CrossMark

A modified adaptive differential evolution algorithm for color image segmentation

Ahmad Khan · M. Arfan Jaffar · Ling Shao

Abstract   Image segmentation is an important low-level vision task. It is a perceptual grouping of pixels based on some similarity criteria. In this paper, a new differential evolution (DE) algorithm, modified adaptive differential evolution, is proposed for color image segmentation. The DE/current-to-pbest mutation strategy with optional external archive and opposition-based learning are used to diversify the search space and expedite the convergence process. Control parameters are automatically updated to appropriate values in order to avoid user intervention of parameters setting. To find an optimal number of clusters (the number of regions or segments), the average ratio of fuzzy overlap and fuzzy separation is used as a cluster validity index. The results demonstrate that the proposed technique outperforms state-of-the-art methods.

1 Introduction

Image segmentation and grouping similar visual objects have received an extensive attention in the last decades. It plays an important role in computer vision and graphics applications such as: object localization or recognition, data compression, tracking, and image retrieval.

A. Khan
National University of Computer and Emerging Sciences, H-11/4, Islamabad, Pakistan
e-mail: ahmadkhan.nu@gmail.com

M. A. Jaffar (✉)
College of Computer and Information Sciences, Al Imam Mohammad Ibn Saud Islamic University
(IMSIU), Riyadh, Saudi Arabia
e-mail: arfan.jaffar@gmail.com; arfan.jaffar@ccis.imamu.edu.sa

A. Khan · L. Shao
Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield, UK

L. Shao
e-mail: ling.shao@sheffield.ac.uk

 Springer

Image segmentation techniques can be divided into image-domain-based, physics-based and feature-space-based techniques [18]. All these techniques with different variations have been used extensively, but every one shares some pros and cons. Image-domain-based techniques use both color features and spatial relationship, among pixels, to carry out segmentation task. The main difficulty lies in these techniques is the selection of seed region. Physics-based techniques use the reflection properties of material to carry out color image segmentation. It performs better in a situation where the reflectance of underlying objects are known; hence, these techniques are application specific. The feature-based techniques use only color features and ignore spatial relationship; therefore, the segmented regions are usually fragmented [17]. The contour-based approaches start from an initial boundary shape in the form of spline curves; these curves are iteratively evolved to minimize some energy function [6,10,22,30]. The difficulty lies in these techniques is the manual initialization of spline curves to the boundaries of an object of interest.

Markov Random Field (MRF) as another statistical model that uses spatial connectivity among neighboring pixels [2]. Many approaches are based on Markov Random Fields such as Graph Cut [12–14,25]. These algorithms solve the two-class problem, i.e., separating the background and foreground objects, and need human interaction to manually specify the seed regions. Similarly, the background and foreground segmentation is treated with watershed segmentation [3,29]. The watershed approach suffers from over-segmentation that can be handled by morphological operations. Mooveover, the segmentation accuracy is highly sensitive to morphological structuring element. Recently, [16] proposed a supper pixel segmentation algorithm that uses entropy rate of random walk on the graphs. This algorithm performs best in the case of supper pixel segmentation, but for general segmentation it produces over-segmented results.

Image segmentation can be treated as a clustering problem. Features of each pixel correspond to a pattern, and the combination of the pixels (i.e., segment) corresponds to a cluster. Keeping this analogy in mind, many clustering algorithms have been used for image segmentation [1,5,7]. Among the clustering algorithms, adopted for color image segmentation, the most famous is the k-means [7]. The fuzzy c-means (FCM) [1] has also extensively used for image segmentation due to its ease of implementation and simplicity. However, its implementation often faces two problems: 1) how many clusters should be made; 2) and how to select the initial centroids that are properly distributed in the image. The number of cluster centers and initialization has considerable impact on final results. Tan and Isa [27] used the concept of histogram thresholding to initialize the cluster centers to the dominating colors. They used compactness as an objective function to optimize the cluster centers. This approach performs poor for complex color images because only cluster compactness is not good enough to validate the segmentation accuracy. Level sets and spatial fuzzy have been combined for medical image segmentation [15]. This algorithm performs well for two-class problem but unable to segment general color images of complex scenes. Another color image segmentation technique based on the merging of maximal similarity of regions is proposed in [23]. Bhattacharyya distance between the histograms of the regions is used as a cluster validity index. The main weakness of this approach is the human intervention to mark the seed regions.

In this paper, we propose a novel differential evolution algorithm based on spatial fuzzy c-mean (sFCM) [5] to make compact and well-separated clusters. A combination of fuzzy separation and fuzzy overlap is used as a cluster validity index to determine the optimal number of clusters automatically. Initialization greatly effects convergence and diversity. Purely random initial solutions have more probability to visit or even revisit the unproductive regions of the search space [24]. The opposite number decreases the chances to visit the

unproductive regions again and again. For this purpose, we take the best individuals from the union of purely random solutions and their opposites as initial population. An external archive is maintained to store the dropped solutions in order to give chances to poor solutions in the evolution process. Poor solutions are not always poor, but it can give best results with slight perturbation. To represent each cluster center, we use the HSV color space relying on the hue, saturation, and value properties the color. Such color space allows us to specify colors in a way that is close to human experience of colors.

The rest of the paper is organized as follows. The basic concept of classical differential evolution algorithm is presented in Sect. 2. Section 3 explains the proposed algorithm, modified adaptive differential evolution (MoADE), in detail. Experimental results are described in Sect. 4 followed by conclusion.

2 Classical differential evolution algorithm

Differential evolution (DE) is a relatively recent population-based evolutionary heuristic. It is used to optimize problems over continuous domains. In DE, a vector of real numbers is used to represent a decision variable or solution. An individual vector can be represented, is a bold face letter $\boldsymbol{x}$, in the following way:

$$\boldsymbol{x} = [x^{(1)}, x^{(2)}, x^{(3)}, \ldots, x^{(D)}] \tag{1}$$

where the superscripts $(1, 2, 3, \ldots, D)$ represent the dimension of the problem. In classical differential evolution, the initial population $\{\boldsymbol{x}_{i,0} = (x_{i,0}^{(1)}, x_{i,0}^{(2)}, x_{i,0}^{(3)}, \ldots x_{i,0}^{(D)})\} \mid i = 1, 2, 3, \ldots N_p$ is generated randomly according to uniform distribution $x_{low}^j \leq x_{i,0}^j \leq x_{up}^j$ for $j = 1, 2, \ldots \ldots D$ where D is the dimension of the problem and N_p is the population size. After initialization DE enters to a loop of evolutionary operations: mutation, crossover, and selection.

