



A comprehensive survey of image segmentation: clustering methods, performance parameters, and benchmark datasets

Himanshu Mittal¹ · Avinash Chandra Pandey¹ · Mukesh Saraswat¹ · Sumit Kumar² ·
Raju Pal¹ · Garv Modwel³

Received: 10 June 2020 / Revised: 7 January 2021 / Accepted: 21 January 2021 /

Published online: 9 February 2021

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC part of Springer Nature 2021

Abstract

Image segmentation is an essential phase of computer vision in which useful information is extracted from an image that can range from finding objects while moving across a room to detect abnormalities in a medical image. As image pixels are generally unlabelled, the commonly used approach for the same is clustering. This paper reviews various existing clustering based image segmentation methods. Two main clustering methods have been surveyed, namely hierarchical and partitional based clustering methods. As partitional clustering is computationally better, further study is done in the perspective of methods belonging to this class. Further, literature bifurcates the partitional based clustering methods into three categories, namely K-means based methods, histogram-based methods, and meta-heuristic based methods. The survey of various performance parameters for the quantitative evaluation of segmentation results is also included. Further, the publicly available benchmark datasets for image-segmentation are briefed.

Keywords Image segmentation · Clustering methods · Performance parameters · Benchmark datasets

1 Introduction

“A picture is worth a thousand words” is a famous idiom which signifies that processing an image may relieve more information than processing the textual data. In computer vision, image segmentation is the prime research area which corresponds to partitioning of

✉ Raju Pal
raju3131.pal@gmail.com

¹ Jaypee Institute of Information Technology, Noida, Uttar Pradesh, India

² Amity University, Noida, Uttar Pradesh, India

³ Valeo India Private Limited, Chennai, Tamil Nadu, India

an image into its constituent objects or region of interests (ROI). Generally, it assembles the image pixels into similar regions. It is a pre-processing phase of many image-based applications like biometric identification, medical imaging, object detection and classification, and pattern recognition [91]. Some of the prominent applications are as follows.

- **Content-based image retrieval:** It corresponds to the searching of query-relevant digital images from large databases. The retrieval results are obtained as per the contents of the query image. To extract the contents from an image, image segmentation is performed.
- **Machine vision:** It is the image-based technology for robotic inspection and analysis, especially at the industrial level. Here, segmentation extracts the information from the captured image related to a machine or processed material.
- **Medical imaging:** Today, image segmentation helps medical science in a number of ways from medical diagnosis to medical procedures. Some of the examples include segmentation of tumors for locating them, segmenting tissue to measure the corresponding volumes, and segmentation of cells for performing various digital pathological tasks like cell count, nuclei classification and many others.
- **Object recognition and detection:** Object recognition and detection is an important application of computer vision. Here, an object may be referred to as a pedestrian or a face or some aerial objects like roads, forests, crops, etc. This application is indispensable to image segmentation as the extraction of the indented object from the image is priorly required.
- **Video surveillance:** In this, the video camera captures the movements of the region of interests and analysis them to perform an indented task such as identification of the action being performed in the captured video or controlling the traffic movement, counting the number of objects and many more. To perform the analysis, segmentation of the region of interest is foremost required.

Though segmenting an image into the constituent ROI may end up as a trivial task for humans, it is relatively complex from the perspective of the computer vision. There are

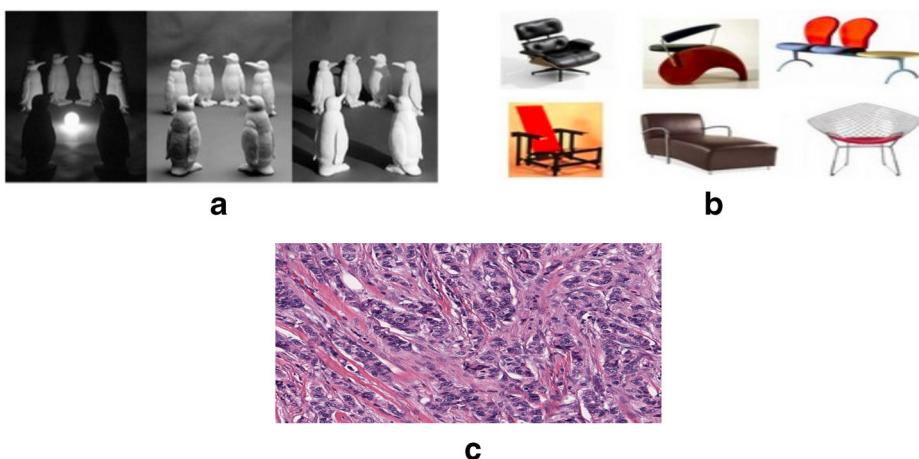


Fig. 1 Challenges in image segmentation **a** Illumination variation [2] **b** Intra-class variation [2] **c** Background complexity [3]

number of challenges which may affects the performance of an image segmentation method. Figure 1 depicts three major challenges of image segmentation which are discussed below.

