# Detection of Heavy Sandstorm Regions Using Composite Differential Evolution Algorithm

**Meera Ramadas and Ajith Abraham**

**Abstract** Dust storm and sand storm are a natural phenomenon that has the potential to affect the health of people and abruptly interrupt the daily activities of public causing hazards, delay in advancement, or even economic losses. Though such natural phenomenon could not be prevented, adequate alertness could prevent the damages and losses that these natural calamities bring to the mankind and its surroundings. The satellite radar images can detect the occurrence of the dust and sandstorm in any portion of the world. These satellite images are revealed with several color-coding to show the severity of the event in any particular area in a given time. Since, quick and accurate analysis of these images are the key for the preparedness against such event, segmenting these radar images according to the severity can help to easily analyze regions of dust storm. Evolutionary algorithms like differential evolution are employed for resolving optimization issues similar to image segmentation. In this writing, an adaptation of differential evolution labeled as composite differential evolution (CODE) is developed. CODE is combined with $k$ means technique for performing segmentation of the radar images. The new technique was found to be efficient, and the image quality of segmentation is greatly improved.

**Keywords** Optimization · $k$ means technique · Control parameters · PSNR · MSE

## 1 Introduction

Sand and dust storms (SDS) are a global environmental issue that can affect the livelihood and health of the mankind. They usually occur in arid and semi-arid regions. These sandstorms can cause destruction to agricultural land, transportation, human health, and infrastructure. Particles of dust move over a large distance which may carry pathogens and harmful substances causing severe health issues. These dust

M. Ramadas (✉) · A. Abraham
Machine Intelligence Research Labs (MIR Labs), Washington, USA

A. Abraham
e-mail: ajith.abraham@ieee.org



B. V. Kumar et al. (eds.), *Differential Evolution: From Theory to Practice*,
Studies in Computational Intelligence 1009,
https://doi.org/10.1007/978-981-16-8082-3_11

storms may be triggered by climatic changes, land degradation, and un-sustainable management of water and land resources.

Radar and microwave images help in the detection of SDS even in cloudy environment. Microwave signals can penetrate through clouds easily, and when a region with dust is spotted, the signal is reflected back. Color bars are used to signify the frequency of the SDS in a particular region. As radar image is complex, it is quite time consuming to precisely specify the hazard severity in a particular area. By segmenting the images on the basis of color bar, the particular risk severity can be observed, and on the basis of intensity, suitable alarms can be triggered.

Image segmentation is a technique of grouping an image into various sections based on the similar characteristics of the image. Problems of image segmentation are categorized as unsupervised learning, and these topics can be cracked through evolutionary approaches of reproduction, mutation, recombination, and selection. Storn and Price [1] familiarized the DE technique which uses the thoughts of evolutionary computing. DE is a simple, hypothetical approach that helps to resolve optimization issues. Effectiveness and robustness of DE is computed on the foundation of control parameters and trial vector generation strategy in usage. Various versions of DE are considered by varying these constraints.

Here, a modification of DE termed composite differential evolution approach (CODE) is created by using three dissimilar mutation constraints. The new constraint F1 gets a random value amid (0, 1), and *N* uses the product of F and F1. This approach is related to other standard variations of DE by tabularization to approve improved efficacy of CODE. This approach is employed in image thresholding for segregating the weather imagery on the basis of definite climatic state. CODE method is used along with *k* means image segmentation technique to achieve the segmented image. The first section of the paper details the differential evolution method and the anticipated variant that was created. The second segment clarifies the idea of *k* means technique on image. The residual section enlightens the implementation of the variant in k means technique and the conclusions found through the study.

### *1.1 Background Study*

Cheng et al. [2] proposed a thresholding technique by executing fuzzy partition on a two- dimensional (2D) histogram on the basis of fuzzy relation and extreme fuzzy entropy principle. The tests conducted show the new technique works better than existing methods. Cheng et al. [3] gave a segmentation procedure for color images on the basis of homogram thresholding and region merging. This technique was compared with histogram-based technique, and efficiency of the new technique was justified. Porte et al. [4] uses Tsallis entropy technique for segmenting images, and the results were tabulated.

Francesca and Schettini [5] performed a study to apply genetic algorithm to find out the borders of skin clusters in multiple color space. Recall and precision scores are used to measure the performance of skin detection approaches. Yin [6] proposed an iterative programming procedure to lessen order of magnitude for calculating MCET objective function. PSO approach is used for probing the near-optimal MCET thresholding. Maitra and Chatterjee [7] devised an enhanced hybrid of PSO which uses both cooperative and comprehensive learning along with few alterations. The hybrid algorithm named HCOCLPSO is assessed through an enhanced GA-based algorithm. Senthilkumar and Rajesh [8] perform a study on the theory of edge detection for image segmentation by means of fuzzy logic, neural networks, and genetic algorithm. Akhilesh et al. [9] projected a modification of PSO for imagery segmentation using prime multilevel thresholding. He also gave an iterative technique for attaining initial values of candidate multilevel thresholds which is based on Otsu technique.