Mutation: To mutate a particular target vector $\boldsymbol{x}$ at iteration t, three vectors are randomly selected from the current population. In other words, the jth component of a trial vector $\boldsymbol{x}$ at iteration t is generated as follows:

$$\vartheta_t^j = x_t^j + F \cdot (x_{r1,t}^j - x_{r2,t}^j) \tag{2}$$

where the indices $r1$ and $r2$ are distinct integers uniformly chosen from the set $\{1, 2, \ldots, N_p\}$, and F is the mutation factor that lies in interval $(0, 1+)$.

Crossover: The crossover operation is performed to increase the diversity of the perturb vector. A binomial crossover operation forms the final trial/offspring vector $\boldsymbol{v}_t = (v_t^1, v_t^2, \ldots, v_t^D)$

$$v_t^j = \begin{cases} \vartheta_t^j & \text{if } rand(0, 1) \leq CR \quad or \quad j = j_{rand}, \\ x_t^j, & \text{otherwise} \end{cases} \tag{3}$$

where $rand(0, 1)$ is a uniform random number in the interval $[0, 1]$ and independently generated for each j and i, respectively, $j_{rand} = randint(1, D)$ is an integer randomly chosen from 1 to D. The crossover probability $CR \in [0, 1]$ roughly corresponds to the average fraction of vector components that are inherited from the mutation vector.

Selection: It decides whether a particular trail individual $\boldsymbol{v}$ or the parent $\boldsymbol{x}$ at generation t should be allowed to next generation. Selection is performed on the bases of fitness values $f(\cdot)$ of trail and parent vectors. For example, if we have a minimization problem, then the

selection is given by:

$$\boldsymbol{x}_{t+1} = \begin{cases} \boldsymbol{v}_t & \text{if } f(\boldsymbol{v}_t) < f(\boldsymbol{x}_t) \\ \boldsymbol{x}_t, & \text{otherwise} \end{cases} \tag{4}$$

The operation in Eq. (4) is called a successful update if the trial vector $\boldsymbol{v}$ is better than parent $\boldsymbol{x}$, i.e., the improvement or evolution progress $\triangle = f(\boldsymbol{x}) - f(\boldsymbol{v})$ is positive. The evolutionary process continues up to a fixed number of iterations. The best vector in the last generation is considered the solution.

3 Modified adaptive differential evolution

In classical differential evolution, the crossover and mutation parameters are fixed to generate all trial vectors at all generations. In adaptive DE algorithms, each individual i is associated with its own crossover probability CR_i and mutation probability F_i. In this paper, an adaptive differential evolution ADE [32]) is applied to the domain of color image segmentation by incorporating a new objective function and opposition-based learning. The new algorithm is called MoADE. The pseudocode of algorithm 1 explains MoADE in detail.

DE algorithm needs to be initialized before start of iterative procedure. In practice, initialization is either done by the user using some sort of the domain knowledge or random procedure. The pure random initialization has proved to be inappropriate since it forces the clustering algorithms to converge to local optima. It means there is more probability to visit or even revisit the unproductive region of the search space. The opposite numbers have high chance to produce best results than purely random ones. Here, the initial population takes the best individuals from the union of purely random solutions and their opposites [24] in order to avoid unproductive samples or solutions. In algorithm 1, steps $2 - 8$ show the implementation of opposition-based initialization. The opposite of each component of each individual can be calculated as:

$$\bar{x}_i^j = a^j + b^j - x_i^j \quad i = 1, 2, \ldots, N_p; \ j = 1, 2, \ldots, D \tag{5}$$

where x_i^j and $\bar{x}_i^j$ denote the jth component/dimension of the ith vector and its opposite, respectively, a^j and b^j represent the intervals of the jth dimension of the search space. The probability $randn(0, 1) < 1/(3 + e^{-1/g})$ encourages high exploration in initial stages while high exploitation in the final stages. If the probability holds for particular individual $\boldsymbol{x}_i$ then its opposite can be calculated as:

$$\bar{\boldsymbol{x}}_{i,g} = \text{MIN}^d + \text{MAX}^d - \boldsymbol{x}_{i,g} \tag{6}$$

where MIN^d and MAX^d are the minimum and maximum values of a particular dimension in the current generation g. The intervals $[\text{MIN}^d, \text{MAX}^d]$ encourages to stay in already shrunken search space. If other boundary intervals are used rather than $[\text{MIN}^d, \text{MAX}^d]$, then we would jump out side of the already shrunken search space and would lost the knowledge of the current reduced space (converged population).

Sometimes inferior solutions, with little change, provide additional information about the progress direction compared to current population [32]. For this purpose, the mutation strategy DE/current-to-pbest/1 with archives [32] is adopted. This mutation strategy uses a set of archived inferior solutions denoted by $\mathbf{A}$ and the current population denoted by $\mathbf{P}$ to generate a mutation vector as given by Eq. (7).

$$\boldsymbol{\vartheta}_{i,g} = \boldsymbol{x}_{i,g} + F_i(\boldsymbol{x}_{best,g} - \boldsymbol{x}_{i,g}) + F_i(\boldsymbol{x}_{r1,g} - \boldsymbol{x}_{r2,g}) \tag{7}$$