- **Illumination variation:** It is a fundamental problem in image segmentation and has severe effects on pixels. This variation occurs due to the different lighting conditions during the image capturing. Figure 1a shows an image that is captured in different illumination conditions. It can be observed that the corresponding pixels in each image contain varying intensity values which pose difficulties in image segmentation.
- **Intra-class variation:** One of the major issues in this field is the existence of the region of interest in a number of different forms or appearances. Figure 1b depicts an example of chairs that are shown in different shapes, each having a different appearance. Such intra-class variation often makes the segmentation procedure difficult. Thus, a segmentation method should be invariant to such kind of variations.
- **Background complexity:** Image with a complex background is a major challenge. Segmenting an image as the region of interests may mingle with the complex environment and constraints. Figure 1c illustrates an example of such image which corresponds to H&E stained breast cancer histology image. The dark blue color regions in the image represent the nuclei region which is generally defined as the region of interests in histopathological applications like nuclei count or cancer detection. It can be observed that the background is too complex due to which the nuclei regions do not have clearly defined boundaries. Therefore, such background complexities degrade the performance of segmentation methods.

Further, the essence of an image segmentation is to represent an image with a few significant segments instead of thousands of pixels. Moreover, image segmentation may be viewed as a clustering approach in which the pixels, that are satisfying a criterion, are grouped into a cluster while dissatisfying pixels are placed in different groups. To exemplify this, consider the images in Fig. 2. The first image consists of some animals on the field. To extract the animals from the background, the ideal result would be to group all the pixels belonging to the animals into the same cluster while background pixels into another cluster as presented in Fig. 2b. However, the pixel labelling is unknown. Moreover, it can be observed that first image presents a complex scenario in terms of different color intensities and shapes of each animal which correspond to illumination and intra-class variations respectively. To mitigate these challenges and learning from such unlabelled data, the common approach is to group the data based on certain similarity or dissimilarity measures followed by labelling of each group. This approach of grouping the data is generally termed as clustering.

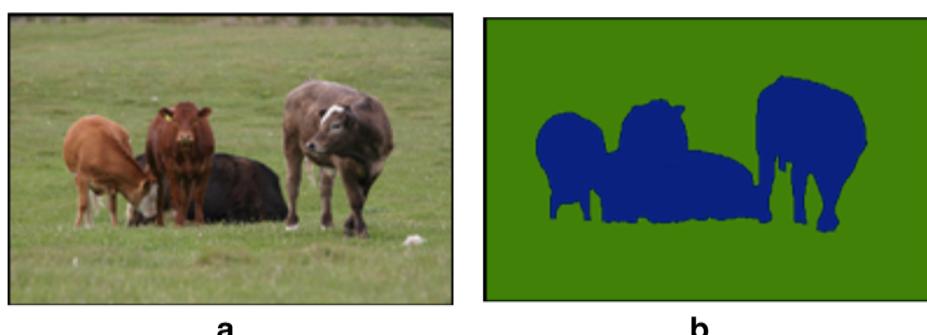


Fig. 2 Example of clustering based image segmentation **a** Original image **b** Segmented image

Generally, clustering has been used in different areas of real-world applications like market analysis, social network analysis, online query search, recommendation system, and image segmentation [66]. The main objective of a clustering method is to classify the unlabelled pixels into homogeneous groups that have maximum resemblance, i.e. to achieve maximum similarity within the clusters and minimum dissimilarity among the clusters. Mathematically, the clustering procedure on an image (X) of size $(m \times n)$, defined over d -dimensions, generates K clusters $\{C_1, C_2, \dots, C_K\}$ subject to the following conditions:

- $C_i \neq \emptyset$, for $i = 1, 2, \dots, K$
- $C_i \cap C_j = \emptyset$, for i and $j = 1, 2, \dots, K$ and $i \neq j$
- $\cup_{i=1}^K C_i = X$

The first condition ensures that there will be at least one pixel in every formed cluster. The next condition implies that all the formed clusters will be mutually exclusive, i.e. a pixel will not be assigned to two clusters. The last condition states that the data values assigned to all the clusters will represent the complete image.

