entropy: The classification entropy (CE) is similar to the PC but it measures the fuzziness of the cluster partition only by Bezdek.

$$CE - I = -\frac{1}{N}\sum_{i=1}^{C}\sum_{j=1}^{N} u_{\mathrm{ij}} \log(u_{\mathrm{ij}})$$

– Xie and Beni's index: The Xie and Beni's index (XB) aims to quantify the ratio of the total variation within clusters and the separation of clusters by Xie and Beni.

$$XB - I = \frac{1}{N}\frac{\sum_{i=1}^{C}\sum_{j=1}^{N} u_{\mathrm{ij}}^m \|x_j - v_i\|^2}{\min_{i,j}\|x_j - v_i\|^2}.$$

Table 2
Hanoi area: Result of land cover classification.

Class	N. of pixels	Percentage (%)	Square (hec.)
1	211 491	6.021	19034.211
2	663 663	18.894	59729.678
3	888 081	25.283	79927.249
4	689 270	19.623	62034.268
5	652 001	18.562	58680.125
6	408 054	11.617	36724.869

4.2.3. Experimental results and discussion

We have implemented classification on the different algorithms such as SFCM, S2KFCM, PS3VM, SKFCM-F and SMKFCM.

The experimental results are shown in Fig. 2 in which (a), (b), (c), (d), (e) and (f) are NDVI image, the classification results of SFCM, S2KFCM, PS3VM, SKFCM-F and SMKFCM, respectively. Table 2 and Fig. 3 show the detailed classification produced by SMKFCM algorithm according to the number of pixels, the percentage and the area of the individual classes.

Table 3 shows the comparative results between SFCM, S2KFCM, PS3VM, SKFCM-F and SMKFCM and the data of The Vietnamese Center of Remote Sensing Technology (VCRST) on each class (in percentage %). There is the significant difference between the algorithm SFCM, S2KFCM, PS3VM, SKFCM-F and SMKFCM in determining the area of regions, the largest difference is around 11%. Comparing the experimental results with the VCRST data, we also consider the accuracy by estimating the percentage of the class with the lowest deviation, the largest deviation and the average deviation of six layers (in percentage %). The largest differences produced by SFCM, S2KFCM, SKFCM-F,

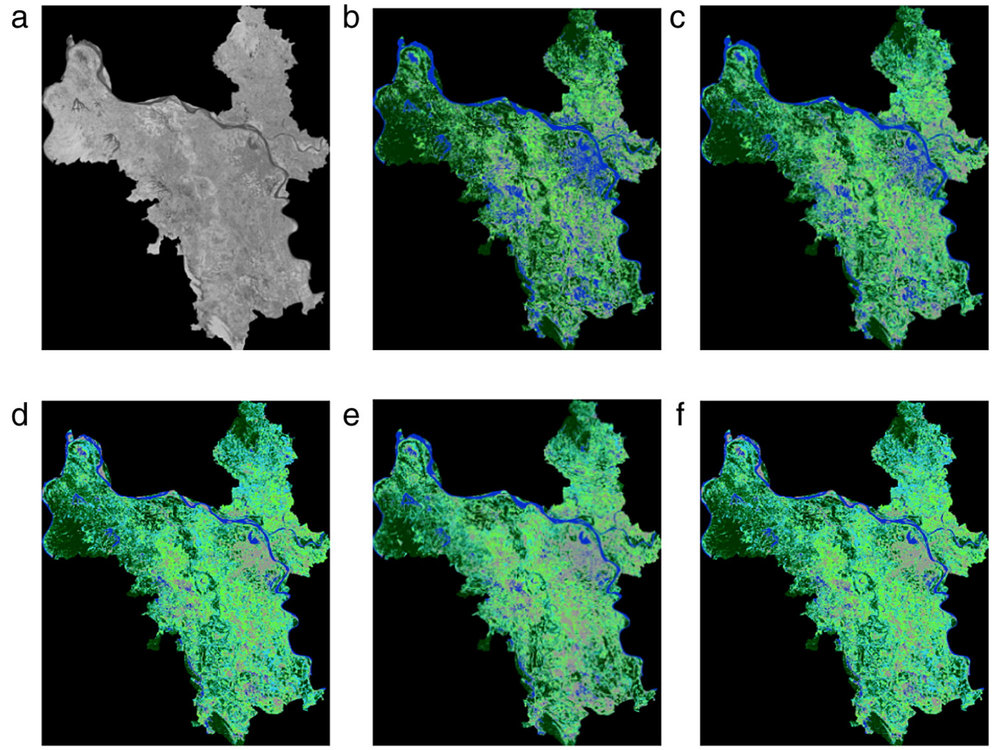

Fig. 2. Result of clustering (a) NDVI Image; (b) SFCM; (c) S2KFCM; (d) PS3VM; (e) SKFCM-F; (f) SMKFCM.