Sarkar and Das [10] devised a technique to integrate 2D histogram associated data for comprehensive multilevel thresholding by means of maximum Tsallis entropy. The results obtained are evaluated using known benchmark segmentation with 300 images. Kurban et al. [11] perform an assessment of evolutionary and swarm-based optimization algorithm for multilevel color image thresholding issues. Here, well-known evolutionary algorithm and swarm-based algorithms are applied to 20 test images for segmentation, and the results are compared. Based on the statistical analysis, swarm-based algorithms show better accuracy in comparison while evolutionary algorithm is faster. Sarkar et al. [12] devised a unique multilevel thresholding method for unsupervised separation between objects and background using minimum cross entropy (MCE). The comparative results show the efficacy of the projected method. Ramadas et al. [13] devised a modification of differential evolution approach coined as FSDE. This technique is used for clustering numerical data. The technique is proposed to be extended for clustering images. Suresh and Shyam [14] proposed modified DE algorithm for enhancing the brightness and contrast of satellite images. Kaur and Kumar [15] proposed image encryption technique using differential evolution. Ramadas et al. [16] projected another modification of DE algorithm termed as FSDE. This approach was used for data clustering, and the efficiency of the approach was verified. Ramadas and Abraham [17] introduced another hybrid of DE named transformed differential evolution which showed heightened results for detecting tumors in MRI images. Krishna and Ravi [18] implemented binary differential evolution for customer segmentation.

## 2 Classical Differential Evolution

In an n-dimensional search space, the chosen number of vectors is recognized casually. In each reiteration, more than one vectors are picked indiscriminately from the populace and are joint to formulate a novel vector. The subsequent vector obtained is equated with pre-obtained target vector to attain a trial vector. If trial vector offers acceptable objective function, then trial vector is recognized for subsequent group. Mutation, recombination, and selection are tracked until certain ending criteria is reached. DE employs the populace of NP candidates indicated by $X_{i,G}$, wherein index $i = 1, 2...NP$ formulates populace while $G$ represent the group of populace.

**Mutation**: For any specified variable $X_{i,G}$, indiscriminately pick three vectors $X_{r1,G}$, $X_{r2,G}$, and $X_{r3,G}$ where $r_1$, $r_2$, $r_3$ are dissimilar from each other. Subsequently donor vector $V_{i,G}$ is calculated by totaling the weighted difference of two vectors to the third vector.

$$V_{i,G} = X_{r1,G} + F \times (X_{r2,G} - X_{r3,G}) \quad (1)$$

The mutation factor $F$ is a fixed value from (0, 1). The mutation function discriminates one DE approach from the other.

**Crossover/Recombination**: Trial vector $U_{i,G}$ is produced for target vector $X_{i,G}$ by means of binomial crossover. Using probability $C_r \in [0, 1][0, 1]$, the modules of donor vector move into trial vector. Crossover probability $C_r$ is designated together with populace dimension $NP$.

$$U_{j,i,G+1} = \begin{cases} V_{j,i,G+1} & if\ rand_{i,j}[0, 1] \le C_r\ or\ if\ j = I_{\text{rand}} \\ X_{j,i,G+1} & if\ \text{rand}_{i,j}[0, 1] > C_r\ or\ if\ j \ne I_{\text{rand}} \end{cases} \quad (2)$$

where, $\text{rand}_{i,j} \approx \cup[0, 1]$ and $I_{\text{rand}}$ is an uninformed number from 1,2…N.

**Selection**: Trial vector and vector in existing population determines the final population that enters the next generation. Target vector $X_{i,G}$ is coupled with trial vector $V_{i,G}$, and the lowermost result of function is considered into subsequent group.

$$X_{i,G+1} = \begin{cases} U_{i,G+1} & if\ f(U_{i,G+1}) \le f(X_{i,G})\ \text{ where } i = 1, 2, ...N \\ X_{i,G} & \text{otherwise} \end{cases} \quad (3)$$

## 3 Composite Differential Evolution

The outcomes of DE algorithm is highly depended on the proper alignment of its parameters. Though studies are being done on parameter alteration of DE, numerous undependable inferences were attained resulting in more research in developing better variants of DE. So different parameter settings are suggested to improve the performance of the resultant variant algorithm. CODE is a variation of classical differential evolution algorithm where the mutation approach in DE is improved. In CODE, three control constraints are employed. The parameter F acknowledged as amplifying parameter uses a fixed value amid (0, 1) and F1 uses a random value amongst (0, 1). The new parameter $N$ uses the product of F and F1. The parameters F, F1, and $N$ permit stochastic deviations in the improvement of the difference vector and hence help to preserve population diversity as the exploration progresses. The variables $X_{r1,G}$,$X_{r2,G}$, $X_{r3,G}$ are chosen arbitrary. During each generation, each of the three randomly chosen trial vector is chosen to generate a new donor vector. The projected approach is defined as:

$$X' = X_{r1,G} + N.\big(F.\big(X_{\text{best},G} - X_{r2,G}\big) - F1.\big(X_{\text{best},G} - X_{r3G}\big)\big) \tag{4}$$

Since three varied control constraints are used, the value of donor vector is heightened greatly, and henceforward, the efficacy of CODE algorithm is also heightened massively. By utilizing the greatest solution vector, this method agrees earlier in contrast to the classical schemes having arbitrary vectors only. The diversity of possible movement in the population increases with increased pairs of solution and hence supports the exploration of search space. This helps to avoid premature convergence and escape local minima. This technique searches the section around each $X_{\text{best},G}$, for each mutated point preserving the exploratory feature and thereby speeding up the convergence. After obtaining the donor vector from mutation strategy in Eq. 4, the trial vector is attained using Eq. 2. Then subsequently, the variables go into the selection stage using Eq. 3.

