Journal of Science and Transport Technology
Journal homepage: https://jstt.vn/index.php/en

Application of secure semi-supervised fuzzy clustering in object detection from remote sensing images

Pham Quang Nam[1], Nguyen Long Giang[2], Le Hoang Son[3], Tran Manh Tuan[4*]
[1]Graduate University of Science and Technology, Hanoi, Vietnam
[2]Institute of Information Technology, Vietnam Academy of Science and Technology, Hanoi, Vietnam
[3]VNU Information Technology Institute, Vietnam National University, Hanoi, Vietnam
[4]Thuyloi University, Hanoi, Vietnam

Article info
Type of article:
Original research paper

DOI:
https://doi.org/10.58845/jstt.utt.2022.en.2.3.33-41

***Corresponding author:**
E-mail address:
tmtuan@tlu.edu.vn

Received: 10/07/2022
Revised: 22/09/2022
Accepted: 26/09/2022

Abstract: In recent years, landslides are taking place very seriously, and tend to increase in both scope and scale, threatening people's lives and properties. Therefore, timely detection of landslide areas is extremely important to minimize damage. There are many ways to detect landslide areas, in which the use of satellite images is also an option worthy of attention. When performing satellite image data collection, there are many outliers, such as weather, clouds, etc. that can reduce image quality. With low quality images, when executing the clustering algorithm, the best clustering performance will not be obtained. In addition, the fuzzy parameter is also an important parameter affecting the results of the clustering process. In this paper will introduce an algorithm, which can improve the results of data partitioning with reliability and multiple fuzzifier. This algorithm is named **TSSFC**. The introduced method includes three steps namely as “labeled data with **FCM**”, “Data transformation”, and “Semi supervised fuzzy clustering with multiple point fuzzifiers”.

The introduced TSSFC method will be used for landslide detection. The obtained results are quite satisfactory when compared with another clustering algorithm, CS3FCM (Confidence-weighted Safe Semi-Supervised Clustering).

Keywords: Semi supervised fuzzy clustering; Safe semi supervised fuzzy clustering; Multiple fuzzifiers: Fuzzy clustering.

1. INTRODUCTION

Clustering is the process of dividing data points into different data clusters, satisfying that the elements in one cluster have more similarities than the elements in other clusters [1,2]. In 1984, Bezdek [3] et al introduced the first fuzzy clustering algorithm, Fuzzy C-Means (FCM). This is an iterative algorithm and at each step it adjusts the cluster center and membership matrix to satisfy the predetermined objective function. Semi-supervised fuzzy clustering algorithms are built on top of fuzzy clustering algorithms combined with additional information. One of the most popular algorithms is the C-Means Semi-Supervised Fuzzy (SSFCM) method [4]. Many improvements of SSFCM were introduced to deal with various problems [5-7]. In the semi supervised fuzzy clustering algorithm, some data is incorrectly labeled. Therefore, Gan et al [8] proposed a safe semi supervised fuzzy clustering algorithm named CS3FCM to solve the

Published online 29/09/2022

above problem. CS3FCM is based on the confidence-weight of each sample to get high clustering performance. By changing the formula of the objective function, the clustering performance can be improved. Fuzzy parameter represents the uncertainty of each data element. Therefore, to increase the performance of fuzzy clustering algorithm, it is necessary to determine different values of m for each data element [9].

Outliers and noise are also factor that affect the performance of the clustering process. In many cases, the data may contain noise or inaccurate information. For example, when collecting satellite images of a landslide, due to the shooting angle or confounding factors such as clouds, fog, etc., the resulting image may contain noise. Therefore, when applying treatment techniques, landslides can be mistakenly identified as mountains. Process of dealing with incorrect data and noisy data is called the data partition with confidence problem, including “safe information” and “noisy data”. The objective of the data clustering problem with confidence can be stated that by using data clustering, the unlabeled data points will be properly labeled of clusters and incorrect labeled data points will be relabeled exactly.

In this paper, an improved algorithm for partitioning data with reliability problems using multiple fuzzifiers named as **TSSFC** is introduced. This method reconciles labeled data using modified FCM with the weights of unlabeled and labeled data neighbors instead of working on the whole dataset as in [8]. The differences of **TSSFC** comparing with CS3FCM is given as below:

i. Although CS3FCM uses all labeled data in the clustering process, **TSSFC** will either set a very low membership value or remove the data point from the original data set after applying the modified FCM, which has been labeled and has little impact on the clustering process.

ii. While CS3FCM only uses labeled data as additional information, **TSSFC** applies modified FCM and uses unlabeled data to calculate membership values, thereby obtaining cluster centers. Therefore, the member values of the unlabeled and labeled data are contained in the previous membership degrees ($\bar{U}$). The supporting information in the TSSFC is a combination of the labeled data and previous membership degrees ($\bar{U}$).

iii. To control the data clustering process, **TSSFC** uses multiple fuzzifiers for each data point. In this step, the previous membership degrees ($\bar{U}$) are used to support clustering progress in generating the final cluster centers and membership values for all data points. We use a semi supervised fuzzy clustering with multiple fuzzifiers method in order to partition the whole dataset with the initial membership ($\bar{U}$).