In literature, there are number of clustering algorithms for image segmentation. However, there is no precise definition of a cluster and different clustering methods have defined their own techniques to group the data. Therefore, based on the cluster formation, the clustering methods may broadly be classified in two main categories, namely hierarchical and partitional [85]. A taxonomy of the clustering methods is presented in Fig. 3. The following section presents each category of clustering with respect to image segmentation.

2 Hierarchical clustering

In hierarchical clustering, the grouping of data is performed at different similarity levels which is schematized by a tree-like structure termed as a dendrogram. Generally, the hierarchical splitting follows two approaches, namely divisive and agglomerative [66]. Figure 4 depicts an example of the dendrogram along with the hierarchical approaches.

Table 1 summaries the methods belonging to these two sub-categories. In divisive clustering, recursive hierarchical data splitting is performed in a top-down fashion to generate

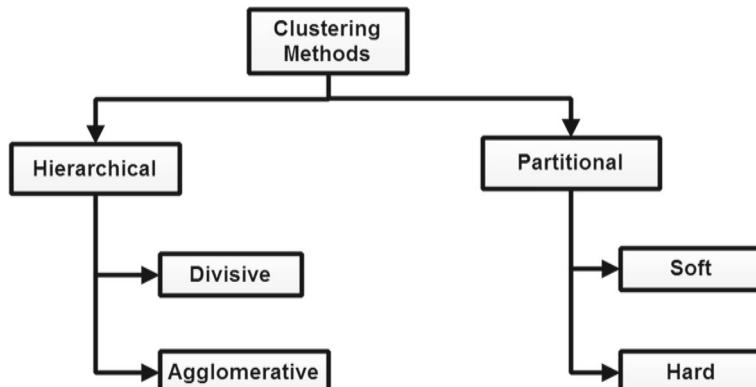


Fig. 3 Classification of clustering based image segmentation methods [85]

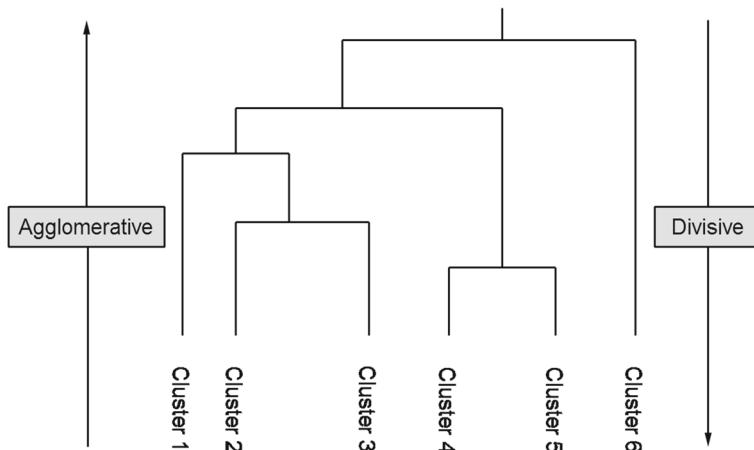


Fig. 4 Types of hierarchical clustering [66]

the clusters. All the data items belong to a single cluster initially. This single cluster is further split into smaller clusters until a termination criteria is satisfied or until each data item forms its own cluster. Divisive clustering (DIVCLUS-T) [17] and divisive hierarchical clustering with diameter criterion (DHCDC) [31] are popular methods of this category. On the other side, the agglomerative clustering is performed in the bottom-up fashion where data points are merged hierarchically to produce clusters. Initially, each data item defines itself as a cluster which is further merged into bigger clusters, until a termination criteria is met or until a single cluster is formed consisting of all the data items. Some popular agglomerative methods are balanced iterative reducing and clustering using hierarchies (BIRCH) [94], clustering using representatives (CURE) [32], and chameleon [41].

In general, divisive clustering is more complex than the agglomerative approach, as it partitions the data until each cluster contains a single data item. The divisive clustering will be computationally efficient if the each cluster is not partitioned to individual data leaves. The time complexity of a naive agglomerative clustering is $O(n^3)$ which can be reduced to $O(n^2)$ using optimization algorithms. On the contrary, the time complexity of divisive clustering is $\Omega(n^2)$ [66]. Moreover, divisive clustering is also more accurate since it considers the global distribution of data while partitioning data in top-level partitioning.

Table 1 Hierarchical clustering methods for image segmentation

Sub-categories	Methods	Remarks
Divisive	DIVCLUS-T [17], DHCDC [31], DITFC [28]	Nested grouping at different similarity levels; Number of clusters needed to be preset.
Agglomerative	SLINK [68], BIRCH [94], CURE [32], Chameleon [41]	Sensitive to noise and outliers; Poor performance on overlapping clusters; Computationally expensive for large dataset.