## 4 Experimental Settings

CODE technique was implemented using MATLABr2017b, and a comparative consequence was achieved. Five old-style mutation strategies are considered, and the projected method CODE and resultant outcomes were related. Fifteen varied benchmark functions were attained, and the consequences were computed by allocating the (VTR) value to reach and number of iterations. The strategy was verified by fitting the magnitude as 25 and 50. One of the results attained is given in Table 1.

By fixing VTR and dimension, the best value, CPU time, and number of function evaluation (NFE) of varied benchmarked techniques are computed. Certain function obtained superior results for all conventional DE and CODE approach. The complete outcome demonstrates that CODE technique has improved efficiency in contrast to the traditional DE strategy. Multiple comparative results from Table 1 is analyzed using a nonparametric test. As more than two algorithms are compared, a multiple comparison statistical type of nonparametric test is employed. Freidman test is the simplest statistical test for multiple comparison which detects if there is a global difference between the related outcomes attained. Friedman statistical test is performed on CODE technique to authenticate the outcomes. On the basis of results from Table 1, the Friedman test was used and the consequences are given in Table 2.

The data from Table 1 is converted to its corresponding ranks by allocating ranks to each value in every row. Then, the average rank of each column is calculated, and the final rank for each algorithm is attained. The ranks gained from the Friedman test is presented in Table 3.

**Table 1** Relative outcomes for dissimilar strategies of DE with CODE

| Function | D | DE /best/1 | DE /best/2 | DE /best-to-rand/1 | DE /rand/1 | DE/rand/2 | CODE |
|---|---|---|---|---|---|---|---|
| Sphere | 50 | 9.614e−016 | 9.561e−016 | 7.645e−016 | 6.923e−016 | 7.217e+0 | 8.63e−016 |
| | 25 | 9.242e−015 | 9.326e−015 | 9.43e−015 | 9.251e−015 | 6.943e+000 | 8.94e−016 |
| Gold | 50 | 3.00e+00 | 3.00e+00 | 3.00e+00 | 3.00e+00 | 3.00e+00 | 3.00e+00 |
| | 25 | 3.00e+00 | 3.00e+00 | 3.00e+00 | 3.00e+00 | 3.00e+00 | 3.00e+00 |
| Beale | 50 | 3.361e−016 | 7.561e−016 | 3.651e−016 | 2.311e−016 | 7.632e−016 | 1.45e−016 |
| | 25 | 4.255e−015 | 1.261e−017 | 1.135e−015 | 7.625e−015 | 7.255e−015 | 1.064e−016 |
| Booth | 50 | 3.43e−016 | 7.041e−016 | 6.03e−016 | 2.04e−016 | 8.232e−016 | 5.92e−016 |
| | 25 | 1.813e−015 | 2.651e−015 | 1.85e−015 | 7.47e−016 | 6.371e−015 | 2.63e−016 |
| Schaffer N.2 | 50 | 6.562e−016 | 6.452e−016 | 4.331e−016 | 8.785e−016 | 8.771e−016 | 2.221e−016 |
| | 25 | 1.232e−015 | 5.253e−015 | 6.567e−016 | 1.236e−015 | 1.243e−015 | 2.222e−016 |
| Schwefel | 50 | −1.71e+003 | −1.283e+003 | −7.74e+001 | −2.13e+003 | −1.54e+003 | −5.81e+002 |
| | 25 | −4.132 + 002 | −4.327e+003 | −1.57e+003 | −4.54e+002 | −1.43e+003 | −9.33e+002 |
| Michlewicz | 50 | −7.541e+00 | −6.826 e+00 | −7.34e+00 | −7.121e+00 | −6.74 e+00 | −1.14e+001 |
| | 25 | −7.569e+00 | −7.28e+00 | −6.77e+00 | −7.543e+00 | −6.89 e+00 | −7.52e+00 |
| Schaffer N.4 | 50 | 3.021e−015 | 2.835e−001 | 2.732–001 | 2.897e−001 | 2.798e−001 | 2.791e−001 |
| | 25 | 2.895e−001 | 2.93e−001 | 2.91e−001 | 2.98e−001 | 2.92e−001 | 2.91e−001 |
| Himmel blau | 50 | 1.561e−016 | 9.025e−016 | 3.753e−016 | 8.08e−016 | 1.548e−016 | 2.021e−016 |
| | 25 | 4.731e−015 | 3.857e−015 | 1.897e−015 | 4.37e−015 | 5.117e−015 | 4.52e−016 |
| Bird | 50 | −1.06e+002 | −1.058e+002 | −1.047e+002 | −1.08e+002 | −1.03e+002 | −1.021e+002 |
| | 25 | −9.25e+001 | −1.024e+002 | −1.056e+002 | −1.09e+002 | −1.07e+002 | −1.056e+002 |
| Extended cube | 50 | 3.232e−015 | 1.898e−005 | 6.217e−008 | 4.89e−006 | 2.589e+00 | 3.95e−002 |