Algorithm 1 Procedure with archives

1: Initialize: $\mu_{CR} = 0.5$, $\mu_F = 0.5$, $\mathbf{A} = \emptyset$ and N_p;
2: Generate a population $\boldsymbol{x}_i$, $i = 1, 2, \ldots, N_p$ randomly.
3: **for** $i = 1$ to N_p **do**
4:    **for** $j = 1$ to D **do**
5:      $\bar{x}_i^j = a^j + b^j - x_i^j$
6:    **end for**
7: **end for**
8: Select the N_p fittest individuals from the set $\{\boldsymbol{x}_i \cup \bar{\boldsymbol{x}}_i \mid i = 1, 2, \ldots, N_p\}$ as initial population $\mathbf{P}$
9: **while** ($g \leq G$) **do**
10:   $S_F = \emptyset$ $S_{CR} = \emptyset$;
11:   **for** $i = 1$ to N_p **do**
12:      Generate $CR_i = randn(\mu_{CR}, 0.1)$, $F_i = randc(\mu_F, 0.1)$
13:      Randomly chose $\boldsymbol{x}_{r1,g}$, $\boldsymbol{x}_{i,g}$ from $\mathbf{P}$
14:      Randomly chose $\boldsymbol{x}_{r2,g}$ from $\mathbf{P} \cup \mathbf{A} \mid \boldsymbol{x}_{r2,g} \neq \boldsymbol{x}_{r1,g} \neq \boldsymbol{x}_{i,g}$
15:      **if** $randn(0, 1) < 1/(3 + e^{-1/g})$ **then**
16:        $\bar{\boldsymbol{x}}_{i,g} = \text{MIN}^d + \text{MAX}^d - \boldsymbol{x}_{i,g}$
17:        **if** $f(\bar{\boldsymbol{x}}_{i,g}) \leq f(\boldsymbol{x}_{i,g})$ **then**
18:          $\boldsymbol{x}_{i,g} = \bar{\boldsymbol{x}}_{i,g}$
19:        **end if**
20:      **end if**
21:      Randomly chose $\boldsymbol{x}_{best}$ from the top p % of $\mathbf{P}$
22:      $\boldsymbol{\vartheta}_{i,g} = \boldsymbol{x}_{i,g} + F_i(\boldsymbol{x}_{best,g} - \boldsymbol{x}_{i,g}) + F_i(\boldsymbol{x}_{r1,g} - \boldsymbol{x}_{r2,g})$
23:      Generate $J_{rand} = randn(1, D)$
24:      **for** $j = 1$ to D **do**
25:        **if** $j = J_{rand}$ or $rand(0, 1) < CR_i$ **then**
26:          $\upsilon_{i,g}^j = \vartheta_{i,g}^j$
27:        **else**
28:          $\upsilon_{i,g}^j = x_{i,g}^j$
29:        **end if**
30:      **end for**
31:      **if** $f(\boldsymbol{x}_{i,g}) \leq f(\boldsymbol{\upsilon}_{i,g})$ **then**
32:        $\boldsymbol{x}_{i,g+1} = \boldsymbol{x}_{i,g}$
33:      **else**
34:        $\boldsymbol{x}_{i,g+1} = \boldsymbol{\upsilon}_{i,g}$; $\mathbf{A} = \mathbf{A} \cup \boldsymbol{x}_{i,g}$; $S_{CR} = S_{CR} \cup CR_i$; $S_F = S_F \cup F_i$
35:      **end if**
36:   **end for**
37:   Randomly remove solutions from $\mathbf{A}$ so that $|\mathbf{A}| \leq N_p$
38:   $\mu_{CR} = (1 - c) \cdot \mu_{CR} + c \cdot mean_W(S_{CR})$
39:   $\mu_F = (1 - c) \cdot \mu_F + c \cdot mean_L(S_F)$
40: **end while**

---

where $\boldsymbol{x}_{best,g}$ is randomly chosen as one of the top p % individual in the current population and F_i is the mutation factor that is associated with $\boldsymbol{x}_i$ and re-generated at each generation by the adaptation process introduced later in Eq. (12). The vectors $\boldsymbol{x}_{i,g}$ and $\boldsymbol{x}_{r1,g}$ are selected from $\mathbf{P}$, while $\boldsymbol{x}_{r2,g}$ is randomly selected from $\mathbf{P} \cup \mathbf{A}$.

Initially the archive is empty. Then after each generation the parent solutions, that fail in the selection process, are added to it. The archive is made of fixed size to avoid significant computation overhead. If the archive size exceeds a certain threshold, say N_p, then some solutions are randomly removed to keep the archive size at N_p.

The archive provides information about the progress direction and improves the population diversity. The larger F increases the population diversity but it tends to converge to local minima. The proposed algorithm with archives and opposition-based learning is able to search the space effectively in order to come out of local optima.

3.1 Parameters adaptation

Similar to [32], the crossover probability CR_i of each individual $\boldsymbol{x}_i$ is independently generated according to a normal distribution of mean μ_{CR} and standard deviation 0.1.

$$CR_i = randn_i(\mu_{CR}, 0.1) \tag{8}$$

The set of all successful crossover probabilities CR_is at generation g is represented by S_{CR}. The mean μ_{CR} is initialized to be 0.5 and then updated at the end of each generation as:

$$\mu_{CR} = (1 - c) \cdot \mu_{CR} + c \cdot mean_W(S_{CR}) \tag{9}$$

where c is a positive constant between 0 and 1 and $mean_W(\cdot)$ is the weighted average of all successful crossover probabilities at generation g. The $mean_W(\cdot)$ can be calculated as:

$$mean_W(S_{CR}) = \sum_{s_j \in S_{CR}} w(\mu, s_j)(s_j) \tag{10}$$

The weights $w(\mu, s_j)$ are then computed as follows:

$$w(\mu, s_j) = \frac{1}{Z_j} e^{-\|\mu - s_j\|_2^2} \tag{11}$$

where Z_j is a normalization constant ensuring that $\sum_{s_j \in S_{CR}} w(\mu, s_j)(s_j) = 1$. Compared to arithmetic means, the weighted average gives more weights to frequently used successful crossover factors and thus maintain an optimal crossover range.

Similar to [32], at each generation g, the mutation factor F_i of each individual $\boldsymbol{x}_i$ is independently generated according to a Cauchy distribution with location parameter μ_F and scale parameter 0.1.

$$F_i = randc_i(\mu_F, 0.1) \tag{12}$$

When the mutation factors are highly concentrated around a certain value, then the greedy mutation strategies (such as DE/best, and DE/current-to-pbest) tend toward premature convergence. The Cauchy distribution helps to diversify the mutation factors and thus avoid premature convergence very effectively compared to normal distribution. At a particular generation, g the set of all successful mutation factors is represented by S_F. The location parameter μ_F of the Cauchy distribution is initialized to be 0.5 and then updated at the end of each generation as:

$$\mu_F = (1 - c) \cdot \mu_F + c \cdot mean_L(S_F) \tag{13}$$

where $mean_L(\cdot)$ is the Lehmer mean

$$mean_L(S_F) = \frac{\sum_{F \in S_F} F^2}{\sum_{F \in S_F} F} \tag{14}$$

The Lehmer mean gives more weights to larger successful mutation factors. Therefore, Lehmer mean is helpful to propagate larger mutation factors to improve the progress rate.

