Decomposition-Based Multiobjective Particle Swarm Optimization for Change Detection in SAR Images

Tao Zhan, Zedong Tang, Maoguo Gong, Xiangming Jiang
Key Lab of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an Shaanxi, China
omegazhant@gmail.com,omegatangzd@gmail.com, gong@ieee.org,chiangshm@gmail.com

Jiao Shi
School of Electronics and Information, Northwestern Polytechnical University, Xi'an Shaanxi, China
jiaoshi@nwpu.edu.cn

ABSTRACT

Owing to the immunity to illumination and atmospheric conditions, synthetic aperture radar (SAR) images have been the main source of data for environmental monitoring. However, it is a challenging task for change detection because of the influence of speckle noise. In this paper, we propose an unsupervised multiobjective particle swarm optimization approach based on decomposition for change detection in SAR images. For the change detection task, it can be modeled as a multiobjective optimization problem (MOP), which consists of two contradictory objectives, namely, retaining image change details and removing the speckle noise. We optimize this MOP by using particle swarm optimization, which decomposes it into a set of subproblems by assigning different weights to these two objectives, thus obtaining the optimal trade-off between them. To accurately identify changed regions, the strategy of majority voting is employed to assemble partial good solutions to generate the final change detection result. The impressive experimental results on both real data sets have demonstrated the effectiveness and superiority of the proposed method.

CCS CONCEPTS

• **Theory of computation** → **Evolutionary algorithms**;

KEYWORDS

Multiobjective particle swarm optimization, decomposition, change detection, synthetic aperture radar images.

ACM Reference Format:
Tao Zhan, Zedong Tang, Maoguo Gong, Xiangming Jiang and Jiao Shi. 2018. Decomposition-Based Multiobjective Particle Swarm Optimization for Change Detection in SAR Images. In *GECCO '18 Companion: Genetic and Evolutionary Computation Conference Companion, July 15–19, 2018, Kyoto, Japan.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3205651.3208279

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07…$15.00
https://doi.org/10.1145/3205651.3208279

1 INTRODUCTION

For decades, the identification of changes occurring on the land surface has become increasingly important for clearly observing our planet. With the rapid development of remote sensing technology, change detection techniques [20] are usually used to detect changes present on the ground by using different types of satellite images, which can be easily obtained from various remote sensing platforms at present. Change detection is such a process that aims to analyze two (or more) images captured at different observation times, thus detecting any changes taking place over the same geographical area [27]. It has been widely used in many real applications, including disaster evaluation [4], land use/land cover transition [23] and environmental monitoring [2] to name a few. Among remote sensing images, synthetic aperture radar (SAR) images have attracted significant attention, because they are insensitive to illumination and atmospheric conditions (such as clouds, fog and haze). Owing to the inherent defect of the imaging mechanism of SAR sensors, however, SAR images are often contaminated by noise, which increases the difficulty in identifying meaningful changes.

In the literature, many methods have been developed for change detection with focus on SAR images, which can be divided into two categories: supervised and unsupervised. In supervised methods, the labeled dataset is often needed to train the classifier to distinguish changed regions. In [9], Dai *et al.* proposed a supervised change detection method based on artificial neural networks. Due to the limitation of actual conditions, generally, it is difficult to collect enough available training data. Furthermore, this process is often time-consuming and costly. In contrast, unsupervised methods are more popular, because they have the ability to detect changed regions without any prior information.

For unsupervised methods, they are often composed of the following three steps: 1) data preprocessing, including radiometric correction and image registration; 2) generating a difference image (DI), which is used to measure the changing degree between the considered images; and 3) detecting changes. Most of the existing change detection approaches are focused on the last two steps. In general, the DI is

generated by the log-ratio operator [5] because of the multiplicative nature of speckle noise. To identify changed regions, threshold and clustering methods are widely used, which segment the DI into two different classes: changed and unchanged. The main idea of threshold-based methods is to find an optimal threshold to distinguish changed pixels from the unchanged ones via establishing a suitable model, e.g., the generalized Kittler-Illingworth method (GKI) [1] and the expectation-maximization (EM) algorithm [22]. Among the classical clustering algorithms, the fuzzy c-mean (FCM) method [12] can reserve more image information. Nevertheless, this method is very sensitive to noise. To overcome this drawback, Krindis and Chatzis proposed a fuzzy local information C-means (FLICM) clustering algorithm [18], which uses the local spatial information of individual pixel and thus is robust to noise. Despite the good performance, it can not be guaranteed that the local minimizers of this method can converge to the true local minima of the objective function.

Actually, the methods described above are designed to detect the remarkable changes, ignoring the identification of weak changes. Therefore, optimization methods are needed to identify both strong and weak changes. In [6], Celik proposed an unsupervised change detection method for satellite images based on a genetic algorithm (GA). Recently, particle swarm optimization (PSO) [13, 26] is widely used to find the optimum solution in the whole search space. Due to its rapid speed of convergence and powerful ability of global optimization, many variants of PSO [14] have been developed to address different tasks and obtained superior performance. In fact, many real-world problems that involves simultaneous several optimization objectives are usually referred to as multiobjective optimization problems (MOPs), which can be solved based on PSO. For the SAR image change detection task, it also can be modeled as a MOP, which is composed of two conflicting objectives: preserving image change details and suppressing the speckle noise. In this paper, we present an unsupervised decomposition-based multiobjective PSO (DMPSO) for detecting changes occurring between multitemporal SAR images. The MOP is decomposed into a set of subproblems with different weight coefficients so as to achieve a better trade-off between these two objectives. In fact, these subproblems can be optimized simultaneously as a whole, thus is easy to obtain a set of solutions. To accurately detect changed regions, the strategy of majority voting is also used to improve the performance of the proposed method.