The introduced **TSSFC** method is implemented on specific datasets and experimentally compared with the CS3FCM.

The remainder of this paper consists of two main parts. The TSSFC method is described in Section 2. The test results of implementing TSSFC and CS3FCM on the test dataset are given in Section 3. We point out future research directions and draw conclusions in the final section.

2. METHOD

2.1. Main idea of TSSFC

TSSFC consists of 3 following steps:

Step 1. (FCM for labeled data)

Split the original data points into clusters by new weights based on unlabeled and labeled neighborhoods using the improved FCM algorithm.

Step 2. (Data transformation process)

To determine the membership levels of unlabeled data points it is necessary to use the cluster centers obtained in *Step 1*. The values of membership in both unlabeled and labeled data will produce the previous membership qualifications ($\bar{U}$) for the next step.

Step 3. (Semi supervised fuzzy clustering with multiple point fuzzifiers)

It is necessary to use a semi-supervised fuzzy clustering algorithm with multiple fuzzifiers to control the data clustering process.

The framework of **TSSFC** algorithm is given in Figure 1 as follows.

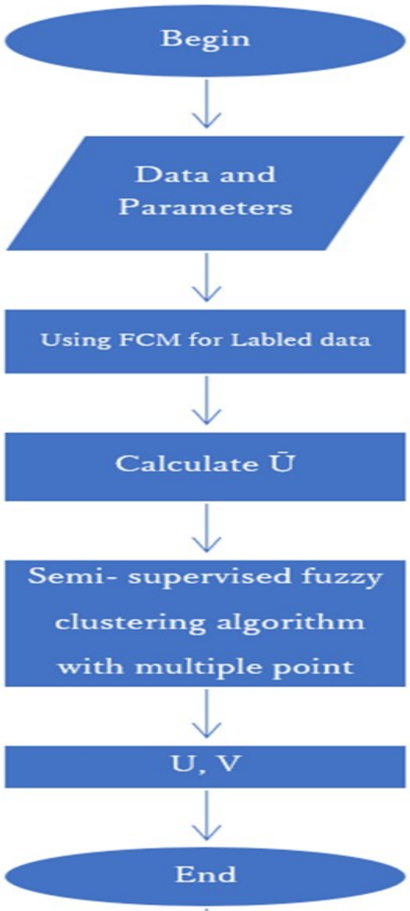

Begin
Data and Parameters
Using FCM for Labled data
Calculate Ū
Semi- supervised fuzzy clustering algorithm with multiple point
U, V
End

Figure 1. The flowchart of **TSSFC** algorithm

2.2. Details of the TSSFC

2.2.1. Step 1 (FCM for labeled data)

In this step, the algorithm compares the labeled data elements to identify the data elements with low and high confidence. To do this, we change the original FCM algorithm with the new objective as follows

$$J = \sum_{k=1}^{L}\sum_{i=1}^{C} \frac{n_{1k}+n_{2k}}{n_{3k}+1} u_{ki}^{m} d_{ki}^{2} \to Min \qquad (1)$$

$$\mathrm{u_{ki}} \in [0, 1]; \mathrm{i} = 1, ..., \mathrm{C}; \mathrm{k} = 1, ..., \mathrm{L} \qquad (2)$$

$$\sum_{i=1}^{C} \mathrm{u_{ki}} = 1 ;\ \mathrm{k} = 1,...,\mathrm{L} \qquad (3)$$

Where n_{3k} is the number of neighbors with different label to x_k; n_{2k} is the number of neighbors with the same label to x_k; n_{1k} is the number of unlabeled data neighbors. These neighbors are defined based on the radius R and are determined using the Euclidean distance. The value of *R* is calculated as $(d_{\max} - d_{\min})/10$ where $d_{\min}$, $d_{\max}$ are the minimum and maximum distance between two universal data points. The symbols *C*, *L* and d_{ki} are the number of clusters, expressed for the amount of labeled data, and the distance between i^{th} cluster center and k^{th} data point. Applying Lagrange method, the membership values and cluster center of the optimization problem (17-19) are specified as below.

$$V_i = \frac{\sum_{k=1}^{L} \frac{n_{1k}+n_{2k}}{n_{3k}+1} u_{ki}^{m} X_k}{\sum_{k=1}^{L} \frac{n_{1k}+n_{2k}}{n_{3k}+1} u_{ki}^{m}} ; i = 1,..,C \qquad (4)$$

$$u_{ki} = \frac{1}{\sum_{j=1}^{C} \left(\frac{d_{ki}}{d_{kj}}\right)^{\frac{2}{m-1}}} ; k = 1,..,L,\ i = 1,..,C \qquad (5)$$