Table 3
Results of land cover classification in Hanoi.

Dataset	Class	VCRST	SMKFCM	SKFCM-F	PS3VM	S2KFCM	SFCM
Hanoi	Class 1	5.833%	6.021%	8.361%	6.389%	9.784%	11.234%
	Class 2	19.665%	18.894%	16.178%	18.169%	13.175%	10.763%
	Class 3	25.041%	25.283%	27.883%	25.517%	32.784%	35.821%
	Class 4	19.857%	19.623%	17.234%	19.273%	14.768%	12.428%
	Class 5	17.702%	18.562%	21.472%	19.627%	22.313%	23.274%
	Class 6	11.903%	11.617%	8.874%	11.026%	7.176%	6.481%

Table 4
The various validity indexes on the LANDSAT-7 images in Hanoi.

Algorithm	PC-I	D-I	S-I	CE-I	XB-I	Min deviation	Max deviation	Average deviation
SFCM	0.5821	0.5711	0.6726	0.9973	3.8734	5.401	10.780	7.2510
S2KFCM	0.6314	0.5422	0.5281	0.9287	2.0831	3.951	7.743	5.4352
PS3VM	0.7216	0.1093	0.1342	0.6241	1.7034	0.476	1.925	0.9857
SKFCM-F	0.6464	0.2548	0.2763	0.8872	1.9652	2.528	3.770	3.0463
SMKFCM	0.7376	0.1125	0.1045	0.5098	1.6912	0.188	0.860	0.4302

PS3VM are 10.78%, 7.743%, 3.77%, 1.925%, respectively. Meanwhile, the value obtained by SMKFCM algorithm is below 1.0% (see Table 4).

Besides, the validity indices in Table 4 show the proposed algorithm obtains better results than other algorithms. The validity indices produced by SMKFCM algorithm obtains the better values than the ones runned by other algorithms, i.e, the PC-I, SI, CE-I and XB-I gain values of 0.7376, 0.1045, 0.5098, 1.6912, respectively. The PC-I value is lower and higher values of S-I, CE-I and XB-I correspond to the PS3VM, SKFCM-F, S2KFCM and SFCM. While, the value D-I index obtains the smallest value of 0.1093 by running PS3VM algorithm, following to SMKFCM, SKFCM-F, S2KFCM and SFCM with 0.1125, 0.2548, 0.5422, 0.5711, respectively.

The results produced from the experiment on dataset no. 2 are demonstrated in Tables 5 and 6. Table 5 shows the percentages of the individual classes coming from algorithms of SFCM, S2KFCM, PS3VM, SKFCM-F and SMKFCM, and VCRST. In which, the differences between VCRST and SMKFCM takes the smallest values.

In Table 6, the validity indices produced by SMKFCM algorithm also the best values i.e, the larger values of PC-I, SI, CE-I and XB-I. While, the value D-I index obtains the smallest value of 0.1061 by running SMKFCM algorithm, following to PS3VM, SKFCM-F, S2KFCM and SFCM with 0.1076, 0.2548, 0.5422, 0.5711, respectively.

In this experiment on dataset no. 3, the smallest differences of the individual classes also comes with SMKFCM in Table 7. The indicators D-I, SI, CE-I and XB-I for SMKFCM algorithm obtain the smallest value of 0.1098, 0.1652, 0.4827 and 1.2871, respectively, in Table 8 and the PC-I also take the best result of 0.8763. Deviation of percentage area from six classes when compared to data collected from VCRST shows that the smallest average deviation of only 0.528% for SMKFCM algorithm, while the algorithm PS3VM, SKFCM-F, S2KFCM and SFCM is 1.077%, 2.442%, 3.882% and 6.157%, respectively.

These results exhibit the SMKFCM produces better clustering solution than the other algorithms such as PS3VM, SKFCM-F, S2KFCM and SFCM. With resolution of 30 m×30 m, classification results can be accepted in quickly assessment of land covers, reducing the expenses

Table 5
Results of land cover classification in Bao Loc, Lam Dong province.