(continued)

**Table 1** (continued)

| Function | D | DE /best/1 | DE /best/2 | DE /best-to-rand/1 | DE /rand/1 | DE/rand/2 | CODE |
|---|---|---|---|---|---|---|---|
| | 25 | 5.786e−008 | 1.676e−005 | 7.117e−008 | 5.12e−005 | 2.89e+009 | 2.52e−002 |
| Ackeley | 50 | 7.094e−015 | 3.58e−013 | 7.99e−015 | 6.38e−012 | 3.045e+00 | 4.44e−015 |
| | 25 | 7.892e−015 | 3.49e−015 | 7.897e−015 | 5.08e−015 | 3.17e+00 | 4.41e−015 |
| Griewank | 50 | 9.891e−016 | 6.468e−013 | 1.567e−013 | 9.88e−016 | 1.046e+00 | 9.99e−016 |
| | 25 | 1.378e−002 | 5.035e−009 | 7.786e−015 | 9.14e−015 | 1.059e+00 | 8.88e−016 |
| Rastrigin | 50 | 1.893e+001 | 1.186e+002 | 7.375e+001 | 1.13e+002 | 1.457e+002 | 0 |
| | 25 | 3.567e+001 | 1.66e+002 | 8.073e+001 | 1.08e+002 | 1.574e+002 | 0 |
| Rosenbrock | 50 | 9.564e−016 | 3.779e−009 | 7.788e−016 | 1.035e−008 | 1.056e+005 | 1.8e+001 |
| | 25 | 3.882e+00 | 1.4565e−011 | 6.896e−015 | 1.48e−008 | 7.057e+004 | 2.09e+001 |

**Table 2** Test statistics

| N | 50 |
|---|---|
| Chi sq | 22.43 |
| Df | 5 |
| Asymptotic significance | 0.0004 |

**Table 3** Mean ranking of various approaches

| Strategies | Mean rank on best value |
|---|---|
| DE/best/1 | 2.73 |
| DE/best/2 | 4.32 |
| DE/rand/1 | 3.12 |
| DE/best-to-rand/1 | 2.76 |
| DE/rand/2 | 5.11 |
| CODE | 3.0 |

## 5 Image Segmentation Technique

The technique used for image segmentation is the $k$ means algorithm. $K$ means algorithm was introduced by MacQueen [19]. This algorithm is commonly used in image segmentation to group the pixel objects on the basis of features into k groups. The pixels of the image are related to colors defined in RGB. The consortium of pixels is performed by using the minimal distance between the object and centroid.

Let us consider an image with resolution $x, y$, and the image has to be segmented into $k$ clusters. Input image is considered as $p(x, y)$ which has to be grouped and $c_k$ is the cluster center. The minimal distance is calculated using the Euclidean distance which is denoted as:

$$\text{distance}(x, y) = \left\{\sum_i (x_i - y_i)^2\right\}^{\frac{1}{2}} \tag{5}$$

The objective function for this technique is the calculation of the centroid of the cluster which is denoted as:

$$c_k = \frac{1}{k} \sum_{y \in c_k} \sum_{x \in c_k} p(x, y) \tag{6}$$

The algorithm for $k$ means for clustering an image is given as:

Step 1: Select the number of cluster $k$.

Step 2: Take an arbitrary point as center.

Step 3: Repeat steps 4 to 6 for each pixel of the image until the condition is satisfied.

Step 4: Compute the Euclidean distance $d$ between the center and each pixel of image using Eq. 5.

Step 5: Allocate all pixels to closest center on the basis of distance $d$.

Step 6: Re-compute the new center for the cluster using Eq. 6

Step 7: End loop.

Step 8: Restructure cluster pixels into an image.

This technique collects data of similar type together. The superiority of the image is subject to the number of clusters and the initial centroid considered.

## 6 Multi-level Thresholding with CODE Approach

$K$ means algorithms often suffer from the optimization of the mean squared error condition and is confined in local minima. The new improved CODE algorithm helps to avoid local minima. CODE approach is applied to $k$ means technique to enhance the performance of image segmentation. The elements of the CODE population takes the form of a vector which contains the coordinates of centroids of $K$ clusters. Each vector is assigned to the cluster represented by the closest centroid. A population is created randomly in which the each individual represents a class center. A pixel is assigned to the class with nearest center. Fitness of each solution is calculated by finding the distance between the pixels and cluster centers. Offspring is generated using the mutation and crossover functions of CODE algorithm. The performance of the offspring are compared with the parent, and the best solutions are considered into the next generation based on the objective function value shown in Eq. 6. These procedures are repeated until the centroid remains unchanged. The flow diagram for the planned work is given in Fig. 1.