The pseudocode of the divisive and agglomerative clustering approaches are presented in Algorithms 1 and 2 respectively.

Algorithm 1 Agglomerative clustering.

```
Input: Initialize the number of clusters to be formed.  
Output: Clustered data.  
Consider each data point as a cluster;  
while clustering condition is not satisfied do  
    Perform merging of two clusters having minimum inter-cluster distance;  
end while
```

Algorithm 2 Divisive clustering.

```
Input: Initialize the number of clusters to be formed.  
Output: Clustered data.  
Consider all data points as a single cluster;  
while clustering condition is not satisfied do  
    Divide the cluster into two clusters resulting in the largest inter-cluster distance;  
end while
```

In divisive hierarchical clustering algorithms, DIVCLUS-T clustering follows the monothetic bipartitional approach which allows the constructed dendrogram to be considered as a decision tree. This method can handle both categorical and numerical data. In DHCDC, the largest cluster is considered in terms of diameter which is partitioned into the two largest diameter clusters. BIRCH clustering is generally used for very large datasets due to its ability to discover a good clustering result in a single scan of datasets. BIRCH clustering algorithm constructs clustering feature tree to discover clusters. However, BIRCH clustering algorithm can only produce spherical and convex shape clusters. Therefore, as an alternative, CURE clustering algorithm is used. CURE clustering employs random sampling approach to cluster all the data items which are further combined to create the final result. The improved version of CURE is ROCK which is suitable for the data of enumeration type. Chameleon clustering first divides the original data into a smaller sized graph and then merge these small-sized graphs to create the final clusters.

Generally, hierarchical methods employ a greedy approach and do not reconsider a data item again after it has been assigned to a cluster. This results in lacking the capability of correcting the misclassified data item. Therefore, these methods lack robustness, especially in the case of noise and outliers. They do not intend to optimize an objective function while forming the clusters. Moreover, they perform poorly when clusters are overlapping. While generating clusters for a particular problem, knowledge about the number of clusters is required. Moreover, the formation of spherical clusters and the reversal of the hierarchical structure are distorted. Additionally, time complexity is one of the major issues while clustering data through hierarchical approach, especially on high-dimensional dataset like image. The time complexity of hierarchical methods is computationally expensive which has been found to be at least $O(n^2)$ for most of the hierarchical methods [66], where n corresponds to the number of data points. Therefore, to overcome this limitation, partitional clustering methods are preferred which are discussed in the following section.

3 Partitional clustering

The partitional clustering is relatively popular and preferred over the hierarchical clustering, especially for a large dataset, due to its computational efficiency [13]. In this clustering approach, the notion of similarity is used as the measurement parameter. Generally, partitional clustering groups the data items into clusters according to some objective function such that data items in a cluster are more similar than the data items in the other clusters. To achieve this, the similarity of each data item is measured with every cluster. Moreover, in partitional clustering, the general notion of the objective function is the minimization of the within-cluster similarity criteria which is usually computed by using Euclidean distance. The objective function expresses the goodness of each formed cluster and returns the best representation from the generated clusters. Like hierarchical clustering, the number of clusters to be formed needs to be defined priorly. Moreover, methods based on partitional clustering assure that each data item is assigned to a cluster even if it is quite far from the respective cluster centroid. This, sometimes, results in distortion of the cluster shapes or false results, especially in the case of noise or outlier.

Table 2 depicts partitional methods which have been used in various areas, such as image segmentation, robotics, wireless sensor network, web mining, business management, and medical sciences. Each application domain has different data distribution and complexity. Thus, a single method of partitional clustering might not fit for all problems. Therefore, based on the problem and dataset, the suitable method is selected. As shown in Table 2, the partitional clustering methods are categorized in two classes, namely soft and hard clustering methods, which are discussed in the following subsections.