The rest of this paper is organized as follows. Section 2 introduces the relevant background knowledge of PSO and the problem of change detection as well as our motivation. Section 3 describes the proposed method in detail. The experimental results on both real data sets will be analyzed in Section 4. Finally, Section 5 draws the conclusion of this paper.

2 BACKGROUND

Here, we first introduce the relevant knowledge of PSO and MPSO briefly. Then, the problem of change detection as well as our motivation are described in detail.

2.1 PSO

The PSO algorithm is a population-based heuristic optimization method that aims to find the optimal solution using a population of particles [13], which is inspired by observing the social behaviors of bird flocking and fish schooling. PSO is characterized as simple to understand, easy to implement and fast convergence, which has been widely used in neural network training, feature selection and image segmentation/clustering [15, 19, 21, 28]. Among the given population (called *swarm*), each individual (called *particle*) updates its search behavior based on its own experience and other members' experiences. In PSO, each particle is associated with a spatial position $\boldsymbol{X}$ and a flying velocity $\boldsymbol{V}$ in the search space. The spatial position and flying velocity of each particle are updated for each generation until the optimal solution is found or a maximum number of iteration is reached.

Differently from evolutionary algorithms, PSO does not require crossover and mutation operations. In PSO, the inertia weight is used to balance the global optimization and local optimization, while acceleration coefficients aim to control the relative importance between the local best and the global best during the search process. The procedure for implementing PSO is as follows:

(1) Randomly initializing spatial positions and flying velocities of all the particles on a D-dimensional space, generation index $g = 1$;
(2) For each particle, calculate its own fitness value f_n^g;
(3) Compare particle's fitness value f_n^g with particle's f_{pbest}^g. If f_n^g is better than f_{pbest}^g, then set the value of f_{pbest}^g equal to f_n^g, and replace the position information of *pbest* by the current particle's position vector values;
(4) Compare fitness evaluation with the pervious best value f_{gbest} of the whole population. If current value is better than f_{gbest}, then set *gbest* to the current particle's position information;
(5) Update the spatial position and flying velocity of each particle by the following formula, which are given by

$$\begin{aligned} V_{nd}^{g+1} \leftarrow & w * V_{nd}^g + c_1 * r_1 * (pbest_{nd}^g - X_{nd}^g) \\ & + c_2 * r_2 * (gbest_d^g - X_{nd}^g) \end{aligned} \tag{1}$$

$$X_{nd}^{g+1} \leftarrow X_{nd}^g + V_{nd}^g \tag{2}$$

where $\boldsymbol{X}_n = (X_{n1}, X_{n2}, \ldots, X_{nD})$ is the spatial position of the nth particle, $\boldsymbol{V}_n = (V_{n1}, V_{n2}, \ldots, V_{nD})$ denotes its corresponding flying velocity. Note that the velocity of each particle is limited to [-V_{max}, V_{max}], where V_{max} indicates its maximum velocity. w represents the inertia weight, whereas c_1 and c_2 are acceleration coefficients. $\boldsymbol{pbest}_n = (pbest_{n1}, pbest_{n2}, \ldots, pbest_{nD})$ represents the best position of the nth particle. $\boldsymbol{gbest} = (gbest_1, gbest_2, \ldots, gbest_D)$ is the global best position among the overall population. r_1 and r_2 are random numbers between 0 and 1.
(6) set $g = g + 1$ and loop to step (2) until a termination condition is reached.

2.2 MPSO

In fact, many real-world problems can be modeled as MOPs, which are usually needed to optimize several objectives simultaneously. Despite these advantages, there are two main issues to be addresses when applying PSO to MOPs. The first issue is how to select the personal best and the global best from the search space, because there is no absolute best solution, but rather a set of non-dominated solutions. Several effective methods have been proposed to solve this problem, such as the crowding distance [11] technique, the dynamic neighborhood [16] strategy and the decomposition [29] approach. Another issue in MPSO is how to remove the individual from the archive so as to retain the distributivity and diversity of solutions. Generally, external elitist archive is used to store the non-dominated solutions achieved by an algorithm and then they are filtered based on a certain measure index, e.g., density. Due to the fact that the number of non-dominated solutions grows very fast that increases the computational cost of updating the archive, thus, the size of archive is unconstrained. In addition, it is important to prune and update the archive, because the physical memory is often limited. There are many approaches designed to resolve this intractable problem, e.g., the kernel density [10] and the clustering strategy [17], which can often obtain good performance.

In the past few years, many MPSOs have been proposed. They can be mainly divided into several categories as follows: 1) Aggregating approaches [24], the basic idea of them is to transform this MOP into a single objective for solving; 2) Lexicographic ordering methods [16], which aim to optimize this MOP based on the order with respect to a certain criterion; 3) Sub-population approaches [8], such methods try to partition this MOP into several sub-populations, and each population is modeled as a single optimizer. Then, generating the optimal solution by exchanging or recombining the information between different populations; and 4) Pareto-based methods [7], they aim at selecting as leaders to the particles that are non-dominated with respect to the swarm.