When dealing with incorrectly labeled data, we use defuzzification to reduce its membership value. If the assigned cluster is different from the data point's label, then the uik membership value is correspondingly reduced according to equation (6).

$$u_{ki} = \begin{cases} \frac{u_{ki}}{2} \\ \text{if label of cluster } i \text{ is same to label of } x_k \\ u_{ki} + \frac{u_{kj}}{2(C-1)} \\ \qquad \text{if } i \neq j \text{ and label of cluster } j \\ \qquad \qquad \text{is same to label of } x_k \end{cases} \qquad (6)$$

The data point is labeled with small impact, or is set to a very low membership value, or is removed from the labeled data set. The modified FCM algorithm is described in Algorithm 1 below.

Algorithm 1. The modified FCM algorithm

Input: Data set X with number of elements (N) in

```
d dimensions, the number of labeled data in $X$: $L<N$ ; threshold $\varepsilon$; fuzzifier $m$; the number of clusters: ($C$); exponent $\alpha$ and MaxStep is the maximal number of iteration.
Output: Membership matrices u and cluster centers V.
BEGIN
1: Set $t = 0$
2: Initialize original cluster centers:
   $V_i^{(t)} \leftarrow random$; $i = 1, . . ., C$
//Repeat 3-7:
3: $t = t + 1$
4: Calculate $u_{ki}^{(t)}$ for labeled data ($k = 1, ..., L$; $i = 1, ..., C$) by (5).
5: Defuzzied $u_{ki}^{(t)}$ according to (6).
6: Calculate $V_i^{(t)}$ ($i = 1, ..., C$) using (4).
7: Check the stop condition: $\left\| V_i^{(t)} - V_i^{(t-1)} \right\| < \varepsilon$ or $t >$ MaxStep. If this condition is satisfied, the algorithm is stop. Otherwise, return 3.
END
```

2.2.2. Step 2 (Data transformation)

This is the transfer step between Step 1 and Step 3 (below). From the output of Step 1, we collect the cluster centers V of the labeled data. Unlabeled data points will use the result just obtained as the initial cluster center. Membership values of both unlabeled and labeled data will generate previous membership qualifications (Ū) for the method in next step. Thus, in our implementation, the mixture of the prior membership levels (Ū) and the labeled data is the predefined information of the semi-supervised fuzzy clustering.

2.2.3. Step 3 (Multiple point fuzzifiers for semi-supervised fuzzy clustering algorithm)

Based on the previous membership values (Ū), we set up the objective function of TSSFC for all data points of TSSFC as follows:

$$J_{\mathbf{TSSFC}} = \sum_{k=1}^{N}\sum_{i=1}^{C} u_{ki}^2 d_{ki}^2 + \lambda \sum_{k=1}^{N}\sum_{i=1}^{C} (u_{ki} - \overline{u_{ki}})^2 d_{ki}^2 \rightarrow \mathrm{Min} \quad (7)$$

with the constraints:

$$u_{ki} \in [0,1], i=1,...,C ; k=1,...,L \quad (8)$$

$$\sum_{i=1}^{C} u_{ki} = 1; k = 1,...,L \quad (9)$$

By using Lagrange and Gradient descent methods below, these problems will be solved:

$$V_i = \frac{\sum_{k=1}^{N}\left(u_{ki}^2 + \lambda\left(u_{ki} - \overline{u_{ki}}\right)^2\right) X_k}{\sum_{k=1}^{N}\left(u_{ki}^2 + \lambda\left(u_{ki} - \overline{u_{ki}}\right)^2\right)}; \quad i = 1,...,C \quad (10)$$

$$u_{ki} = \frac{1 + \lambda - \lambda \sum_{j=1}^{C} \overline{u_{kj}}}{(1+\lambda)\sum_{j=1}^{C}\left(\frac{d_{ki}}{d_{kj}}\right)^2} - \frac{\lambda \overline{u_{ki}}}{1+\lambda}; \quad k = 1,...,N,\ i = 1,...,C \quad (11)$$

The TSSFC algorithm is shown in Algorithm 2. In our implementation, the entire dataset with the initial membership (Ū) will be partitioned using the TSSFC model described in the 2nd block.