3.1 Soft clustering methods

Soft clustering methods assign each data to either two or more clusters with a degree of belongingness (or membership) iteratively. The degree of belongingness illustrates the level of association among data more reasonably. The belongingness of a data item with a cluster is a continuous value in the interval $[0, 1]$ and depends upon the objective function. Usually, the minimization of the sum of squared Euclidean distance among the data and formed clusters is considered. Some popular methods are fuzzy c-means (FCM) [10], fuzzy c-shells (FCS) [27], and mountain method [88]. In FCS, each cluster is considered as a multidimensional hypersphere and defines the distance function accordingly. Mountain method uses the Mountain function to find the cluster centres. Particularly, FCM is the most widely used

Table 2 Partitional clustering methods for image segmentation

Sub-categories	Methods	Remarks
Soft	FCM, FCS, FLAME	Grouping based on objective function; Preferred for large datasets.
	Kmeans-based histogram-based Metaheuristic-based	Low time complexity; Number of clusters need to be known priorly; Dependance over the initial clusters.
Hard		

and popular method of this approach [53]. It returns a set of K fuzzy clusters by minimizing the objective function defined in (1).

$$\sum_{i=1}^N \sum_{k=1}^K \mu_{ik}^m \|x_i - v_k\|^2, m \geq 1. \quad (1)$$

where, $\mu_{ik} \in [0, 1]$ and corresponds to membership degree for i^{th} pixel with k^{th} cluster. Equation (1) is optimized iteratively by updating μ_{ik} and v_k according to (2) and (3) respectively.

$$\mu_{ik} = \frac{1}{\sum_{j=1}^K \left(\frac{\|x_i - v_j\|}{\|x_i - v_j\|} \right)^{\frac{2}{m-1}}} \quad (2)$$

$$v_k = \frac{\sum_{i=1}^N \mu_{ik}^m x_i}{\sum_{i=1}^N \mu_{ik}^m} \quad (3)$$

Normally, the exponent of the fuzzy partition matrix (m) is kept as $m > 1$. This regulates the number of pixels that can have membership with more than one cluster. The pseudo-code of FCM is presented in Algorithm 3.

Algorithm 3 Fuzzy C-Means (FCM).

Input: Initialize the number of clusters to be formed.

Output: Clustered data.

Randomly initialize the cluster centroids;

while clustering condition is not satisfied or centroids do not change **do**

 Use (2) to compute the fuzzy partition matrix (U);

 Compute the objective function;

 Update the cluster centroids by (3);

end while

The inclusion of degree of belongingness benefits soft clustering methods in a number of ways like, relatively high clustering accuracy, faster in generating approximate solutions, and efficient in handling incomplete or heterogeneous data. This has resulted in wide applicability of these approaches ranging from discovering association rules to image retrieval. However, there are certain limitations. It requires prior knowledge about the number of clusters to be formed, may return local optimal solutions, low scalability, and sensitive towards initial parameter settings, noise, outliers, and number of formed clusters.

3.2 Hard clustering methods

Hard clustering methods iteratively partition the data into disjoint clusters according to the objective function. Generally, the objective function is the sum of squared Euclidean distance between data and associated centroid which is to be minimized. Usually, the centre of the clustered data is considered as the centroid of the clusters in these methods. Moreover, in contrast to soft clustering, hard clustering assigns data to a single cluster only i.e., each data will have the degree of belongingness as either 0 or 1. The hard clustering approach is relatively simple and scalable with high computation efficiency. Moreover, it is competent for datasets which have spherical shape and are well-separated. However, it suffers from number of demerits like, formed cluster centroids are relatively poor cluster descriptors, sensitive towards initial parameter settings, and needs the pre-knowledge about the

number of clusters to be formed. The various hard-clustering methods may be grouped into three categories, namely Kmeans-based methods, histogram-based thresholding methods, and meta-heuristics based methods as depicted in Fig. 5.

3.2.1 Kmeans-based methods

In Kmeans-based methods, the cluster centroid is updated by taking the mean of all the data items assigned to the corresponding cluster. This is iteratively continued until some defined convergence criterion is met. Although methods of this category have merits like relatively low time complexity, simple in nature, and guaranteed convergence, there are number of limitations which need to be handled. These limitations include the number of clusters to be formed needs to be known priorly, solution quality depends on the initial clusters and number of formed clusters, not appropriate on data having non-convex distribution, follow hill-climbing strategy and hence usually traps into local optima, and relatively sensitive to outliers, noise, and initialization phase.

Generally, this category includes all the methods which are inspired from the K-means method which is the simplest partitional clustering method. It is a golden standard for clustering in literature [66]. An overview of the K-means method is detailed below.

K-means: K-means [48] partitions a set of data points, $X = \{x_1, \dots, x_n\}$, into a number of clusters (k). It performs partition based on the similarity criteria which is usually the sum of squared error defined in (4).

$$J = \sum_{i=1}^k \text{sum}_{x_j \in X} \|x_j - m_i\|^2 \quad (4)$$

where, m_i is the centroid of cluster i which is collectively represented as $M = \{m_1, \dots, m_k\}$ for corresponding clusters, $C = \{c_1, \dots, c_k\}$. This method iteratively minimizes the criterion function J . Further, the formed clusters C and corresponding centroids M are updated as given by (5) and (6) respectively.

$$x_i \in c_l, \text{ if } l = \operatorname{argmin}_{l=1}^k \|x_j - m_i\|^2 \quad (5)$$

$$m_i = \frac{\sum_{x_i \in c_l} x_i}{|c_l|} \quad (6)$$

for $1 \leq i \leq N$ and $1 \leq l \leq k$.