2.3 Problem Statements and Motivation

Given two coregistered multi-temporal SAR images, $\mathbf{I}_1 = \{I_1(x,y)|1 \le x \le H, 1 \le y \le W\}$ and $\mathbf{I}_2 = \{I_2(x,y)|1 \le x \le H, 1 \le y \le W\}$, where H and W are the height and width of the considered images, respectively. They are acquired over the same geographical area at different observation times t_1 and t_2, respectively. The goal of change detection is to accurately detect meaningful changes that have occurred between them. The basic flowchart of change detection is shown in Fig. 1.

However, due to the inherent defect of the SAR sensor imaging mechanism, the obtained images are often contaminated by serious noise, which further increases the difficulty in identifying real changes. For most of traditional change detection methods, image denoising is deemed as an indispensable step in the process of data preprocessing. Despite noise removing, some useful information is also lost, which may result in bad performance. In fact, it is difficult to preserve image details and remove the speckle noise simultaneously. To address this intractable task, we attempt to model the change detection problem as a MOP, which can be decomposed into a set of subproblems for solving based on PSO, thus obtaining superior performance.

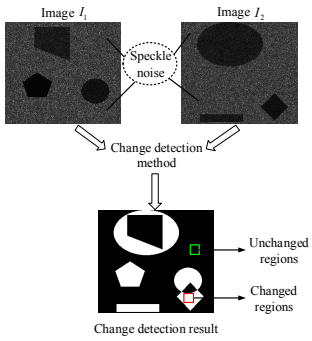

Figure 1: The basic flowchart of change detection.

3 METHODOLOGY

3.1 Multiobjective Clustering for Change Detection

Let consider two coregistered SAR images $\mathbf{I}_1$ and $\mathbf{I}_2$, which are of the same size $H \times W$. In order to measure the changing degree between them, the log-ratio operator is used to generate an initial difference image $\mathbf{I}$, which is computed as

$$\mathbf{I}(x,y) = |\log \mathbf{I}_2(x,y) - \log \mathbf{I}_1(x,y)| \tag{3}$$

From the view of retaining more image details, the cost function of FCM clustering is selected as the first objective function for the MOP, which is defined by

$$f_1(v_1, v_2) = \sum_{i=1}^{N}\sum_{j=1}^{c} u_{ji}^m \|x_i - v_j\|^2 \tag{4}$$

where N represents the total number of pixels in the image $\mathbf{I}$, and c is the number of clustering centers. Here, it is equal to 2, because we aim to classify the difference image $\mathbf{I}$ into two classes: changed and unchanged. m is the fuzzy factor, and u_{ji} indicates the fuzzy membership degree of the ith pixel associated with the cluster j, which satisfies the following conditions

$$\begin{aligned} &\sum_{j=1}^{c} u_{ji} = 1, \forall i = 1, 2, \ldots, N \\ &\text{s.t.} \quad 0 \le u_{ji} \le 1 \end{aligned} \tag{5}$$

Despite the preservation of image details, the noisy spots are also retained. To alleviate the influence of speckle noise, an average filtering operation is applied to the image $\mathbf{I}$, and the generated denoised image $\bar{\mathbf{I}}$ can be obtained by

$$\bar{x}_i = \frac{1}{S}\sum_{i \in N_i} x_i \tag{6}$$

where $\bar{x}_i$ represents the gray value of the ith pixel in the image $\bar{\mathbf{I}}$. N_i is the neighborhood of the ith pixel (for example, 3×3) of image $\mathbf{I}$, whereas S denotes its total number of pixels.

For the purpose of removing the speckle noise, we use the similar clustering strategy to analyze the denoised image $\bar{\mathbf{I}}$, and the second objective function is defined as

$$f_2(v_1, v_2) = \sum_{j=1}^{c} \sum_{r \in N_i} u_{jr}^m \|\bar{x}_r - v_j\|^2 \tag{7}$$

Through the above analysis, we can construct a MOP by combining these two contradictory objective functions, which is shown as follows:

$$\begin{aligned} &\min F(v_1, v_2) = (f_1, f_2)^{\mathrm{T}} \\ &\text{s.t.} \quad (v_1, v_2)^{\mathrm{T}} \in \Omega \end{aligned} \tag{8}$$

where $(v_1, v_2)^{\mathrm{T}}$ is a decision vector that is composed of two clustering centers v_1 and v_2, and Ω is the corresponding decision space. As described previously, we know that FCM aims at classifying an image into two classes by minimizing its cost function. Thus, the MOP above can be solved as a minimization problem. In order to obtain different trade-offs between these two contradictory objectives, we decompose this problem into a series of subproblems, which is defined as

$$\begin{aligned} &\min h(v|\boldsymbol{\alpha}) = \alpha_1 f_1 + \alpha_2 f_2 \\ &\text{s.t.} \quad \alpha_1 + \alpha_2 = 1, \alpha_1 \geq 0, \alpha_2 \geq 0 \end{aligned} \tag{9}$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)^{\mathrm{T}}$ is a weight vector, which is used to control the balance between the preservation of image details and the restriction to the speckle noise. The fuzzy membership degree u_{ji} and the clustering center v_j can be calculated as

$$u_{ji} = \frac{\left(\|x_i - v_j\|^2 + \frac{\alpha_2}{\alpha_1}\|\bar{x}_i - v_j\|^2\right)^{1/(m-1)}}{\sum_{k=1}^{c}\left(\|x_i - v_k\|^2 + \frac{\alpha_2}{\alpha_1}\|\bar{x}_i - v_k\|^2\right)^{1/(m-1)}} \tag{10}$$

$$v_j = \frac{\sum_{i=1}^{N} u_{ji}^m (x_i + \bar{x}_i)}{(1 + \frac{\alpha_2}{\alpha_1}) \sum_{i=1}^{N} u_{ji}^m} \tag{11}$$