3.2 Objective function formulation

The objective function or the cluster validity index (CVI) plays an important role in clustering and segmentation. It checks whether the provided clustering $(U; V)$ of X reflects the original structure of the data or not. Clustering algorithms need c, the number of clusters, to be known in advanced. Due to unsupervised nature of clustering, the user does not have

prior knowledge about c. A reliable cluster validity index (CVI) should take into account both compactness and separation for the c-partition of U. If only the former is considered then the best partition consists of as many clusters as that of points. On the other hand, if only the latter is considered, the best partition consists of a single cluster. For the accurate and reliable segmentation, it is important to combine both measures in the CVI. But the separation generally finds distances between centroids of different clusters and are insufficient to represent the geometrical structure of the data.

In this paper, we use an aggregation functions, for the cluster validity, that map a collection of values $\boldsymbol{u}$ in [0,1] to a value in [0,1], formally a vector $\boldsymbol{u} = (u_1; \ldots; u_c)^t \longmapsto A(u)$. The norms (t-norms) and conorms (t-conorms) are the frequently used aggregation operators in the class of triangular norms. They are used to implement conjunctive and disjunctive operators in field of fuzzy logic and fuzzy set theory. A t-norm $\top$ and t-conorm $\bot$ are commutative, associative and monotone functions having 1 and 0, respectively, for the neutral elements [11].

If $\boldsymbol{v}_1, \boldsymbol{v}_2,\ldots, \boldsymbol{v}_c$ denote clusters' centers, then the membership u_{ij}, $i = 1, 2, \ldots, c$ and $j = 1, 2, \ldots, n$, can be computed as:

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} (\frac{D(\boldsymbol{v}_i, \boldsymbol{x}_j)}{D(\boldsymbol{v}_k, \boldsymbol{x}_j)})^{\frac{1}{m-1}}}, \; for \; 1 \leq i \leq c, \; 1 \leq j \leq n \tag{15}$$

where $D(\boldsymbol{v}_i, \boldsymbol{x}_j)$ is the Euclidean distance between point $\boldsymbol{x}_j$ and cluster center $\boldsymbol{v}_i$, m is the weighting coefficient. While computing u_{ij} using Eq. (15), if $D(\boldsymbol{v}_i, \boldsymbol{x}_j)$ is equal to zero for some k, then u_{ij} is set to zero for all $i = 1, 2, \ldots, c$, $i \neq k$, and u_{ij} is set equal to one.

In images, neighboring pixels are highly correlated [5]. In other words, neighboring pixels possess similar features and there is high probability that they belong to the same cluster. A spatial function is defined in Eq. (16) to exploit the idea of spatial information [5].

$$h_{ij} = \sum_{k \in NB(\boldsymbol{x}_j)} u_{ik} \tag{16}$$

where $NB(\boldsymbol{x}_j)$ represents a square window centered on pixel $\boldsymbol{x}_j$ in the spatial domain. The spatial function h_{ij} represents the probability that pixel $\boldsymbol{x}_j$ belongs to ith cluster. The spatial information can be incorporated in membership function as:

$$u'_{ij} = \frac{u_{ij}^p h_{ij}^q}{\sum_{k=1}^{c} u_{kj}^p h_{kj}^q} \tag{17}$$

where p and q are parameters to control the relative importance of both functions. The centers can be updated using (18).

$$\boldsymbol{v}_i = \frac{\sum_{j=1}^{n} (u'_{ij})^m \boldsymbol{x}_j}{\sum_{j=1}^{n} (u'_{ij})^m}, \; 1 \leq i \leq c \tag{18}$$

The above equations gives the fuzzy clustering $(U; V)$ of data X, where u_{ij} defines the similarity of a data point $\boldsymbol{x}_j$ to a center $\boldsymbol{v}_i$. Using the membership vector $\boldsymbol{u}'_j = (u'_{1j}; \ldots; u'_{cj})^t$ (j^{th} column of U), to assign a particular data point to the most matching group the standard t-conorm (max operator) is commonly used. However, the data point $\boldsymbol{x}_j$ may satisfy more than one group description (i.e., lower values interact with the greatest) then using the exclusive partitioning (only the t-conorm) is not efficient. The aggregate value of $\boldsymbol{u}'_j$ is adequate to assess that how much a particular point $\boldsymbol{x}_j$ belongs to several other clusters. We use the fuzzy OR operator (fOR-ℓ for short) defined in [20] to evaluates the degrees of similarity at a given

order (ℓ). Let ρ be the power set of $C = 1, 2, \ldots, c$ and $\rho_\ell = A \in |A| = \ell$ where $|A|$ denotes the cardinality of subset A, then the fOR-(ℓ) associates a single value $\perp^\ell(\boldsymbol{u}'_j) \in [0; 1]$ to $\boldsymbol{u}'_j$ as given by the following equation:

$$\overset{\ell}{\perp}(\boldsymbol{u}'_j) = \underset{i=1,2,\ldots,c}{\overset{\ell}{\perp}} u'_{ij} = \underset{A \in \rho_{l-1}}{\top} \left(\underset{j \in \backslash A}{\perp} \right) u'_{ij} \tag{19}$$

It must be viewed as some kind of generalization of the ℓth highest value, $\ell \in C$. The standard t-norms, $\overset{\ell}{\perp}(\boldsymbol{u}'_j)$ uses the ℓth highest element of u'_j [20].

In this paper, we have used two measures, the fuzzy overlap and fuzzy separation [4], to find the relation of a point $\boldsymbol{x}_j$ to the clusters and overcome the drawbacks of compactness and separation measures. The fuzzy overlap measure evaluates the degree of overlap of a specified number (ℓ) of fuzzy clusters, and a fuzzy separation measure indicates the degree of overlap of the most probable cluster, i.e., the one corresponding to the highest membership degree, with respect to $c - 1$ other clusters. A low value of this latter measure will denote a large separation of the most probable cluster from the others. For each point, $\boldsymbol{x}_j$ of X, described by its membership vector $\boldsymbol{u}'$, the overlap measure between ℓ fuzzy clusters can be obtained by (20). A combination of ℓ-order overlap degrees for $\boldsymbol{x}_j$ is obtained by successive computing of $\overset{\ell}{\perp}(\boldsymbol{u}'_j)$ for different values of ℓ. To determine the over all degree of overlap for a given point $\boldsymbol{x}_j$, the order that induces high overlap is find out. The most satisfied order(s) can be quantified by the fuzzy disjunction of the ℓ-order overlap measures ($\ell = 2; c$), so the overall overlap measure of a point $\boldsymbol{x}_j$ is given by:

$$O_\perp(\boldsymbol{u}'_j; c) = \underset{\ell=2,c}{\overset{1}{\perp}} \left(\underset{i=1,c}{\overset{\ell}{\perp}} u'_{ij} \right) \tag{20}$$