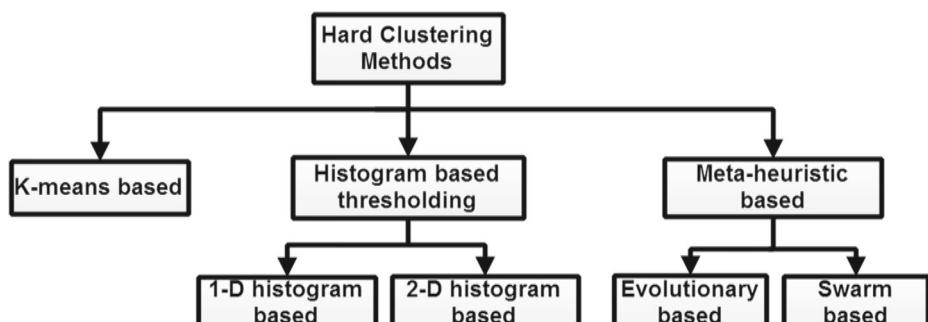


Fig. 5 Classification of hard clustering methods

K-means method has the time complexity of $O(nkt)$ where, n , k , and t correspond to the number of data items, number of clusters to be formed, and maximum iterations respectively. However, this method is biased towards initial cluster centroids and usually traps into local minima. Moreover, solutions vary with the number of clusters. The pseudo-code of the K-means method is presented in Algorithm 4.

Algorithm 4 K-means.

Input: Initialize the number of clusters to be formed.
Output: Clustered data.
Randomly select distinct data points as initial cluster centroids;
while clustering condition is not satisfied or centroids do not change **do**
 Compute the objective function, defined in (4);
 Assign each data point to the cluster which is closest;
 Update the cluster centroids;
end while

Some other popular methods of this category are bisecting K-means [72], sort-means [62], K-harmonic means [92], K-modes [16], K-medoids [61], partition around medoids (PAM) [40], and clustering large applications based upon randomized search (CLARANS) [55]. The K-medoids method corresponds to the variant of K-means which defines the cluster centroid as the data point, nearest to the centre of the cluster. This method is befitting for discrete data. PAM is reminiscent of the medoid-based method in which there are two phases, namely build and swap. It is a greedy search method where the aim is the minimization of the dissimilarity among the objects with the closest medoid. In the first phase, a set of data points are considered as medoids which define the selected objects set. The remaining data points are kept in another set, termed as unselected objects. This corresponds to the steps of the build phase. The next phase tries to improve the cluster quality by swapping the data in the selected objects set with the data in the unselected objects set. Further, CLARANS is an efficient clustering method, especially for the large datasets. It uses the concept of the graph to find the medoids for the given data. It applies PAM on complete data and considers only a subset of the possible swaps between the selected objects and unselected objects.

3.2.2 Histogram-based methods

This category includes methods which perform segmentation by constructing histogram according to the frequency of the intensity values in an image. These methods identify a set of optimal threshold values which partitions the histogram. This partitioning results in grouping of intensity values in different clustering. Therefore, histogram-based segmentation methods can be considered as clustering approach. Generally, a single feature of an image is considered, usually the intensity, to define a 1-dimensional (1D) histogram. For segmentation, histogram information, such as peaks, valleys, and curvatures, are analyzed. Figure 6 illustrates a 1D histogram of an image taken from Berkeley segmentation dataset and benchmark (BSDS) 300 [49] where the x-axis represents the grey levels while the y-axis depicts the frequency. It can be noticed from the histogram that grey levels vary from 20 to 220. Additionally, higher peaks can be observed for grey levels between 50 to 100 and the frequently occurring intensity value is 80. This signifies that many regions are dark which can also be observed in the considered image.