Generally, the complexity of this MOP is associated with the number of decision variable. For the SAR image change detection task, its goal is to discriminate changed regions from the unchanged ones. Thus, the number of decision variable is equal to 2, which makes the complexity of the proposed method very low. Note that the objective space of this MOP is decomposed into a set of sub-regions according to a set of direction vector with respect to the reference point, then the mutation operator is used to improve the convergence of the generated non-dominated solution set. In addition, the neighborhood strategy is also employed to update the global best individual for each run and thus maintaining the diversity of solutions. In DMPSO, $\boldsymbol{P}$ and $\boldsymbol{G}$ are the personal best and global best individual, respectively. The concrete details of the proposed method for change detection is concluded in **Algorithm 1**.

Algorithm 1 The proposed DMPSO for change detection.

Input: N_p: the number of population size; T: neighborhood size; G_{max}: the maximum number of generations; w: the inertia weight; c_1, c_2: acceleration constants; m: the fuzzy factor; $\mathbf{I}$: the difference image; $\bar{\mathbf{I}}$: the denoised difference image.
Ouput: Pareto solution set.
1: Randomly initializing the position and velocity of each particle in the problem space.
2: Randomly initializing personal and global positions.
3: Calculate the fuzzy membership vectors $u_1, u_2, ..., u_{N_p}$ by Eq.(10) and calculate $f_1, f_2, h(\boldsymbol{X}_n|\boldsymbol{\alpha})$.
4: Generate a identically distributed of N_p weight vectors $\boldsymbol{\alpha_n}, n = 1, 2, ..., N_p$ and find T closest neighbors for each vector, $B(n) = \{n_1, ..., n_T\}$.
5: Set $g = 1$.
6: **repeat**
7: **for** $n = 1, 2, ..., N_p$ **do**
8: Update the flying velocity and spatial position of the nth particle by Eq.(1) and Eq.(2).
9: Mutate $\boldsymbol{X}_n$ by polynomial mutation.
10: Update the fuzzy membership vector u_n corresponding to $\boldsymbol{X}_n$ by Eq.(10).
11: Calculate $h(\boldsymbol{X}_n|\boldsymbol{\alpha})$. If $h(\boldsymbol{X}_n|\boldsymbol{\alpha}_n) < h(\boldsymbol{P}_n|\boldsymbol{\alpha}_n)$, then set $\boldsymbol{P}_n = \boldsymbol{X}_n$ and $h(\boldsymbol{P}_n|\boldsymbol{\alpha}_n) = h(\boldsymbol{X}_n|\boldsymbol{\alpha}_n)$.
12: **end for**
13: **for** $n = 1, 2, ..., N_p$ **do**
14: Q is the random permutation of $B(n)$.
15: **for each** q in Q **do**
16: **if** $h(\boldsymbol{X}_n|\boldsymbol{\alpha}_q) < h(\boldsymbol{G}_q|\boldsymbol{\alpha}_q)$ **then**
17: set $\boldsymbol{G}_q = \boldsymbol{X}_n$.
18: set $h(\boldsymbol{G}_q|\boldsymbol{\alpha}_q) = h(\boldsymbol{X}_n|\boldsymbol{\alpha}_n)$.
19: set $\mu_q = \mu_n$.
20: **end if**
21: **end for**
22: **end for**
23: $g++$.
24: **until** $g > G_{max}$.

3.2 Ensemble-based Strategy for Detecting Changes

By optimizing these subproblems, a set of solutions can be generated, which correspond to different trade-offs between the preservation of image details and the capability of removing the speckle noise. Here, note that each solution of this set represents a single change detection result. However, how to select an excellent solution from the set is still a challenging problem. In [30], Zhou *et al.* proposed the theory of "many could be better than all" that verifies the effectiveness of partial ensemble strategy. Thereby, an ensemble strategy based on majority voting is employed to eliminate some spurious and meaningless changes, thus identifying changed regions more accurately.

As shown in Fig. 2, a non-dominated Pareto front can be obtained by optimizing the MOP. It should be noted that a smaller f_1 denotes a better ability of retaining image details, whereas a smaller f_2 stands for a better performance of removing the speckle noise. However, it is impossible to minimize these two objective functions simultaneously. Thereby,

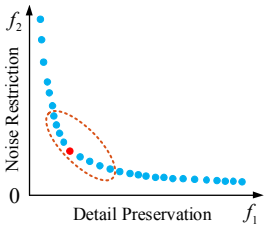

Figure 2: A schematic diagram for choosing candidate solutions based on the non-dominated Pareto front.

we should achieve a balance between them. In Fig. 2, we use the approach in [3] to find the knee point from the Pareto front, which is marked in red color. Then, we consider its neighborhood framing with ellipse in orange color as the candidate solution set, which are the relatively good solutions for the change detection task. For the pixel at position (x, y), its class label can be determined by

$$L(x,y) = \begin{cases} 1, & N_s/N_c > 0.5 \\ 0, & else \end{cases} \tag{12}$$

where N_s denotes the number of same class labels (i.e., the changed class) of the candidate solution set at the same position, N_c is the number of candidate solutions. By implementing this process on all the pixels, the final change detection result can be obtained.