The inter-cluster separation plays an important role in cluster validity. For this purpose, the fuzzy separation [4] of each point $\boldsymbol{x}_j$ with ($\boldsymbol{u}'_j$) is find out. It is the overlap measure within one fuzzy cluster, i.e., its separation from the other fuzzy clusters, since $\boldsymbol{u}'_j$ sums to one. This aggregation corresponds to the fuzzy disjunction of membership degrees for a given point $\boldsymbol{x}_j$, which selects the most probable cluster. The fuzzy separation of $\boldsymbol{x}_j$ with respect to the c clusters is given as:

$$S_\perp(\boldsymbol{u}'_j; c) = \overset{1}{\perp} \left(\underbrace{\underset{i=1,c}{\overset{1}{\perp}} u'_{ij}, \ldots, \underset{i=1,c}{\overset{1}{\perp}} u'_{ij}}_{c-1 \ times} \right) \tag{21}$$

The highest membership degree of a point $\boldsymbol{x}_j$ corresponds to the small value of overlapping degree $O_\perp(\boldsymbol{u}'_j; c)$ and a large value of the separation $S_\perp(\boldsymbol{u}'_j; c)$. It means that if a point $\boldsymbol{x}_j$ has a small $O_\perp(\boldsymbol{u}'_j; c)$ and large $S_\perp(\boldsymbol{u}'_j; c)$, it will be well-separated and not overlapping cluster. Then, the ℓ-order overlap and separation index (OSI) taking values in [0; 1] as the average value of the ratios of both measures:

$$OSI_\perp(c) = \frac{1}{n} \sum_{j=1}^{n} \frac{O_\perp(\boldsymbol{u}'_j; c)}{S_\perp(\boldsymbol{u}'_j; c)} \tag{22}$$

Running MoADE for $c \in [c_{\min}; c_{\max}]$ and selecting that value of c, which best minimizes Eq. (22). The $c_{\min}$ and $c_{\max}$ are set to 2 and 20, respectively. The small regions less than 100 pixels are merged to its neighboring segments on the basis of similarity to avoid over-segmentation.
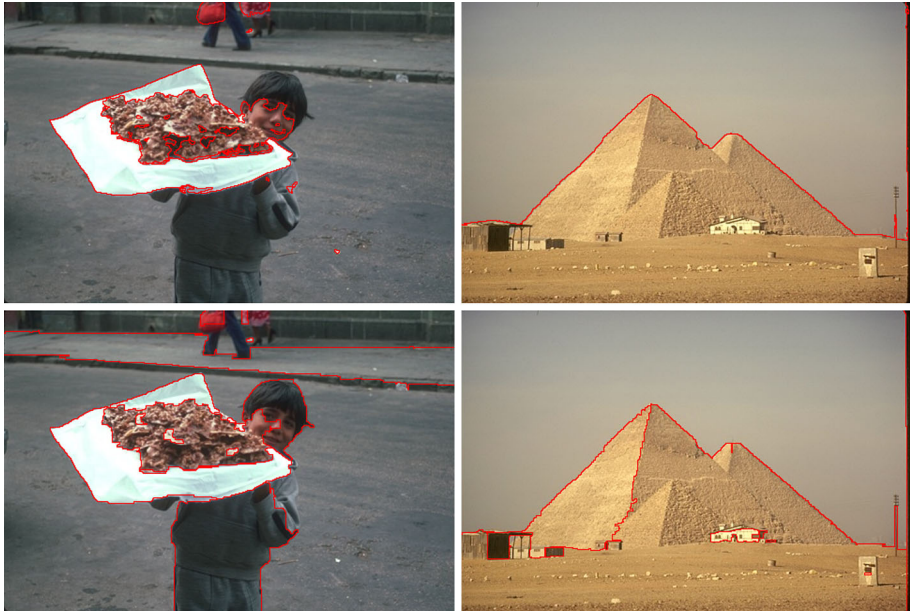
Fig. 1 Results of ADE [32] and MoADE after 100 iterations. The *first row* contains the results of ADE, while the *second row* contains the results of proposed MoADE, respectively

MoADE explores the underlying search space efficiently using the opposition-based learning. Besides to diversify the search process, MoADE also speeds up the convergence considerably. Figure 1 shows the results of ADE [32] and proposes MoADE after 100 iterations by optimizing the new objective function. First row in Fig. 1 shows the results of ADE, while the second row shows the results of MoADE, respectively. It is clear from the figure that ADE does not detect the body of the boy, while MoADE performs better in this case. Similarly, for the pyramid image, (second column of Fig. 1) ADE fails to detect the black-gate-shaped and white-building-shaped objects. But the proposed MoADE performs better to detect the underlying objects accurately. Note that, ADE uses random procedure for initialization and is unable to uniformly initialize the search space to the productive regions. Once the search space is initialized to the unproductive regions, then the algorithm converges to local optimas'. On the other hand, the proposed MoADE uses the opposition-based learning strategy to initialize and evolve the population. By this way, best individuals from the union of current population and its opposition are selected for new generation in order to get global best solution.

4 Experimental results

Experimentations have been carried out on images taken from the Berkeley image segmentation database (BSD) [19]. The Berkeley image segmentation database (BSD) contain 300 natural images of sizes (321 × 481) or (481 × 321). The results of the proposed algorithm are compared with three state-of-the-art image segmentation algorithms: segmentation via lossy data compression (LDC) [31], efficient graph-based image segmentation (EG) [8], and normalized cuts and image segmentation (NC) [26].

The region number in NC is set to 20, which is the average number of segments marked by the human subjects in each image. In EG algorithm, the Gaussian smoothing parameter $\sigma = 0.8$, the threshold value $k = 300$, and the minimal region = 50 pixels are set, as described in [8]. Source codes of the all algorithms are downloaded from their respective web sites.