Algorithm 2. Semi-supervised fuzzy clustering algorithm

```
Input: Data set $X$ with number of elements (N) in $d$ dimensions, the number of labeled data in $X$: $L<N$ ; fuzzifier $m$; threshold $\varepsilon$; the number of clusters: ($C$); exponent $\alpha$ and MaxStep is the maximal number of iteration; the previous membership values for all data points ($\bar{U}$).
Output: Final membership matrices $u$ and cluster centers $V$
BEGIN
Step 1: Initialize the iteration: $t=0$
Step 2: Repeat the following steps 3-6:
Step 3: $t=t+1$
Step 4: Calculate $u_{ki}^{(t)}$ (i=1,...,C; k=1,...,N) by equation (11).
Step 5: Calculate $V_i^{(t)}$ (i=1,...,C) by equation (10).
Step 6: Check the stopping conditions: $\left\| V_i^{(t)} - V_i^{(t-1)} \right\| \le \varepsilon$ or $t >$ MaxStep. If satisfied then stop.
END
```

2.3. Remarks

- **Complexity Analysis**: There are two separate loops in the introduced method. In the first, the labeled data is partitioned using FCM, then its complexity is approx. $O\left(steps_1 \times LC^2\right)$, where $steps_1$ was the number of the first loop. In the last one, all datasets are clustered using **TSSFC** then the complexity is about $O\left(steps_2 \times NC^2\right)$. Obviously, L << N, then usually $steps_2 < steps_1$. The complexity of the introduced **TSSFC** method is $O\left(steps_2 \times NC^2\right)$ compared to $O\left(steps \times NLC^2\right)$ of CS3FCM. Therefore, **TSSFC** is better in terms of time calculation.

- Advantages of the TSSFC algorithm:

The introduced algorithm can be better in terms of computation time than other safe semi-supervised fuzzy clustering methods. For clustering, the algorithm performs two steps. The first step performs a labeled data partition to compute the initial membership of all the data in the second step. The last one is modified based on the semi-supervised FCM, then less complex than other algorithms when partitioning the whole data.

By eliminating or reducing the influence of data points labeled as suspicious, TSSFC can provide better clustering quality than other safe semi-supervised fuzzy clustering methods.

- Disadvantages of the TSSFC algorithm:

a) In the first step, the FCM algorithm may have to perform more iterations when performing the membership values reduction of the data labeled with doubt.

b) Given the diverse distribution of data points, it is difficult to accurately calculate the radius to determine the neighbors of labeled data. The first step becomes more complicated the larger the radius.

3. RESULTS

3.1. Environment setting

CS3FCM and **TSSFC** algorithms are implemented on Lenovo laptop with Core i7 processor, using DevC++ IDE.

The dataset is provided by Faculty of Water Resources Engineering - Thuy Loi University which is satellite image of the Cua Dai riverbank area, Quang Nam province, Vietnam. The original size of the satellite image is 7651 x 7811 pixels. The original format of satellite images was TIFF images. These images will be converted to PNG image format for further processing.

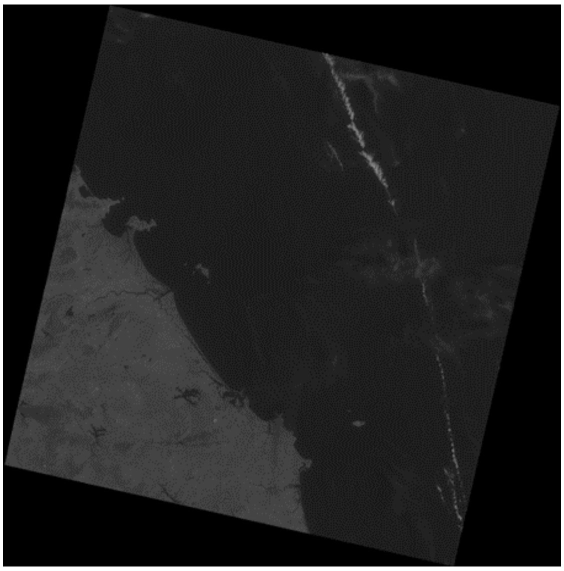


Figure 2. The origin satellite image

For processing convenience, we rotate the image along the vertical axis by an angle of 13 degrees. Then, we split the image obtained from the previous step into smaller images of size 201x201 pixcels using the *InterArea* interpolation supported in the OpenCV image processing library for the convenience of algorithm implementation. Some images after splitting as shown below.

From satellite images, we use *cvat.org* to locate pixels containing landslides. Landslide areas are areas where cracks appear in the soil surface. The landslide areas will be marked with different colors.

The number of attributes is reduced by converting the RGB to a grayscale image. Using a 3x3 sliding window to scan the surface of the image, the obtained results are used to synthesize the result of attributes in images. The properties are saved to a text file that will be used as input to the algorithm program. In the data file, each line contains 10 values: the first value is the attribute's label, if there are at least 5/9 pixels in the landslide area, the label is 1, otherwise the label is 2; The next 9 values are pixel values obtained from the 3x3 sliding window.

Based on the remaining attributes, our program runs 10 times for each image and initializes the label for 20% of random pixels, the other pixels are unlabeled. In labeled pixels, we run the experiments with the amount of incorrect label as 0%, 10%, 20%, 30%, respectively.