Dataset	Class	VCRST	SMKFCM	SKFCM-F	PS3VM	S2KFCM	SFCM
Bao Loc	Class 1	2.076%	2.130%	2.374%	2.490%	3.004%	4.373%
	Class 2	10.259%	11.227%	13.831%	11.099%	15.359%	17.495%
	Class 3	18.756%	19.806%	21.085%	21.085%	22.764%	25.074%
	Class 4	26.838%	27.824%	30.149%	27.940%	31.904%	36.639%
	Class 5	23.199%	21.532%	15.311%	18.625%	14.138%	9.951%
	Class 6	18.873%	17.481%	17.249%	18.761%	12.830%	6.468%

Table 6
The various validity indexes on the LANDSAT-7 images in Bao Loc, Lam Dong province.

Algorithm	PC-I	D-I	S-I	CE-I	XB-I	Min deviation	Max deviation	Average deviation
SFCM	0.4896	0.6851	0.6986	0.9938	4.0982	2.297	13.248	8.551
S2KFCM	0.6482	0.4982	0.3761	0.9492	2.8723	0.928	9.061	5.034
PS3VM	0.7653	0.1076	0.1089	0.7862	1.6982	0.112	4.574	1.562
SKFCM-F	0.6976	0.3652	0.1974	0.9273	2.0874	0.298	7.888	3.170
SMKFCM	0.7784	0.1061	0.0988	0.4762	1.3723	0.054	1.667	1.019

Table 7
Results of land cover classification in Thai Nguyen city.

Dataset	Class	VCRST	SMKFCM	SKFCM-F	PS3VM	S2KFCM	SFCM
Thai Nguyen	Class 1	2.631%	2.722%	3.778%	3.247%	3.918%	5.702%
	Class 2	14.894%	15.592%	18.036%	15.989%	20.028%	22.813%
	Class 3	24.458%	25.253%	27.495%	25.979%	29.685%	31.939%
	Class 4	17.559%	17.469%	15.813%	16.723%	14.310%	12.827%
	Class 5	23.428%	22.869%	20.724%	21.937%	18.436%	15.251%
	Class 6	17.029%	16.095%	14.153%	16.125%	13.623%	11.467%

Table 8
The various validity indexes on the LANDSAT-7 images in Thai Nguyen city.

Algorithm	PC-I	D-I	S-I	CE-I	XB-I	Min deviation	Max deviation	Average deviation
SFCM	0.5287	0.4272	0.7297	0.9652	5.8936	3.071	8.177	6.157
S2KFCM	0.5872	0.3987	0.5872	0.8869	4.0673	1.286	5.227	3.882
PS3VM	0.7987	0.1265	0.1986	0.5091	1.9033	0.616	1.521	1.077
SKFCM-F	0.6098	0.2987	0.3981	0.7898	2.6732	1.147	3.142	2.442
SMKFCM	0.8763	0.1098	0.1652	0.4827	1.2871	0.090	0.934	0.528

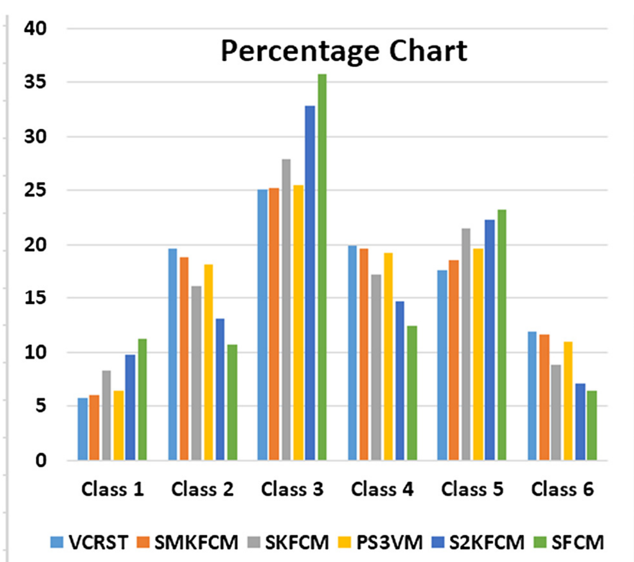

Fig. 3. The result of algorithms: VCRST data, SMKFCM, SKFCM-F, PS3VM, S2KFCM and SFCM.

than the traditional methods. These results not only make predictions about the land cover changes, but also support urban planning, natural resources management etc.