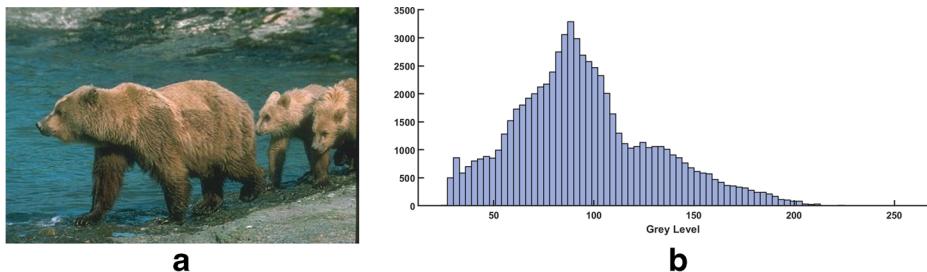


Fig. 6 1D view of grey level histogram of an image taken from BSDS300 [49] **a** Image **b** 1D histogram

To understand the mathematical formulation of 1D histogram-based segmentation method, consider an RGB image of N number of pixels each having intensity from $\{0, 1, 2, \dots, L - 1\}$ in each plane. Suppose, n_i corresponds to the number of pixels for i^{th} intensity level. Therefore, the probability distribution (p_i) of i^{th} intensity level in the image is defined as the probability of occurrence of the image pixels with i^{th} intensity level which is formulated as (7).

$$p_i = \frac{n_i}{N}, \quad 0 \leq i \leq L - 1 \quad (7)$$

The mean intensity of an image plane is determined by (8).

$$\mu = \sum_{i=1}^L i p_i \quad (8)$$

To group the image pixels into n clusters $\{C_1, C_2, \dots, C_n\}$, $n - 1$ thresholds, $(t_1, t_2, \dots, t_{n-1})$, are required which are represented as (9).

$$v(x, y) = \begin{cases} 0, & v(x, y) \leq t_1 \\ \frac{t_1 + t_2}{2}, & t_1 < v(x, y) \leq t_2 \\ \vdots & \vdots \\ \frac{t_{n-2} + t_{n-1}}{2}, & t_{n-2} < v(x, y) \leq t_{n-1} \\ L - 1, & v(x, y) > t_{n-1} \end{cases} \quad (9)$$

where $v(x, y)$ corresponds to pixel intensity at (x, y) location of a $M \times N$ image. The partition C_j , $1 \leq j \leq n$ consists of pixels with intensity value more than t_{j-1} and less than equals to t_j . The frequency of cluster C_j for each plane is computed as (10).

$$w_j = \sum_{i=t_{j-1}+1}^{t_j} p_i \quad (10)$$

The mean for the cluster C_j can be calculated by (11) and the inter-class variance is formulated in (12).

$$\mu_j = \sum_{i=t_{j-1}+1}^{t_j} i p_i / w_j \quad (11)$$

$$\sigma^2 = \sum_{j=1}^n w_j (\mu_j - \mu)^2 \quad (12)$$

To group the pixels into clusters based on intensity, the maximization of inter-class variance [(12)] is considered. Therefore, the objective function tries to maximize the fitness function, defined in (13):

$$\phi = \max_{1 < t_1 < \dots < t_{n-1} < L} \{\sigma^2(t)\} \quad (13)$$

The performance of 1D histogram-based methods is generally unsatisfactory as they consider only single feature like intensity level information of an image and do not deal with spatial correlation among the pixels which is an important parameter for segmentation. To achieve the same, a new histogram has been introduced for image segmentation, termed as 2D-histogram, which has been proven to be better. The concept of 2D histogram-based thresholding was introduced by Abutaleb [2] where the original grey-level information is integrated with local averaging information of pixels to form the grey-local 2D histogram. The experimental results of this method are promising in comparison to 1D histogram-based methods.

A 2D-histogram considers two features of an image at a time. In literature, these features are selected from a set of three image features, namely pixel intensity, average pixel intensity, and pixel gradient. Based on the considered features, two forms of 2D-histograms based image segmentation methods exist, namely grey-local 2D histogram [65] and grey-gradient 2D histogram [87].

Figure 7 presents a grey-local 2D histogram for the sample image taken from the BSDS300 dataset [49]. In the figure, the 2D-histogram is constructed by considering the intensity values of a grey-scale image on the x-axis while the y-axis consists of the average intensity values generated by using the local mean filter on the same image. This representation corresponds to the spatial correlation among the pixels.