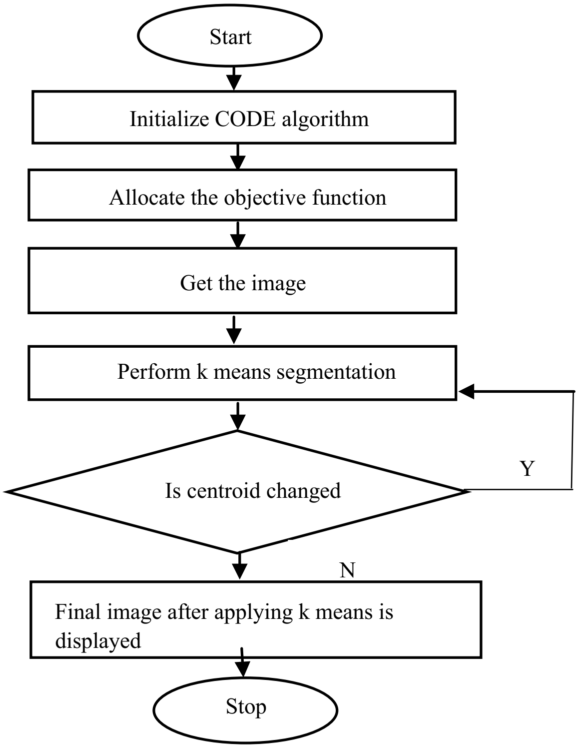
## 7 Test Results on Image Segmentation

CODE strategy was executed and was merged with the $k$ means technique to achieve image segmentation. This technique is implemented on two groups of weather images. The conclusions achieved establish that the projected CODE approach gives boosted consequences for segmentation in comparison with traditional DE technique. Weather images from www.vizrt.com are considered, and $k$ means segmentation is applied on these images. The original images are segmented based on $k$ means and CODE with $k$ means, and the outcomes are given in Figs. 2, 3, 4, and 5.

The superiority of the technique is substantiated with the peak-to-signal ratio (PSNR) and CPU time. PSNR is the degree of superiority among the real imagery and the sectional imagery on the basis of mean square error (MSE). It is denoted as:

$$\mathrm{PSNR}(\sigma, s) = 20 \log_{10}\left[\frac{255}{\sqrt{\mathrm{MSE}(\sigma, s)}}\right] \tag{7}$$

**Fig. 1** Flowchart for CODE with *k* means



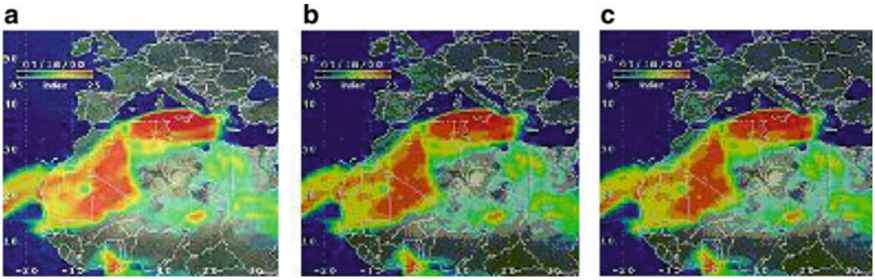




**Fig. 2** Breakdown of image 1 **a** Real imagery **b** Sectional imagery using *k* means **c** Sectional imagery using CODE with *k* means

where $\sigma$ is the real imagery and $s$is the sectional imagery. If the magnitude of imagery is $m \times n$, then the mean square error (MSE) is computed as:

$$\mathrm{MSE} = \frac{1}{m * n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [\sigma(m, n) - s(m, n)] \tag{8}$$

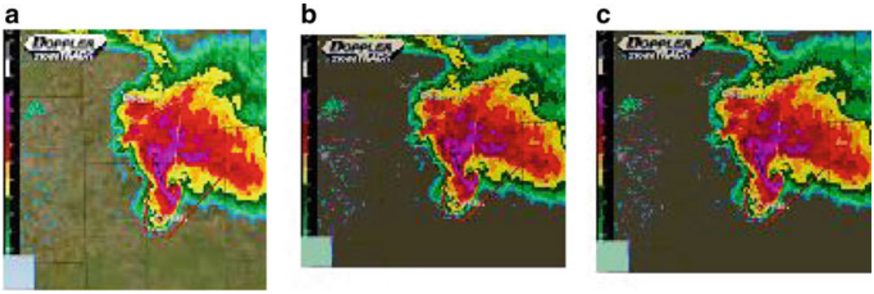


**Fig. 3** Breakdown of image 2 **a** Real imagery **b** Sectional imagery using *k* means **c** Sectional imagery using CODE with *k* means

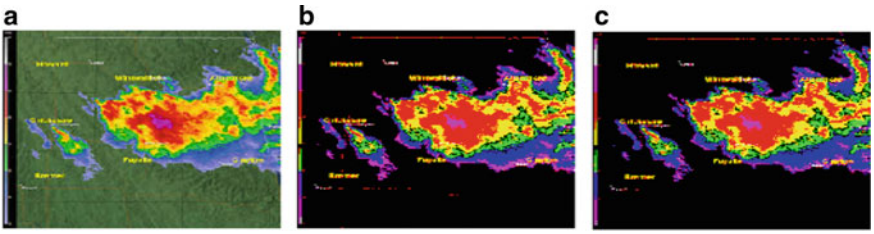


**Fig. 4** Breakdown of image 3 **a** Real imagery **b** Sectional imagery using *k* means **c** Sectional imagery using CODE with *k* means