After the model execution is complete, the probability of the labels corresponding to each block 3x3 pixel will be saved. The label with the higher probability will be assigned to the block under consideration.

Criteria for evaluation are classification accuracy (*CA*) and computational time (*CT*). The CA (classification accuracy) [10] for the semi-supervised clustering methods was determined as follows,

$$CA=\frac{\sum_{k=1}^{n}\delta(y_k,map(\tilde{y}_k))}{n} \tag{12}$$

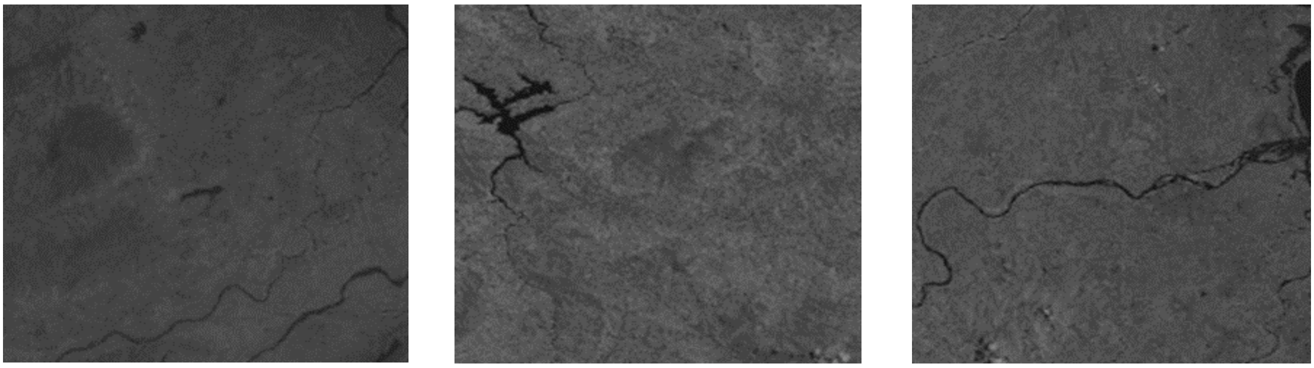
where the function $\delta(x,y)$ has value of 1 if $x=y$ and 0 if $x\neq y$. $map(y_k)$ is a function that maps y_k with equivalent labels using the Kuhn–Munkres algorithm [11]. The largest value indicates better performance for the CA metric. The unit is percent (%).

The *CT* is the amount of time it takes to perform a calculation in the equation below

$$CT = T_2 - T_1, \tag{13}$$

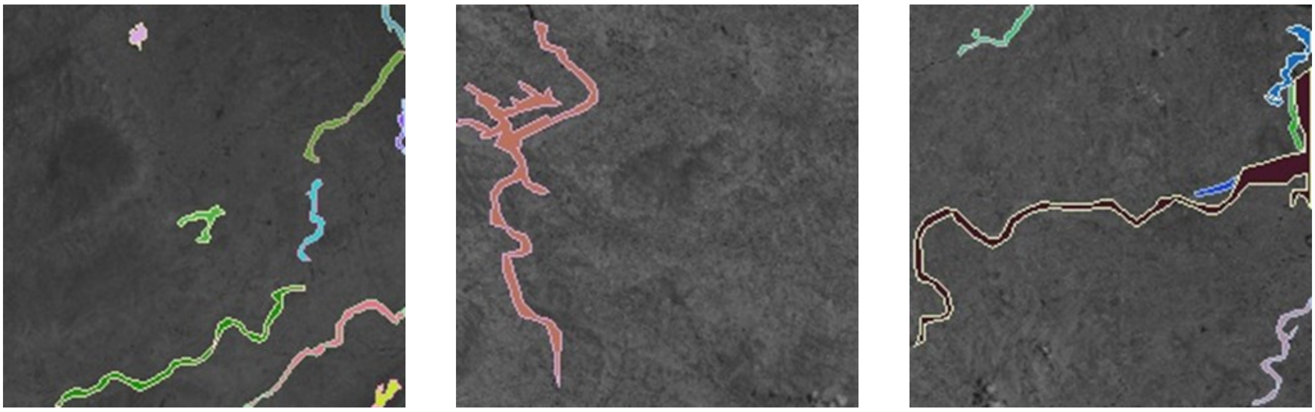
where *T1* is the start time and *T2* is the end time of the algorithm. Smaller values represent better performance for CT index. The unit of this quantity is second (s).

The introduced **TSSFC** method is experimentally compared with CS3FCM algorithm [8]. The validity indices in these implementations are classification accuracy, clustering quality and computational time.