4.1 Qualitative comparisons

In this section, the segmentation results of the MoADE, NC, EG, and LDC approaches are evaluated visually. Eight images: shrub (481×321), church1 (481×321), eagle1 (481×321), eagle2 (481×321), texture (321×481), building (481×321), lion (481×321), and church2 (481×321) are selected from the Berkeley image segmentation database for qualitative comparison. Figure 2 shows the results of proposed MoADE, NC, EG, and LDC. For the Shrub image, the result of MoADE is closely matching the human perception, i.e., the shrub and non-shrub areas are separated in a best way, whereas the NC divides the uniform background into unnecessary segments. The performance of EG and LDC is not good for the shrub image because they make over-segmentation. In the case of eagle1 image, the proposed algorithm outperforms the other state-of-the-art algorithms. LDC performs good for eagle1 image compared to NC, which divide the uniform background into multiple segments, and EG, which fails to detect the actual boundaries of the object. Note that, the proposed MoADE also outperforms the other techniques in the case of church1 image. For eagle2 image, the proposed technique performs better by accurately separating the tree and eagle from the background region, but the performance of the other three methods are not good enough. Similar to the human perception, the proposed MoADE correctly segment the texture image by separating the two different textures. For the same image, the LDC performs good enough compared to NC, which divides the same texture into different segments, and EG, which makes over-segmentation, but fails to detect boundaries of the texture correctly. It is clear from the figure that the proposed algorithm also outperforms the other state-of-the-art methods in the case of building, lion, and church2 images.

NC tends to partition an image into regions of similar sizes, resulting in the region boundaries different from the real edges. EG gives strongly over-segmented results. LDC usually obtains better visual results than NC, and EG, but it often fails to find real edges and creates strongly over-segmented regions. It is easy to judge from the results that the proposed method performs better than the other state-of-the-art algorithms by generating boundaries which match the real edges of the objects.

4.2 Quantitative comparisons

Quantitative comparisons are also important to evaluate the performance of an algorithm objectively. Region differencing [13] measures how much a particular segmentation can be viewed as a refinement of other. The boundary matching [9] compares the results of an algorithm and that of human subjects by finding the average displacement error of the boundary pixels. Region differencing and boundary matching are not sufficiently good to objectively evaluate segmentation results [28]. If a single pixel is considered a separate segment, then region differencing and boundary matching will have best scores. Thus, the over-segmented results may be ranked good, which are far away from human perception of image understanding.

In this paper, we have used probabilistic rand index (PRI) [28] and variation of information (VoI) [21] to objectively evaluate the segmentation results. Consider a set of ground truths, labeled by K persons, $S_1, S_2, \ldots, S_K$, of an image consisting of N pixels. Let S_{test} be the
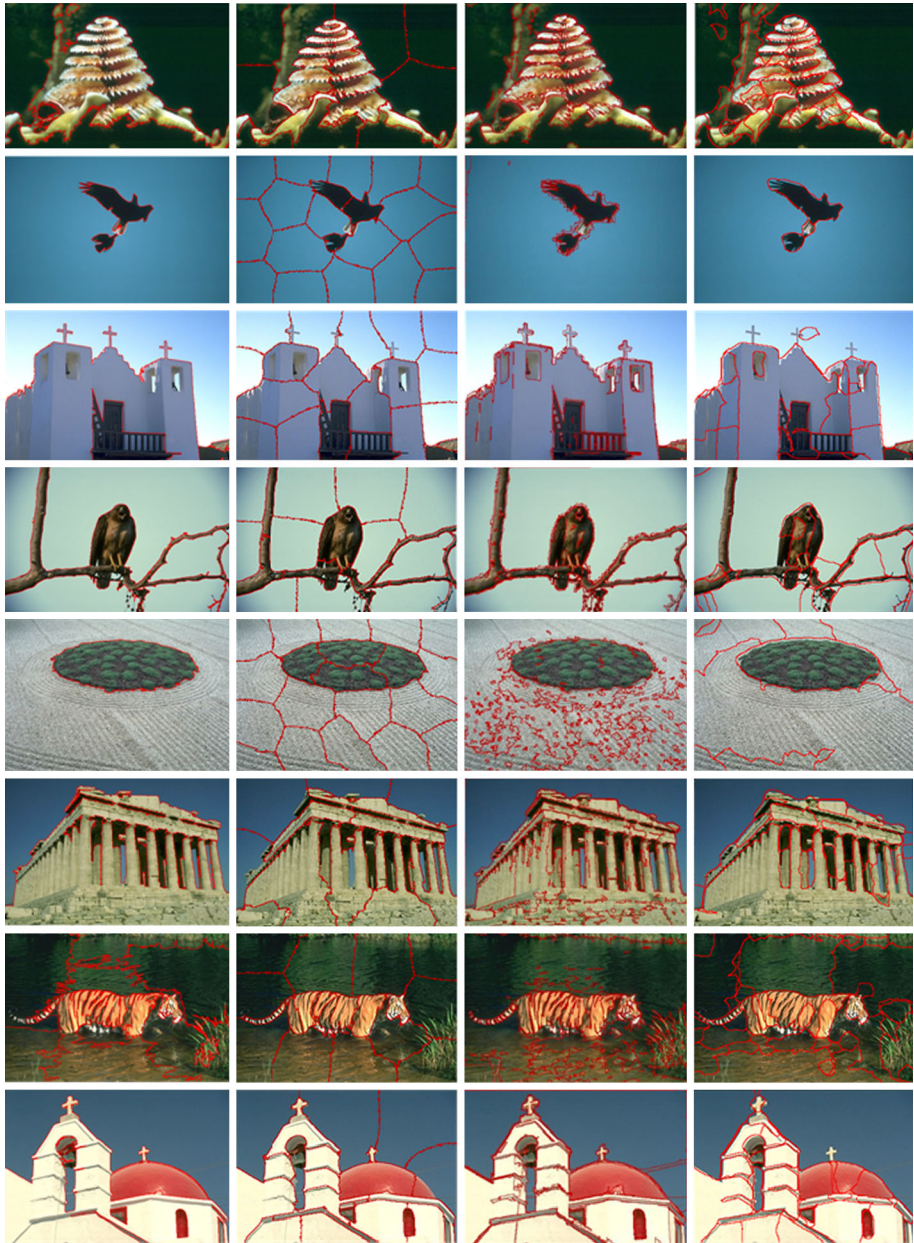
Fig. 2 Segmentation results on some of the images from the Berkeley segmentation dataset and benchmark. The *first*, *second*, *third*, and *fourth* columns contain the MoADE, NC, EG, and LDC segmented images, respectively