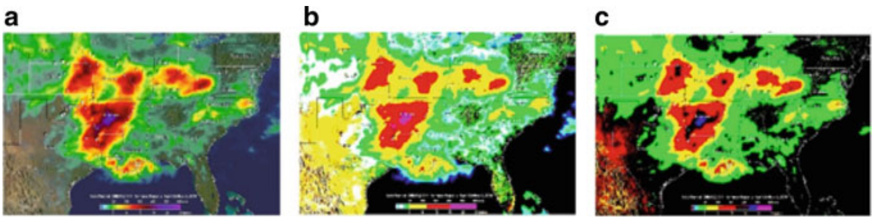


**Fig. 5** Breakdown of image 2 **a** Real imagery **b** Sectional imagery using *k* means **c** Sectional imagery using CODE with *k* means

Table 4 gives the comparative values of PSNR and CPU time taken for segmenting the weather imagery using traditional *k* means technique and various variants of DE algorithm and CODE algorithm with *k* means technique. The outcomes depict that the values attained for CODE with *k* means technique is finest in assessment to the classical techniques. Sandstorm radar images are quite complex, and it is quite difficult to detect the regions of extreme sandstorm. By segmenting the image, we can extract regions of the image based on hazard severity. This helps in giving alert to the public to take precautions during such extreme conditions. The precise segmented

**Table 4** Assessment of PSNR value and CPU time

| Imagery | PSNR | | | | | | | CPU time | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *K* means | DE/best/1 with *k* means | DE/best/2 with *k* means | DE/rand/1 with *k* means | DE/best-to-rand/1 with *k* means | DE/rand/2 with *k* means | CODE with *k* means | *K* means | DE/best/1 with *k* means | DE/best/2 with *k* means | DE/rand/1 with *k* means | DE/best-to-rand/1 with *k* means | DE/rand/2 with *k* means | CODE with *k* means |
| Image 1 | 30.05 | 30.5 | 29.7 | 30.5 | 29.75 | 30.28 | 32.1 | 1.2 | 1.25 | 1.19 | 1.2 | 1.18 | 1.19 | 2.01 |
| Image 2 | 30.2 | 30.3 | 29.4 | 30.28 | 29.98 | 30.03 | 28.3 | 1.3 | 1.28 | 1.25 | 1.27 | 1.25 | 1.01 | 1.02 |
| Image 3 | 30.18 | 30.2 | 30.1 | 29.8 | 30.05 | 30.13 | 30.1 | 1.2 | 1.19 | 1.21 | 1.18 | 1.19 | 1.2 | 1.18 |
| Image 4 | 29.89 | 30.1 | 29.8 | 29.75 | 30.12 | 30.05 | 29.56 | 1.25 | 1.3 | 1.28 | 1.28 | 1.31 | 1.26 | 1.26 |

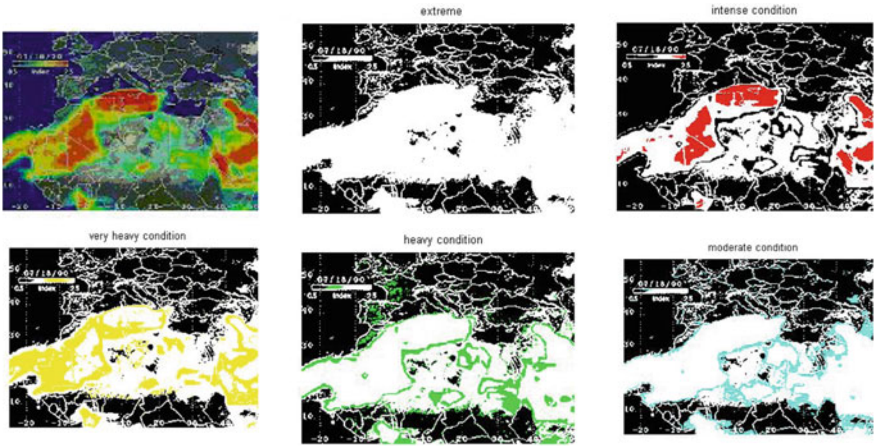



**Fig. 6** Image 1 segmented on the basis of sandstorm risk severity

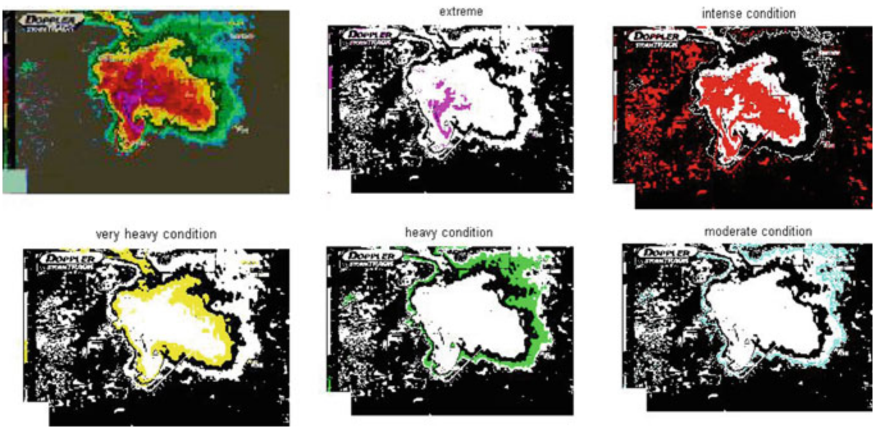



**Fig. 7** Image 1 segmented on the basis of sandstorm risk severity

section of radar images for the model images based on hazard severity are shown in Figs. 6, 7, 8, and 9.