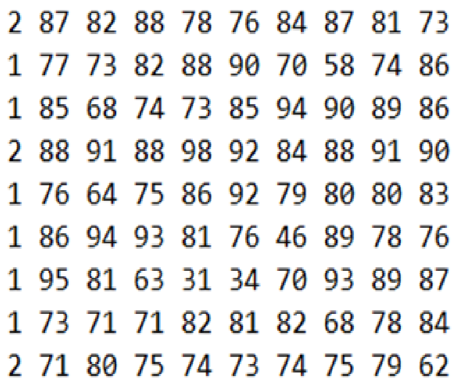
(a) (b) (c)

Figure 3. a) The first satellite image; b) The second satellite image; c) The third satellite image



(a) (b) (c)

Figure 4. a) The visual image id 1; b) The visual image id 2; c) The visual image id 3



```
2 87 82 88 78 76 84 87 81 73
1 77 73 82 88 90 70 58 74 86
1 85 68 74 73 85 94 90 89 86
2 88 91 88 98 92 84 88 91 90
1 76 64 75 86 92 79 80 80 83
1 86 94 93 81 76 46 89 78 76
1 95 81 63 31 34 70 93 89 87
1 73 71 71 82 81 82 68 78 84
2 71 80 75 74 73 74 75 79 62
```

Figure 5. Attribute data

3.2. Results

Using all the data elements in selected datasets, the classification accuracy, and calculation time of **TSSFC** and CS3FCM are calculated and showed in Table 1.

Table 1. The values of validity indices on satellite data (Bold values indicate the best ones in given dataset)

CRITERIA	Classiftcation accuracy		Computational time (s)	
METHOD	TSSFC	CS3FCM	TSSFC	CS3FCM
Image ID 1	0.79	0.61	11.91	15.4
Image ID 2	0.94	0.64	20.12	16.98
Image ID 3	0.88	0.82	19.55	26.87

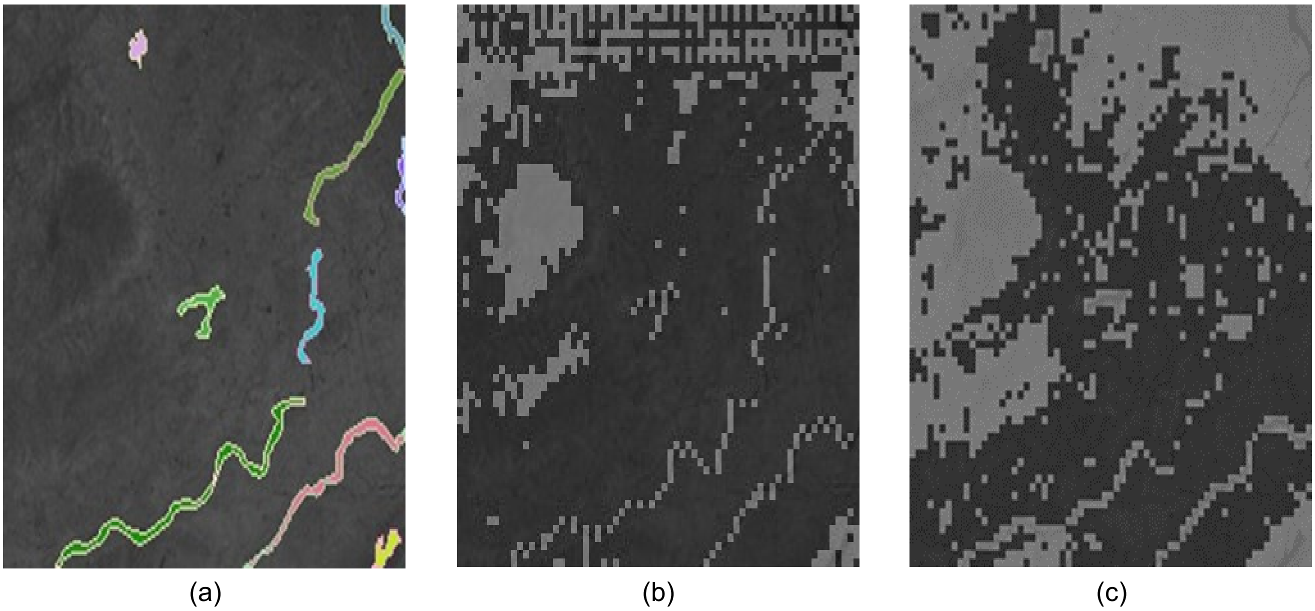
Comparing these algorithms on 6 datasets by different validity indices, we get:

i. Classification accuracy: From the data in Table 1, **TSSFC** obtained better results on all three satellite images.

ii. Computational time: As showed in Table 1, CS3FCM is better than TSSFC in time consuming on Image ID 2, while TSSFC achieves better results on the other 2 satellite images.

On overall, **TSSFC** gets better performance than CS3FCM in term of clustering accuracy. In run time, **TSSFC** takes a bit longer than CS3FCM in some cases. To illustrate the performance of **TSSFC** and CS3FCM visually, the results of running these algorithms on three satellite landslide images are given as in figures below.