segmentation result to be compared with the ground truths. Then, the PRI value is defined as:
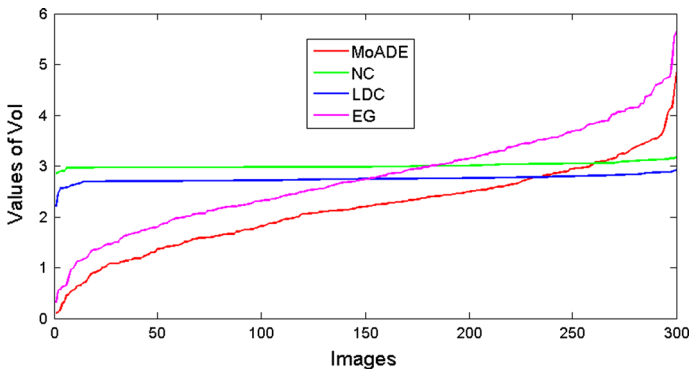
$$\mathrm{PRI}(S_{test}, \{S_k\}) = \frac{1}{\binom{N}{2}} \sum_{p<q} [\bar{p}_{pq}^{c_{pq}} (1 - \bar{p}_{pq})^{1-c_{pq}}] \tag{23}$$

Table 1 Average values of PRI and VoI for the four algorithms on 300 BSD images

Measures	Human	MoADE	NC	EG	LDC
PRI	0.8961	0.7953	0.7212	0.7857	0.7529
VoI	0.9219	1.9052	2.9893	2.7589	2.0288
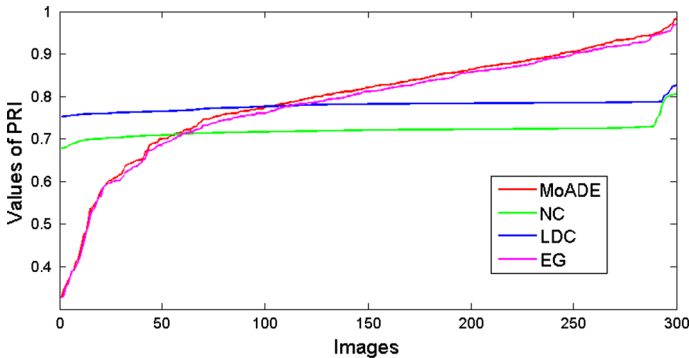
Fig. 3 PRI values achieved on individual images by the four algorithms. The values are plotted in increasing order

Fig. 4 VoI values achieved on individual images by the four algorithms. The values are plotted in increasing order

where (p, q) is a pixel pair in image, $c_{pq} = T(l_p^{S_{test}} = l_q^{S_{test}})$ denotes the event of a pair of pixel p and q having the same label in the test result S_{test}, and $\bar{p}_{pq} = 1/K \sum_{k=1}^{K} T(l_p^{S_k} = l_q^{S_k})$ is regarded the probability of p and q having the same labels. The value of PRI lies in the range [0,1], and the larger the better. The variation of information (VoI) is defined as:

$$\text{VoI}(S_{test}, \{S_k\}) = \frac{1}{K} \sum_k [H(S_{test}) + H(S_k) - 2I(S_{test}, S_k)] \quad (24)$$

where H and I denote the entropy and the mutual information, respectively. The detailed definitions of H and I can be found in [21]. VoI is an information-based measure, which determines how much one segmentation gives information about other. The value of VoI falls in $[0, \infty)$, and the smaller, the better.

Table 1 shows the average values of PRI and VoI for the four algorithms on 300 images of the Berkeley segmentation database. In this table, the second column shows the average PRI and VoI values between different human subjects, which are the best scores. It can be observed from the table that the proposed algorithm performs better than the other three state-of-the-art algorithms in terms of VoI and PRI scores. The average PRI value of the EG is close to proposed algorithm, but the average VoI value is much greater. Similarly, the average VoI of the LDC is close to proposed MoADE and better than that of the other two algorithms, but the PRI value is not better. Figures 3 and 4 graphically represent the PRI and VoI scores, respectively, for the four algorithms on 300 BSD images. It is clear from the curves in Figs. 3 and 4 that the proposed algorithm performs better compared to other state-of-the-art methods.

5 Conclusion

In this paper, we presented a novel image segmentation algorithm. It is the combination of spatial fuzzy c-mean (sFCM), which looks for similarity in the local neighborhood among the pixels, and differential evolution (DE), which diversifies the search space to find the optimal solution. The ratio of fuzzy overlap and fuzzy separation is used as a cluster validity index in order to find out the optimal number of clusters automatically. The results of the proposed algorithm are compared with three state-of-the-art image segmentation algorithms. The qualitative and quantitative results demonstrate that the proposed algorithm performs better compared to the given state-of-the-art image segmentation algorithms. In the future, we are going to construct a unified matrix, from different feature spaces. i.e., color and texture, in such away that the important information of each feature space is to be preserved. Then, the segmentation will be performed on a joint or combined search space.

References

1. Ahmed MN, Yamany SM, Mohamed N, Farag AA, Moriarty T (2002) A modified fuzzy C-means algorithm for bias field estimation and segmentation of MRI data. IEEE Trans Med Imaging 21(3):193–199
2. Besag J (1986) On the statistical analysis of dirty pictures. J R Stat Soc B 48:259–302
3. Beucher S, Meyer F (1993) The morphological approach to segmentation: the watershed transformation. Math Morphol Image Process 34:433–481
4. Capitaine HL, Frelicot C (2011) A cluster-validity index combining an overlap measure and a separation measure based on fuzzy-aggregation operators. IEEE Trans Fuzzy Syst 19(03):580–588
5. Chuang KS (2006) Fuzzy C-means clustering with spatial information for image segmentation. Comput Med Imaging Graph 30:9–15
6. Cohen L (1991) On active contour models and balloons. CVGIP Image Underst 53(2):211–218
7. Duda R, Hart P, Stork D (2001) Fast discovery of association rules. Pattern Classification (2nd ed). Wiley, Hoboken, NJ
8. Felzenszwalb P, Huttenlocher D (2004) Efficient graph-based image segmentation. Int J Comput Vis 59(2):167–181
9. Freixenet J, Munoz X, Raba D, Marti J Cufi X (2002) Yet another survey on image segmentation: region and boundary information integration. In: Proceedings of European conference on computer vision (ECCV 2002), Copenhagen, Denmark, May 2831, 2002. Lecture Notes in Computer Science 2352. Springer, Berlin, pp 408–422
10. Kass M, Witkin A, Terzopoulos D (1988) Snakes: active contour models. Int J Comput Vis 1(4):321–331
11. Klement E, Mesiar R (2005) Logical, algebraic, analytic, and probabilistic aspects of triangular norms. Elsevier, Amsterdam