## 8 Conclusions

CODE was implemented, and the outcome was related with the traditional mutation techniques. The outcome showed that CODE was highly efficient in comparison with traditional techniques. CODE was then combined with *k* means algorithm to perform segmentation of weather radar images. The efficiency of the segmented image was

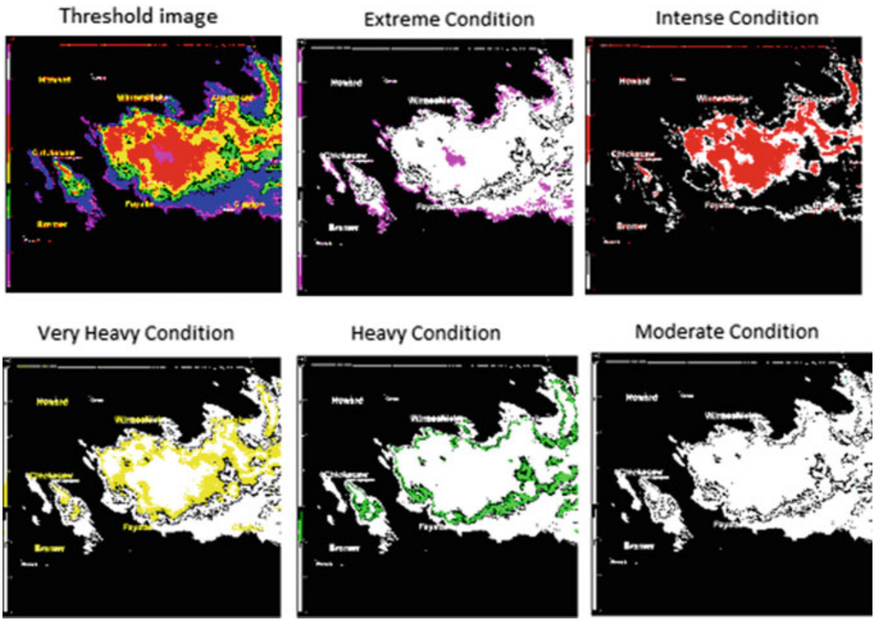



**Fig. 8** Image 3 segmented on the basis of sandstorm risk severity

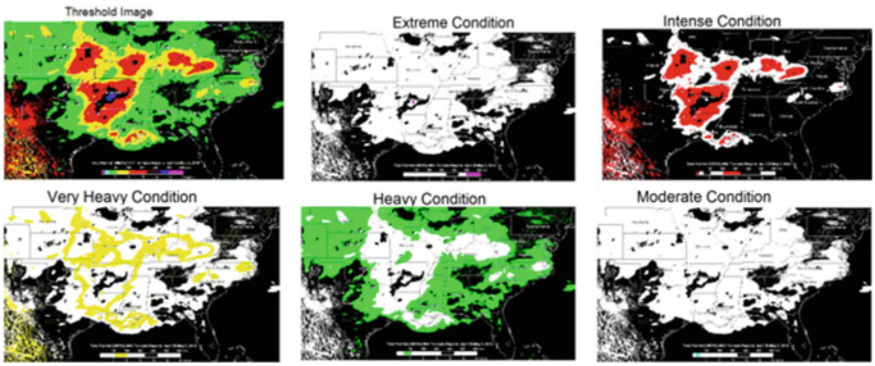



**Fig. 9** Image 4 segmented on the basis of sandstorm risk severity

verified by testing the performance of algorithm used by means of PSNR and CPU time taken. The segmented imagery was then clustered on the basis of hazard severity. This technique can be applied for image enhancement, texture analysis, etc.

# Appendix

*A. Benchmark functions used:*

Sphere function:

$$f(x) = \sum_{i=1}^{d} x_i^2$$

where $-5.12 < x_i < 5.12$.

Beale function:

$$f(x) = (1.5 - x_1 + x_1x_2)^2 + \left(2.25 - x_1 + x_1x_2^2\right)^2 + \left(2.625 - x_1 + x_1x_2^3\right)^2$$

where $-4.5 < x_i < 4.5$.

Booth function:

$$f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2$$

where $-10 < x_i < 10$.

Schwefel function:

$$f(x) = 418.9829d - \sum_{i=1}^{d} x_i \sin\left(\sqrt{|x_i|}\right)$$

where $-500 < x_i < 500$.

Michalewicz function:

$$f(x) = -\sum_{i=1}^{d} \sin(x_i) \sin^{2m}\left(\frac{ix_i^2}{\pi}\right)$$

where $0 < x_i < \pi$.

Schaffner function N.2:

$$f(x, y) = 0.5 + \frac{\sin^2(x^2 - y^2) - 0.5}{(1 + 0.001(x^2 + y^2))^2}$$

where $-100 < x_i < 100$.

Schaffner function N.4:

$$f(x, y) = 0.5 + \frac{\cos^2(\sin^2(\left|x^2 - y^2\right|)) - 0.5}{(1 + 0.001(x^2 + y^2))^2}$$

where $-100 < x_i < 100$.