5. Conclusion

In fact, a small amount of the labeled data is able to improve the quality of classification results. This paper describes an approach on semi-supervised fuzzy clustering for satellite images using kernel technique and centroid information retrieved from the labeled data part. The kernel techniques used, involve of two cases: single kernel function for all data features and multiple kernel functions for data features, i.e. spatial features and Landsat-band valued features. The proposed method improves the clustering results and overcomes the drawbacks of the conventional clustering algorithms. The experiments were carried out based on the well-known datasets (ULC, WQ, FTM and Iris) and satellite image segmentation on three datasets (Hanoi, Bao Loc and Thai Nguyen) in Vietnam.

The next goals are to implement further research on hyper-spectral satellite imagery for environmental problems and assessment of land surface temperature changes or speed-up the proposed methods based on GPUs platforms.

Acknowledgment

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.05-2016.09;

References

Bezdek, J., Pal, N., 1998. Some new indexes of cluster validity. IEEE Trans. Syst. Man Cybern. 28 (3), 301–315.
Bezdek, James C., Ehrlich, Robert, William, Full, 1984. FCM: The fuzzy c-meams clustering algorithm. Comput. Geosci. 10 (2–3), 191–203.

Chen, L., Philip Chen, C.L., 2012. A multiple-kernel fuzzy c-means algorithm for image segmentation. IEEE Trans. Syst. Man Cybern. 41 (5), 1263–1274.
Chiang, J.-H., Hao, P.-Y., 2003. A new kernel-based fuzzy clustering approach: support vector clustering with cell growing. IEEE Trans. Fuzzy Syst. 11 (4), 518–527.
Frigui, Hichem, Bchir, Ouiem, Baili, Naouel, 2013. An overview of unsupervised and semi-supervised fuzzy kernel clustering. Int. J. Fuzzy Logic Intell. Syst. 13 (4), 254–268.
Ganesh, M., Palanisamy, V., 2012. Multiple-kernel fuzzy c-means algorithm for satellite image segmentation. Eur. J. Sci. Res. 83 (2), 255–263.
Genitha, C.H., Vani, K., 2013. Classification of satellite images using new fuzzy cluster centroid for unsupervised classification algorithm. In: 2013 IEEE Conference on Information and Communication Technologies (ICT). pp. 203–207.
Girolami, M., 2002. Mercer kernel-based clustering in feature space. IEEE Trans. Neural Netw. 13 (3), 780–784.
Gonen, Mehmet, Alpaydın, Ethem, 2010. Localized multiple kernel regression. In: 2010 IEEE International Conference on Pattern Recognition. pp. 1425–1428.
Gordo, O., Martinez, E., Gonzalo, C., Arquero, A., 2013. Classification of satellite images by means of fuzzy rules generated by a genetic algorithm. Lat. Am. Trans. 9 (1), 743–748. Revista IEEE 2013 America Latina.
Graves, D., Pedrycz, W., 2007. Fuzzy c-means, GustafsonKessel FCM, and kernel-based FCM: a comparative study. Adv. Soft Comput. 41, 140–149.
Graves, D., Pedrycz, W., 2010. Kernel-based fuzzy clustering and fuzzy clustering: A comparative experimental study. Fuzzy Sets Syst. 161 (4), 522–543.
Hathaway, Richard J., Bezdek, James C., Huband, Jacalyn M., 2005. Kernelized non-Euclidean relational c-means algorithms. Neural Parallel Sci. Comput. 13, 305–326.
Hoppner, F., Klawonn, F., Kruse, R., Runkler, T., 1999. Fuzzy Cluster Analysis. New York.
Huang, H.C., Chuang, Y.Y., Chen, C.S., 2012. Multiple kernel fuzzy clustering. IEEE Trans. Fuzzy Syst. 20 (1), 120–134.
Karvelis, P., Likas, A., Fotiadis, D.I., 2000. Semi unsupervised MFISH chromosome image classification. In: 10th IEEE Int Cont on Digital Object Identifier. pp. 1–4.
Liu, Y., Zhang, B., Wang, L., Wang, N., 2013. A self-trained semi-supervised SVM approach to the remote sensing land cover classification. Comput. Geosci. 59, 98–107.