12. Kolmogorov V, Zabih R (2004) What energy functions can be minimized via graph cuts? IEEE Trans Pattern Anal Mach Intell 26(2):147–159
13. Kov YB, Jolly M (2001) Interactive graph cuts for optimal boundary region segmentation of objects in $N - D$ images. In: Proceedings of eighth IEEE international conference on computer vision, 2001 (ICCV 2001), Jully 714, 2002. 1, pp 105–112
14. Li Y, Sun J, Tang CK, Shum HY (2004) Lazy snapping. ACM Trans Graph 23(3):303–308
15. Li BN, Chui CK, Ong SH (2011) Integrating spatial fuzzy clustering with level set for autamated medical image segmentation. Comput Biol Med 41:1–10
16. Liu MY, Tuzel O, Ramalingam S, Chellappa R (2011) Entropy rate superpixel segmentation. In: Proceedings of IEEE international conference on computer vision, (ICCV 2011), 2011, pp 2097–2104
17. Loo PK, Tan CL (2004) Adaptive region growing color segmentation for text using Irregular pyramid. In: Proceedings of 6th international workshop international workshop on document analysis systems, Florence, Italy, 2004, September 8–10. Lecture Notes in Computer Science 3163. Springer, Berlin, pp 264–275
18. Lucchese L, Mitra SK (2001), Color image segmentation: a state of the art survey. In: Proceedings of the Indian National Science Academy on image processing. Vis. Pattern Recognition vol 2, pp 207–221
19. Martin D, Fowlkes C, Tal D, Malik J (2001) A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of eighth IEEE international conference on computer vision, 2001 (ICCV 2001), Jully 714, 2001. 2, pp 416–423
20. Mascarilla L, Berthier M, Frelicot C (2008) A k-order fuzzy or operator for pattern classification with k-order ambiguity rejection. Fuzzy Sets Systems 159(15):2011–2029
21. Meila M (2005) Comparing clusterings-An axiomatic view. In: ICML '05 Proceedings of the 22nd international conference on Machine learning, New York, NY, USA, pp 577–584
22. Mortensen E, Barrett W (1998) Interactive segmentation with intelligent scissors. Graph Models Image Process 60(5):349–384
23. Ninga J, Zhanga L, Zhanga D, Wub C (2010) Interactive image segmentation by maximal similarity based region merging. Pattern Recognit 43:445–456
24. Rahnamayan S, Tizhoosh HR, Salama MMA (2008) Opposition-based differential evolution. IEEE Trans Evolut Comput 12(2):64–79
25. Rother C, Kolmogorov V, Blake A (2004) Opposition-based differential evolution. ACM Trans Graphs 23(3):309–314
26. Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22(8):888–905
27. Tan KS, Isa NAM (2011) Color image segmentation using histogram thresholding fuzzy C-means hybrid approach. Pattern Recognit 44:1–15
28. Unnikrishnan R, Pantofaru C, Hebert M (2007) Toward objective evaluation of image segmentation algorithms. IEEE Trans Pattern Anal Mach Intell 29(6):929–944
29. Vincent L, Soille P (1999) Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Trans Pattern Anal Mach Intell 13(6):583–598
30. Xu C, Prince J (1998) Snakes, shapes, and gradient vector flow. IEEE Trans Image Process 7(3):359–369
31. Yang AY, Wright J, Sastry SS, Ma Y (2008), Unsupervised segmentation of natural images via lossy data compression. Comput Vis Image Underst 13(5):212–225
32. Zhang J, Sanderson AC (2009) ADE: adaptive differential evolution with optional external archive. IEEE Trans Evolut Comput 13(5):945–958

Ahmad Khan received the M.Sc. degree in computer science from the University of Peshawar, Pakistan, the M.S. degree in computer science from the National University of Computer and Emerging Sciences Islamabad, Pakistan and is currently pursuing the Ph.D. degree at the National University of Computer and Emerging Sciences Islamabad, Pakistan. From June 2012 to December 2012, he joined the Department of Electronic and Electrical Engineering University of Sheffield, UK, as is visiting Scholar. His research interests include computer vision and machine learning.

M. Arfan Jaffar received his B.Sc degree from Bahauddin Zakariya University Multan, Pakistan, in 2000 and got distinction (Gold Medal). He later received M.Sc. degree in computer science in 2003 from Quaid-e-Azam University Islamabad, Pakistan. Then, he earned his PhD degree in computer science in 2009 from National University of Computer and Emerging Sciences, NU-FAST, Islamabad, Pakistan. Currently he is working as a Assistant Professor at the College of Computer and Information Sciences, Imam Muhammad Ibn Saud Islamic University (IMSIU), Riyadh, Saudi Arabia. His research interests include image processing and computational intelligence.

Ling Shao received the B.Eng. degree in electronic engineering from the University of Science and Technology of China (USTC), the M.Sc. degree in Medical Image Analysis, and the Ph.D. (D.Phil.) degree in Computer Vision at the Robotics Research Group from the University of Oxford. Dr. Ling Shao is currently a Senior Lecturer (Associate Professor) in the Department of electronic and electrical engineering at the University of Sheffield. Before joining Sheffield University, he worked for 4 years as a Senior Scientist in Philips Research, The Netherlands. His research interests include computer vision, pattern recognition, and video processing. He has published over 80 academic papers in refereed journals and conference proceedings and over 10 awarded patents and patent applications. Ling Shao is an associate editor of IEEE Transactions on Systems, Man and Cybernetics Part B: Cybernetics, the International Journal of Image and Graphics, the EURASIP Journal on Advances in Signal Processing, and Neurocomputing and has edited several special issues for journals of IEEE, Elsevier and Springer. He has organized several workshops with top conferences, such as ICCV, ACM Multimedia, and ECCV. He has been serving as Program Committee member for many international conferences, including CVPR, ECCV, ICIP, ICASSP, ICME, ICMR, ACM MM, CIVR, BMVC. He is a Fellow of the British Computer Society and a Senior Member of the IEEE.