Himmel Blau function:

$$f(x, y) = (x^2 + y - 11)^2 + (y^2 + x - 7)^2$$

where $-5 < x_i < 5$.
Bird function:

$$f(x, y) = \sin(x)e^{(1-\cos(y))^2} + \cos(y)e^{(1-\sin(x))^2} + (x - y)^2$$

where $-2\pi < x_i < 2\pi$.
Extended cube function:

$$f(x) = \sum_{i=1}^{n} 100(x_{i+1} - x_i^3)^2 + (1 - x_i)^2$$

where $-100 < x_i < 100$.
Ackley function:

$$f(x) = -a \ \exp\left(-b\sqrt{\frac{1}{d}\sum_{i=1}^{d} x_i^2}\right) - \ \exp\left(\frac{1}{d}\sum_{i=1}^{d}\cos(cx_i)\right) + a + \exp(1)$$

where $-15 < x_i < 30$; $a = 20$, $b = 0.2$, $c = 2\pi$.
Goldstein-price function:

$$\begin{aligned} f(x) = (1 + (x + y + 1)^2(19 - 14x + 3x^2 - 14y6xy + 3y^2))(30 \\ + +(2x - 3y)^2(18 - 32x + 12x^2 + 48y - 36xy + 27y^2)) \end{aligned}$$

where $-2 < x_i < 2$.
Griewank function:

$$f(x) = \sum_{i=1}^{d} \frac{x_i^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$$

where $-600 < x_i < 600$.
Rastrigen function:

$$f(x) = 10d + \sum_{i=1}^{d} [x_i^2 - 10\cos(2\pi x_i)]$$

where $-5.12 < x_i < 5.12$.
Rosenbrock function:

$$f(x) = \sum_{i=1}^{d-1}\left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right]$$

where $-5 < x_i < 10$.

## References


1. Storn, R., Price, K.: Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. J. Global OPTIM **11**(4), 341–359 (1997)
2. Cheng, H.D., Chen, Y.H., Jiang, X.H.: Thresholding using two-dimensional histogram and fuzzy entropy principle. IEEE T Image Process **9**(4), 732–735 (2000)
3. Cheng, H.D., Jiang, X.H., Wang, J.: Color image segmentation based on homogram thresholding and region merging. Pattern Recogn. **35**(2), 373–393 (2002)
4. Portes, D.A.M., Israel, A.E., Gesualdi, M.A.R.: Image thresholding using Tsallis entropy. Pattern Recogn. Lett. **25**(9), 1059–1065 (2004)
5. Gasparini, F., Schettini, R.: Skin segmentation using multiple thresholding. Proc. SPIE **6061**, 128–135 (2006)
6. Yin, P.Y.: Multilevel minimum cross entropy threshold selection based on particle swarm optimization. Appl. Math. Comput. **184**(2), 503–513 (2007)
7. Maitra, M., Chatterjee, A.: A hybrid cooperative–comprehensive learning based PSO algorithm for image segmentation using multilevel thresholding. Expert. Syst. Appl. **34**(2), 1341–1350 (2008)
8. Senthilkumaran, N., Rajesh, R.: Edge detection techniques for image segmentation–a survey of soft computing approaches. Int. J. Rec. Trends. Eng. **1**(2), 250–254 (2009)
9. Akhilesh, C., Chatterjee, A., Patrick, S.: A new social and momentum component adaptive PSO algorithm for image segmentation. Expert. Syst. Appl. **38**(5), 4998–5004 (2011)
10. Sarkar, S., Das, S.: Multilevel image thresholding based on 2D histogram and maximum Tsallis entropy—a differential evolution approach. IEEE T Image Process **22**(12), 4788–4797 (2013)
11. Kurban, T., Pinar, C., Rifat, K., Erkan, B.: Comparison of evolutionary and swarm based computational techniques for multilevel color image thresholding. ADV Soft. Comp. **23**, 128–143 (2014)
12. Sarkar, S., Das, S., Sheli, S.C.: A multilevel color image thresholding scheme based on minimum cross entropy and differential evolution. Pattern. Recogn. Lett. **54**, 27–35 (2015)
13. Ramadas, M., Abraham, A., Kumar, S.: FSDE-forced strategy differential evolution used for data clustering. J. King. Saud. Univ. Comp. Info. Sci. (2016).
14. Suresh, S., Lal, S.: Modified differential evolution algorithm for contrast and brightness enhancement of satellite images. Appl. Soft. Comput. 622–641
15. Kaur, M., Kumar, V.: Colour image encryption technique using differential evolution in non-subsampled contourlet transform domain. IET IMAGE Process. **12**(7), 1273–1283 (2018)
16. Ramadas, M., Abraham, A., Kumar, S.: FSDE-Forced strategy differential evolution used for data clustering. J. King. Saud. Univ. Comput. Inf. Sci. (2019)
17. Ramadas, M., Abraham, A.: Detecting tumours by segmenting MRI images using transformed differential evolution algorithm with Kapur's thresholding. Neural. Comput. Appl. **32**(10) (2020)
18. Krishna, G.J., Ravi, V.: High utility item set mining using binary differential evolution: an application to customer segmentation. Expert. Syst. Appl. **181**, 115–122 (2021)
19. MacQueen, J.B.: On the asymptotic behavior of k-means. California Univ. Los Angeles Western Manage.ment Sci.ence Inst. (1965)