In general, it can be seen that the normal land area and the landslide area through satellite images have certain similarities, especially in color, etc., so it is difficult to recognize. This can be clearly seen in the illustration: many landslide areas are mistakenly detected, especially the results obtained from CS3FCM. However, real landslide areas have been discovered, from which to promptly evaluate solutions to prevent and overcome the consequences of natural disasters.



(a) (b) (c)

Figure 6. Clustering results of image 1: a) The visual image; b) By applying TSSFC; c) By applying CS3FCM

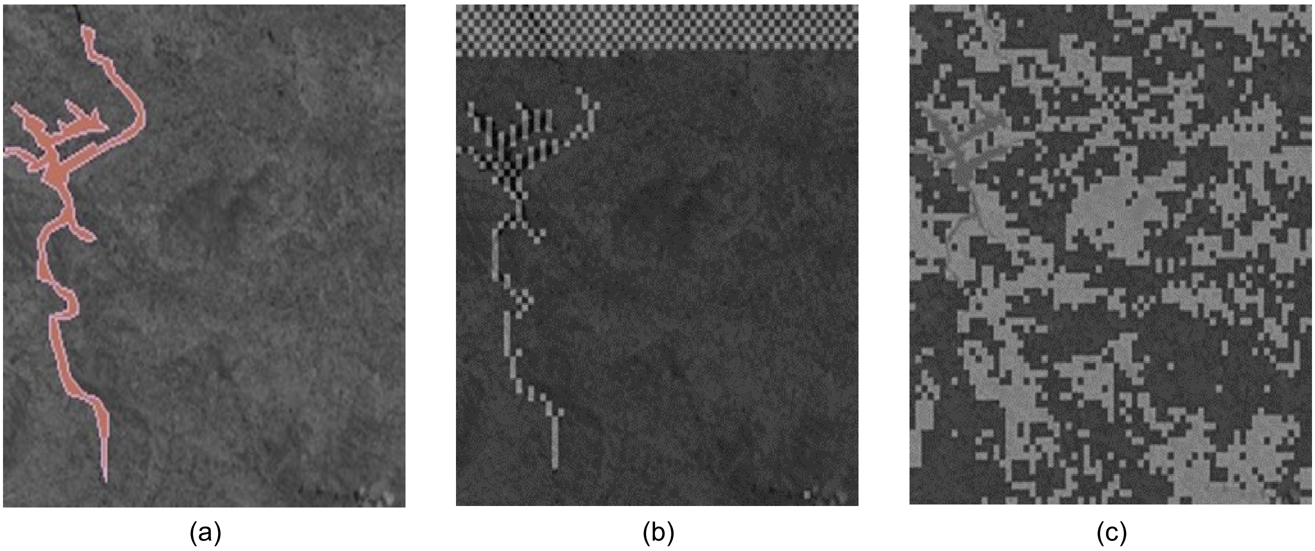


Figure 7. Clustering results of image 2: a) The visual image; b) By applying TSSFC; c) By applying CS3FCM