4 EXPERIMENTAL STUDY

In this section, the datasets used in the experiments and the corresponding evaluation criteria are described briefly. To verify the effectiveness of the proposed method, several traditional methods including two threshold-based methods (i.e., GKI [1] and EM [22]) and two clustering approaches (i.e., FCM [12] and FLICM [18]) as well as a population-based method (i.e., GA [6]) are used as comparison algorithms in the experiments.

4.1 Datasets

The first dataset is composed of two SAR images, which were obtained by RADARSAT satellite in May 1997 and August 1997, respectively, as shown in Fig. 3(a) and (b). Both of them share the same size of 290×350 pixels. This dataset records the changes caused by the flood over the region of Ottawa City, Canada. Fig. 3(c) is the reference image, which is obtained via manual marking by the combination of specialistic knowledge and surface prior information.

The second dataset is the Red River dataset, which records the changes occurred on the basin of Red River, Vietnam. This dataset consists of two SAR images of the same size 512×512 pixels, which were acquired by ESAERS-2 satellite on August 24, 1996 and August 14, 1999, respectively, as shown in Fig. 4(a) and (b). The reference image is generated by integrating surface prior information with photo interpretation, as shown in Fig. 4(c).

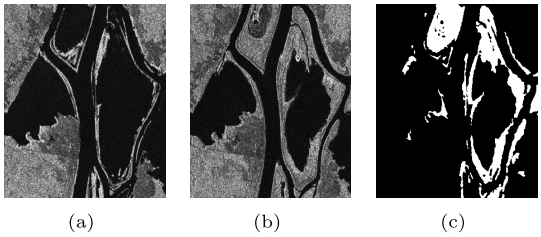

Figure 3: Ottawa dataset. (a) Image captured in May 1997. (b) Image captured in August 1997. (c) Reference image.

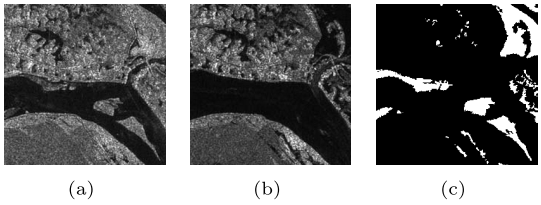

Figure 4: Red River dataset. (a) Image captured on August 24, 1996. (b) Image captured on August 14, 1999. (c) Reference image.

4.2 Evaluation Criteria

To quantitatively evaluate the performance of the proposed method, some evaluation criteria are used in our experiments, including *false alarms* (FAs), *missed alarms* (MAs), *overall errors* (OEs), *overall accuracy* (OA) and *Kappa coefficient* (KC). FA and MA represent the number of those pixels that are misclassified as the changed class and unchanged class, respectively. Then, OE and OA can be computed by

$$\text{OE} = \text{FA} + \text{MA} \tag{13}$$

$$\text{OA} = \frac{N - \text{OE}}{N} \tag{14}$$

where N denotes the total number of pixels in the change map. In addition, KC is usually to measure the performance of classification [25]. The larger its value is, the better the classification result is, which is calculated as

$$\text{KC} = \frac{\text{OA} - \text{PRE}}{1 - \text{PRE}} \tag{15}$$

$$\text{PRE} = \frac{(\text{TP} + \text{FA}) \cdot \text{RC} + (\text{TN} + \text{MA}) \cdot \text{RU}}{N^2} \tag{16}$$

where TP and TN are the number of truly changed and unchanged pixels comparing with the reference image, respectively. RC and RU denote the number of changed pixels and unchanged pixels in the reference image.

Table 1 lists the values of some parameters in the proposed method, which will be used in our experiments to validate its effectiveness based on different data sets. Note that all algorithms are executed on a single Intel Core i5-4590 CPU at 3.30Hz with 8G RAM memory and the evaluation criteria are the average values of the change detection result by running them for 30 times independently.

Table 1: The parameters used in the proposed DMPSO

Parameter	Description	Value
N_p	Population size	100
G_{max}	The maximum number of generation	200
w	Inertia weight	0.4
c_1, c_2	Acceleration coefficient	1.49
m	The fuzzy factor	2
N_c	The number of candidate solutions	9

4.3 Results on the Ottawa Dataset

The first experiment is implemented on the Ottawa dataset, which records the changes caused by the flood. By optimizing the MOP based on the proposed method, a uniformly-distributed Pareto front (see Fig. 5) can be obtained. This Pareto front is composed of one hundred solutions, and each one represents a single change detection result. In order to intuitively illustrate the results acquired by the proposed method, we select six different solutions from the front to display, as shown in Fig. 6. From the view of visual analysis, we can find that the results obtained by solutions S_1 and S_2 can preserve image details well, but lots of noisy spots are also retained, as shown in Fig. 6(a) and (b). At this point, the values of f_1 are small, while the values of f_2 are large, which indicates these solutions have powerful capacity of preserving image details but with weak robustness to noise. In contrast, solutions S_5 and S_6 have strong restriction to noise but with poor ability of retaining image details, which can be observed in Fig. 6(e) and (f). On the whole, the results generated by solutions S_3 and S_4 are better than others, as shown in Fig. 6(c) and (d). They can keep balance between the preservation of image details and the robustness to the speckle noise, though f_1 and f_2 are not the minimum among all solutions.