Mai, Dinh Sinh, Ngo, Long Thanh, 2015. Semi-supervised fuzzy c-means clustering for change detection from multispectral satellite image. In: IEEE Conf. on Fuzzy Systems.
Ngo, Long Thanh, Mai, Dinh Sinh, Pedrycz, Witold, 2015. Semi-supervised interval type-2 fuzzy c-means clustering with spatial information for multi-spectral satellite image classification and change detection. Comput. Geosci. 83, 1–16.
Rakotomamonjy, A., Bach, F., Canu, S., Grandvalet, Y., 2007. More efficiency in multiple kernel learning. In: Proc. 24th Int. Conf. Machine Learning. pp. 775–782.
Roy, M., Ghosh, S., Ghosh, A., 2014. A novel approach for change detection of remotely sensed images using semi-supervised multiple classifier system. Inf. Sci. 269, 35–47.
Sahoo, Doyen, Hoi, Steven C.H., Li, Bin, 2014. Online multiple kernel regression. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 293–302.
Shawe-Taylor, J., Cristianini, N., 2004. Kernel Methods for Pattern Analysis. Cambridge University Press.
Shen, H., Yang, J., Wang, S., Liu, X., 2006. Attribute weighted mercer kernel based fuzzy clustering algorithm for general non-spherical datasets. Soft Comput. 10 (11), 1061–1073.
Stavrakoudis, D.G., Galidaki, G.N., Gitas, I.Z., Theocharis, J.B., 2010. Enhancing the interpretability of genetic fuzzy classiers in land cover classication from hyperspectral satellite imagery. In: IEEE World Congress on Computational Intelligence, WCCI2010, pp. 1277–1284.
Stavrakoudis, D.G., Theocharis, J.B., Zalidis, G.C., 2011. A boosted genetic fuzzy classifier for land cover classification of remote sensing imagery. ISPRS J. Photogramm. Remote Sens. 66, 529–544.
Torahi, Ali Asghar, Rai, Suresh Chand, 2011. Land cover classification and forest change analysis, using satellite imagery — a case study in Dehdez area of Zagros Mountain in Iran. J. Geogr. Inf. Syst. 3, 1–11.
Tzortzis, G., Likas, A., 2009. The global kernel k-means algorithm for clustering in feature space. IEEE Trans. Neural Netw. 20 (7), 1181–1194.
Varma, M., Babu, B.R., 2009. More generality in efficient multiple kernel learning. In: Proc. 26th Annu. Int. Conf. Machine Learning. pp. 1065–1072.
Wang, Na, Li, Xia, Luo, Xuehui, 2008. Semi-supervised kernel-based fuzzy c-means with pairwise constraints. In: 2008 International Joint Conference on Neural Networks, IJCNN 2008, pp. 1098–1102.
Wang, W., Zhang, Y., 2007. On fuzzy cluster validity indices. Fuzzy Sets and Systems 158, 2095–2117.
Yu, C., Li, Y., Liu, A., Liu, J., 2011. A novel modified kernel fuzzy c-means clustering algorithm on image segmentation. In: The 14th International Conference on Computational Science and Engineering (CSE). pp. 621–626.
Yuan, Yuan, Lv, Haobo, Lu, Xiaoqiang, 2015. Semi-supervised change detection method for multi-temporal hyperspectral images. Neurocomputing 148, 363–375.
Yugander, P., Babu, J.S., Sunanda, K., Susmitha, E., 2012. Multiple kernel fuzzy c-means algorithm with ALS method for satellite and medical image segmentation. In: Int. Conf. on Devices, Circuits and Systems (ICDCS). pp. 244–248.
Zhang, D.Q., Chen, S.C., 2003a. Kernel-based fuzzy and possibilistic c-means clustering. In: Proceedings of the International Conference on Artificial Neural Networks, Istanbul, Turkey. pp. 122–125.
Zhang, D.Q., Chen, S.C., 2003b. Clustering incomplete data using kernel based fuzzy c-means algorithm. Neural Process. Lett. 18 (3), 155–162.
Zhang, D., Chen, S., 2004. A novel kernelized fuzzy c-means algorithm with application in medical image segmentation. Artif. Intell. Med. 32, 37–50.
Zhang, Daoqiang, Tan, Keren, Chen, Songcan, 2004. Semi-supervised kernel-based fuzzy c-means. In: ICONIP 2004. In: LNCS, vol. 3316, pp. 1229–1234.
Zhao, B., Kwok, J., Zhang, C., 2009. Multiple kernel clustering. In: Proc. 9th SIAM Int. Conf. Data Mining. pp. 638–649.