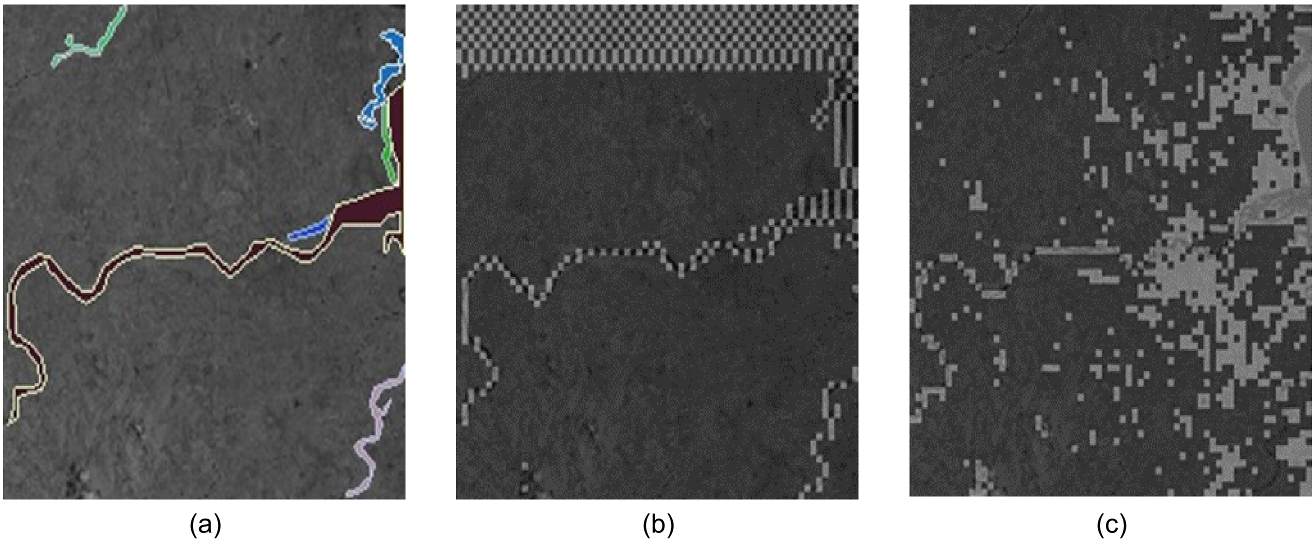


Figure 8. Clustering results of image 3: a) The visual image; b) By applying TSSFC; c) By applying CS3FCM

4. DISCUSSION

By using semi-supervised clustering and multiple fuzzifiers, in this paper, the introducted TSSFC algorithm has improved for data partitioning with confidence problems. The implementation steps of the algorithm are mentioned in detail in Section 2. The introducted method is compared with CS3FCM in the experiment part, the evaluation criteria given are: clustering accuracy and computation time.

Introduces a three steps model (TSSFC) in order to partition objects from a dataset with confidence and to deal with noise/outlier data. Introduces a modification of the FCM to evaluate the impact of labeled data using both unlabeled and labeled data. Present the process of using semi-supervised clustering method with different fuzzifier for each data element. Compare the obtained results of TSSFC and CS3FCM through experiment using valid indicators and visual images on satellite images.

With the results obtained from the introducted algorithm, it can be seen that the performance on clustering accuracy has been significantly improved. In addition, noise elements are removed or significantly reduced after applying the introducted algorithm.

5. CONCLUSIONS

In this paper, we have introduced a semi-supervised fuzzy clustering model called TSSFC and applied this algorithm to the landslide detection problem in satellite images. With the results obtained when testing with data images, it can be seen that the introduced algorithm can be applied to the problem of detecting landslide areas. This is very useful for individuals and organizations operating in the field of disaster prevention, shortening search and detection time. Thereby quickly making decisions and solutions to prevent possible damage to people and property.

Besides the shown advantages, the TSSFC algorithm still has some limitations. The first is the limitation of computation time, the reason for this is that the TSSFC algorithm needs a lot of parameters to execute. Secondly, there are still cases of false detection when checking with satellite images.

For future work, the calculation of parameters with many parameters is very complicated, so it is necessary to optimize the parameters for the calculation, thereby improving the performance in terms of time. Increasing image resolution, applying image processing algorithms to improve input image quality are also our goals. Today, deep learning models often have certain advantages in processing time and accuracy. Therefore, the combination of TSSFC algorithm with deep learning models is also a concern.

REFERENCES

[1] J.C. Bezdek. (1981). Pattern recognition with fuzzy objective function algorithms. *Kluwer Academic Publishers*.

[2] Salem Saleh Al-amri, N.V. Kalyankar, S.D. Khamitkar. Image segmentation by using thershod techniques. *Journal of Computing*, 2(5), 2010, 83-86.

[3] James C. Bezdek, Robert Ehrlich, William Full. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, 10(2-3), 191-203.

[4] W. Pedrycz and J. Waletzky. (1997). Fuzzy clustering with partial supervision. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 27(5), 787-795.

[5] S. Kundu, U. Maulik, and A. Mukhopadhyay. (2021). A game theory-based approach to fuzzy clustering for pixel classification in remote sensing imagery. *Soft Computing*, 25, 5121-5129.

[6] F. Salehi, M.R. Keyvanpour, and A. Sharifi. (2021). SMKFC-ER: Semi-supervised multiple kernel fuzzy clustering based on entropy and relative entropy. *Information Sciences*, 547, 667-688.

[7] J. Xiong, X. Liu, X. Zhu, H. Zhu, H. Li and Q. Zhang. (2020). Semi-supervised fuzzy c-means clustering optimized by simulated annealing and genetic algorithm for fault diagnosis of bearings. *IEEE Access*, 8, 181976-181987.

[8] H. Gan, Y. Fan, Z. Luo, R. Huang, and Z. Yang. (2019). Confidence-weighted safe semi-supervised clustering. *Engineering Applications of Artificial Intelligence*, 81, 107-116.

[9] G. Casalino, G. Castellano, C. Mencar. (2019). Data stream classification by dynamic incremental semi-supervised fuzzy clustering. *International Journal on Artificial Intelligence Tools,* 28(8), 1960009.

[10] H. Gan, Y. Fan, Z. Luo, R. Huang, and Z. Yang. (2019). Confidence-weighted safe semi-supervised clustering. *Engineering Applications of Artificial Intelligence*, 81, 107-116.

[11] L. Lov´asz, M.D. Plummer. (2009). Matching theory, Vol. 367. *Ams Chelsea Publishing*.