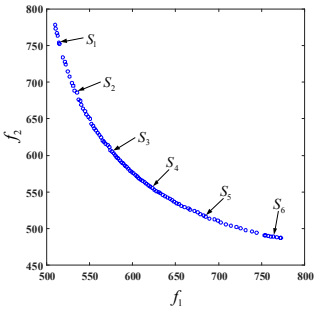

Figure 5: The Pareto front of the Ottawa dataset.

In order to further validate the performance of the proposed method, several representative change detection methods are used as comparison algorithms. The change detection results produced by different methods on the Ottawa dataset are shown in Fig. 7. In Fig. 7(a) and (b), it can be observed that the results obtained by GKI and EM are full of noisy spots, especially the latter, which demonstrate that classical threshold techniques are difficult to find an optimal threshold

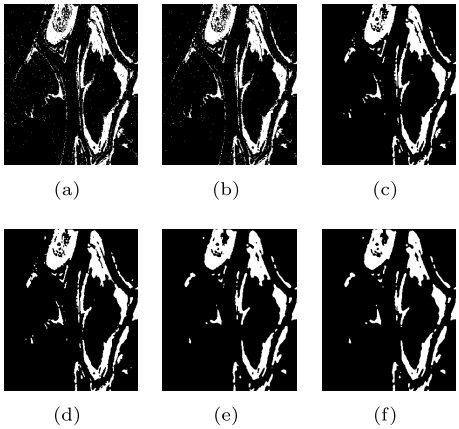

Figure 6: Change detection results acquired by partial solutions on the Ottawa dataset. (a)-(f) are the change detection results obtained by the solution S_1, S_2, S_3, S_4, S_5 and S_6, respectively.

to accurately identify changed regions. As shown in Fig. 7(c), the result generated by FCM is not good, because it is very sensitive to noise. In contrast, better result can be obtained by FLICM, due to the use of spatial information, as shown in Fig. 7(d). The result obtained by GA is shown in Fig. 7(e), we can see that many unchanged regions are falsely detected as changed and some details are lost. Furthermore, it always needs very high computational cost to perform the change detection task.

Among all the change detection results, it is clear that the proposed method can accurately detect changes with less noisy points. Table 2 reports the corresponding evaluation criteria obtained by different methods, we can find that the performance of the proposed method outperforms other approaches, wherein its values of OA and KC are up to 98.19% and 0.9326, respectively. Furthermore, it requires only 135.67s (less than 3 minutes) in total to complete the change detection task, which demonstrates its computational cost is very low.

Table 2: Quantitative evaluation of experimental results obtained by different methods on the Ottawa dataset

Method	FA	MA	OE	OA(%)	KC
GKI	4063	2087	6150	93.94	0.7833
EM	7139	1552	8691	91.44	0.7182
FCM	1945	2747	4692	95.38	0.8228
FLICM	**230**	2615	2845	97.20	0.8880
GA	4653	**351**	4914	95.16	0.8357
DMPSO	1065	770	**1835**	**98.19**	**0.9326**

4.4 Results on the Red River Dataset

Fig. 8 shows the Pareto front of the Red River dataset obtained by the proposed method, which is approximately evenly-distributed. This front consists of one hundred solutions that correspond to different trade-offs between the

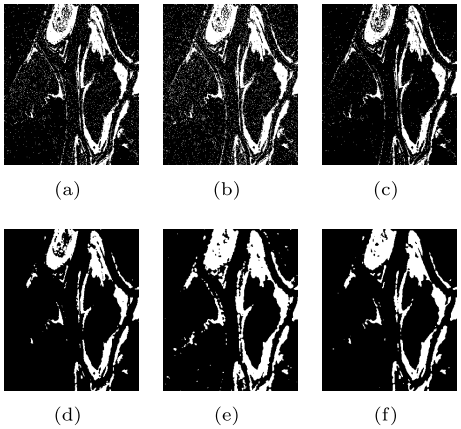

Figure 7: Change detection results obtained by different methods on the Ottawa dataset. (a) GKI. (b) EM. (c) FCM. (d) FLICM. (e) GA. (f) DMPSO.

preservation of image details and the robustness to noise. The change detection results obtained by six different solutions are illustrated in Fig. 9. Intuitively, we can see that the results generated by solutions S_3 and S_4 are better than other four solutions, major and subtle changes present on the ground can be well highlighted. They can balance these two contracting objectives well, hence, the strong capacities of retaining image details and removing the speckle noise can be acquired simultaneously, which demonstrates the effectiveness of the designed objective functions. Compared to traditional change detection methods, the proposed methods can give more insights for addressing the change detection task, which is benefit to accurately identify changed regions.

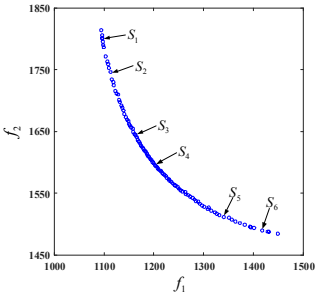

Figure 8: The Pareto front of the Red River dataset.

The corresponding experimental results and evaluation criteria on the Red River dataset are shown in Fig. 10 and listed in Table 3. For threshold-based methods (i.e., GKI and EM), generally, the generated change detection results are filled with many noisy spots, as shown in Fig. 10(a) and (b). Especially the latter, it suffers from the highest OE value (i.e., 18533) and thus leading to the worst performance. Compared with them, clustering-based methods can often obtain good performance. The results generated by FCM and FLICM are shown in Fig. 10(c) and (d). It can be seen that

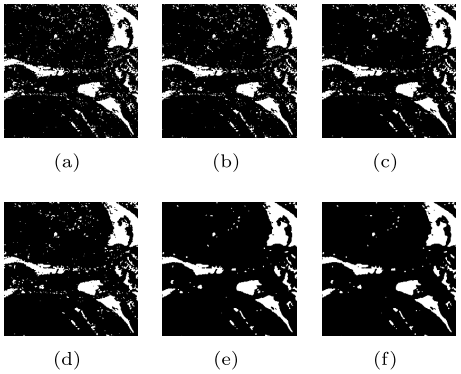

Figure 9: Change detection results acquired by partial solutions on the Red River dataset. (a)-(f) are the change detection results obtained by the solution S_1, S_2, S_3, S_4, S_5 and S_6, respectively.

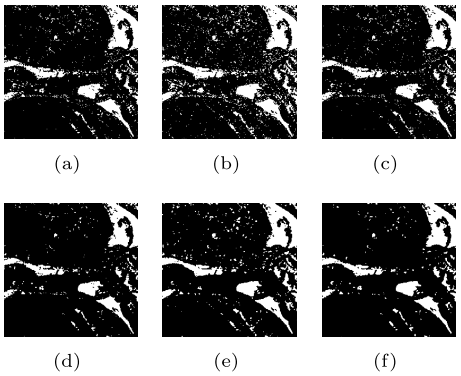

Figure 10: Change detection results obtained by different methods on the Red River dataset. (a) GKI. (b) EM. (c) FCM. (d) FLICM. (e) GA. (f) DMPSO.

the latter obtains better result because of the use of local spatial relationship of individual pixel, thus is robust to noise, though some change details are also missed. Fig. 10(e) shows the change detection result of GA, it can be observed that there are some noisy points and spurious changes. In fact, it is a single objective method that does not make full use of image information and the context information of individual pixel, which is sensitive to noise and thus leading to bad performance with high computational cost.

In fact, the change detection task can be modeled as a MOP, due to the importance of preserving image details and suppressing the speckle noise. Compared with these comparison algorithms, the proposed method can accurately discriminate changed regions from the unchanged ones with less noisy points, as shown in Fig. 10(f). From the perspective of quantitative analysis, we can find that the highest OA value (i.e., 97.87%) and KC value (i.e., 0.9039) can be obtained by the proposed method, as listed in Table 3. For the computational cost, the larger the data sets is, the more

running time it takes. For the Red River dataset, the proposed method spends 294.56s (less than 5 minutes) obtaining the final change detection result.

Table 3: Quantitative evaluation of experimental results obtained by different methods on the Red River dataset

Method	FA	MA	OE	OA(%)	KC
GKI	6935	2866	9801	96.26	0.8386
EM	17643	**890**	18533	92.93	0.7361
FCM	6514	2713	9227	96.48	0.8475
FLICM	**2929**	4164	7093	97.29	0.8750
GA	7331	1575	8906	96.60	0.8563
DMPSO	3039	2540	**5579**	**97.87**	**0.9039**

5 CONCLUSION

In this paper, we propose an unsupervised decomposition-based multiobjective particle swarm optimization method for change detection with focus on SAR images. We transform the change detection task into a MOP that consists of two conflicting objectives, namely, preserving image change details and restricting the speckle noise. We optimize this MOP by using PSO, which decomposes it into a set of subproblems with different weights so as to obtain the optimal trade-off between these two objectives. In addition, the strategy based on majority voting is employed to improve the performance of the proposed method. The impressive results over two different datasets demonstrate the effectiveness and superiority of the proposed method.

In the future, we will focus on addressing the change detection task based on the idea of multiobjective optimization, especially when the two images are acquired from different satellite sensors or not registered well.

REFERENCES

[1] Yakoub Bazi, Lorenzo Bruzzone, and Farid Melgani. 2005. An unsupervised approach based on the generalized Gaussian model to automatic change detection in multitemporal SAR images. *IEEE Trans. Geosci. Remote Sens.* 43, 4 (2005), 874–887.

[2] Mourad Bouziani, Kalifa Goïta, and Dong-Chen He. 2010. Automatic change detection of buildings in urban environment from very high spatial resolution images using existing geodatabase and prior knowledge. *ISPRS J. Photogramm. Remote Sens.* 65, 1 (2010), 143–153.

[3] Jürgen Branke, Kalyanmoy Deb, Henning Dierolf, and Matthias Osswald. 2004. Finding knees in multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*. Birmingham, UK, Springer, 722–731.

[4] Dominik Brunner, Guido Lemoine, and Lorenzo Bruzzone. 2010. Earthquake damage assessment of buildings using VHR optical and SAR imagery. *IEEE Trans. Geosci. Remote Sens.* 48, 5 (2010), 2403–2420.

[5] Florentin Bujor, Emmanuel Trouvé, Lionel Valet, J-M Nicolas, and J-P Rudant. 2004. Application of log-cumulants to the detection of spatiotemporal discontinuities in multitemporal SAR images. *IEEE Trans. Geosci. Remote Sens.* 42, 10 (2004), 2073–2084.

[6] Turgay Celik. 2010. Change detection in satellite images using a genetic algorithm approach. *IEEE Geosci. Remote Sens. Lett.* 7, 2 (2010), 386–390.

[7] Somayyeh Chamaani, Seyed Abdullah Mirtaheri, Mohammad Teshnehlab, and Mahdi Aliyari Shooredeli. 2007. Modified multi-objective particle swarm optimization for electromagnetic absorber design. In *Proc. IEEE Asia-Pacific Conf. Applied Electromagnetics (APACE 2007)*. Melaka, Malaysia, IEEE, 1–5.

[8] Chi-kin Chow and Hung-tat Tsui. 2004. Autonomous agent response learning by a multi-species particle swarm optimization. In *Proc. IEEE Congr. Evolutionary Computation (CEC 2004)*, Vol. 1. Portland, Oregon, USA, IEEE, 778–785.

[9] XL Dai and Slamak Khorram. 1999. Remotely sensed change detection based on artificial neural networks. *Photogramm. Eng. Remote Sens.* 65 (1999), 1187–1194.

[10] Kalyanmoy Deb. 1989. An investigation of niche and species formation in genetic function optimization. In *Proc. 3rd Int. Conf. Genetic Algorithms*. Fairfax, Virginia, USA, 42–50.

[11] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* 6, 2 (2002), 182–197.

[12] Joseph C Dunn. 1974. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *J. Cyberrn.* 3 (1974), 32–57.

[13] Russell Eberhart and James Kennedy. 1995. A new optimizer using particle swarm theory. In *Proc. 6th Int. Symp. Micromachine Human Sci.* Nagoya, Japan, IEEE, 39–43.

[14] Ahmed AA Esmin, Rodrigo A Coelho, and Stan Matwin. 2015. A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data. *Artif. Intell. Rev.* 44, 1 (2015), 23–45.

[15] Pedram Ghamisi, Micael S Couceiro, Fernando ML Martins, and Jon Atli Benediktsson. 2014. Multilevel image segmentation based on fractional-order Darwinian particle swarm optimization. *IEEE Trans. Geosci. Remote Sens.* 52, 5 (2014), 2382–2394.

[16] Xiaohui Hu and Russell Eberhart. 2002. Multiobjective optimization using dynamic neighborhood particle swarm optimization. In *Proc. IEEE Congr. Evolutionary Computation (CEC 2002)*, Vol. 2. Hawaii, USA, IEEE, 1677–1681.

[17] Joshua D Knowles and David W Corne. 2000. Approximating the nondominated front using the Pareto archived evolution strategy. *Evolut. Comput.* 8, 2 (2000), 149–172.

[18] Stelios Krinidis and Vassilios Chatzis. 2010. A robust fuzzy local information C-means clustering algorithm. *IEEE Trans. Image Process.* 19, 5 (2010), 1328–1337.

[19] Xiaolei Liang, Wenfeng Li, Yu Zhang, and MengChu Zhou. 2015. An adaptive particle swarm optimization method based on clustering. *Soft Comput.* 19, 2 (2015), 431–448.

[20] Dengsheng Lu, P Mausel, E Brondizio, and Emilio Moran. 2004. Change detection techniques. *Int. J. Remote Sens.* 25, 12 (2004), 2365–2401.

[21] SeyedAli Mirjalili, Siti Zaiton Mohd Hashim, and Hossein Moradian Sardroudi. 2012. Training feedforward neural networks using hybrid particle swarm optimization and gravitational search algorithm. *Appl. Math. Comput.* 218, 22 (2012), 11125–11137.

[22] Gabriele Moser and Sebastiano B Serpico. 2006. Generalized minimum-error thresholding for unsupervised change detection from SAR amplitude imagery. *IEEE Trans. Geosci. Remote Sens.* 44, 10 (2006), 2972–2982.

[23] W Muttitanon and NK Tripathi. 2005. Land use/land cover changes in the coastal zone of Ban Don Bay, Thailand using Landsat 5 TM data. *Int. J. Remote Sens.* 26, 11 (2005), 2311–2323.

[24] Konstantinos E Parsopoulos and Michael N Vrahatis. 2002. Particle swarm optimization method in multiobjective problems. In *Proc. ACM Symp. Applied Computing (SAC 2002)*. Madrid, Spain, ACM, 603–607.

[25] George H Rosenfield and Katherine Fitzpatrick-Lins. 1986. A coefficient of agreement as a measure of thematic classification accuracy. *Photogramm. Eng. Remote Sens.* 52, 2 (1986), 223–227.

[26] Yuhui Shi et al. 2001. Particle swarm optimization: developments, applications and resources. In *Proc. IEEE Congr. Evolutionary Computation (CEC 2001)*, Vol. 1. IEEE, 81–86.

[27] Ashbindu Singh. 1989. Review article digital change detection techniques using remotely-sensed data. *Int. J. Remote Sens.* 10, 6 (1989), 989–1003.

[28] Bing Xue, Mengjie Zhang, and Will N Browne. 2013. Particle swarm optimization for feature selection in classification: A multi-objective approach. *IEEE Trans. Cybern* 43, 6 (2013), 1656–1671.

[29] Qingfu Zhang and Hui Li. 2007. MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans. Evol. Comput.* 11, 6 (2007), 712–731.

[30] Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. 2002. Ensembling neural networks: many could be better than all. *Artif. Intell* 137, 1-2 (2002), 239–263.