The grey-local 2D histogram takes the mean value of a group of pixels surrounding a target pixel to smooth the image which results in loss of information as the fine details such as points, lines, and edges get blurred. To minimize this loss of information, Mittal et al. [51] employed the non-local means filtering instead of local filtering and proposed non-local means 2D histogram. This advantaged in better post-filtering clarity. Moreover, Guang et al. [87] proposed a grey-gradient 2D histogram to consider the edge information. In a grey-gradient 2D histogram, the x-axis represents the grey intensity values while the

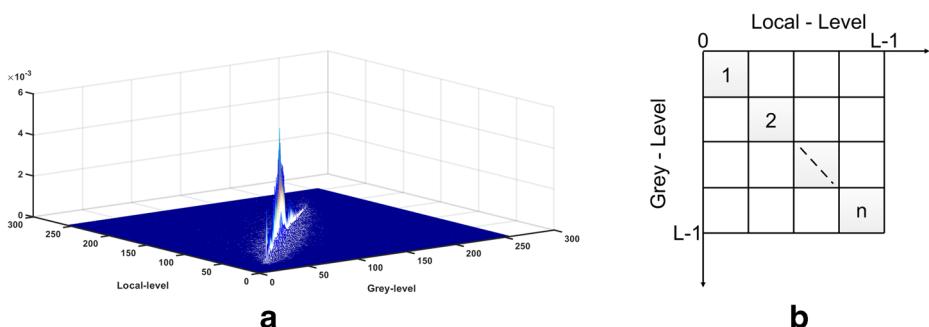


Fig. 7 Grey-local 2D histogram of the image, depicted in Fig. 6a **a** Three dimensional view **b** Two dimensional view

y-axis defines the gradient values of the same image. However, this approach often generates inferior results than grey-local 2D histogram. Figure 8 represents the grey-gradient 2D histogram for the sample image considered in Fig. 6a.

Generally, the segmentation methods based on the histogram are efficient as they require only a single pass through the pixels to construct the histogram. However, these methods have flaws like prior knowledge about the number of clusters is required, highly sensitive towards the noise, and in the overlapping regions of the histogram, it is quite difficult to identify significant peaks and valleys. Moreover, certain limitations which prevail in 2D histograms are that the diagonal regions are not smooth and the off-diagonal information is not adding improvement.

3.2.3 Metaheuristic-based methods

This category involves the use of metaheuristic-based approaches to obtain optimal clusters by updating random solutions according to mathematical formulation and optimality criteria (or objective function). It is a special branch of soft computing that have been leveraged to solve complex real-world problems in reasonable time as compared to classical methods. Broadly, algorithms of this category belong to the class of optimization algorithms which can solve computationally hard problems, like NP-complete problems [8]. However, no single meta-heuristic algorithm is efficient to solve every problem as stated by No Free Lunch Theorem [84]. Generally, a meta-heuristic method solves an optimization problem whose objective function is to perform either maximization or minimization of a cost function ($f(x)$) with a given set of constraints. The mathematically formulation of a maximization optimization problem is presented in (14).

$$\text{Maximize}_{\{x \in \mathbb{R}^d\}} f(x) \quad (14)$$

$$\text{such that : } a_i(x) \geq 0, \quad (15)$$

$$b_j(x) = 0, \quad (16)$$

$$c_k(x) \leq 0 \quad (17)$$

where $x = (x_1, x_2, x_3, \dots, x_d)^T$ is the set of decision variables which is defined over d -dimensions and \mathbb{R}^d is the search space of the problem. $a_i(x)$, $b_j(x)$, and $c_k(x)$ correspond to the different constraints applicable to an optimization problem. Actual constraints depend on the considered optimization problem.

Over the last three decades, more than sixty meta-heuristic algorithms have been proposed based on the inspiration from nature to provide the optimal solution. Each meta-heuristic

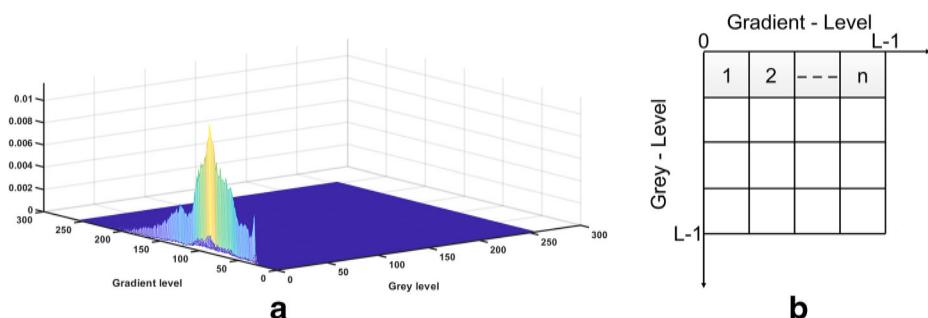


Fig. 8 Grey-gradient 2D histogram of an image, depicted in Fig. 6a **a** Three dimensional view **b** Two dimensional view

algorithm mimics a particular natural phenomena which may belong to evolutionary, physical, or biological. In literature, two common aspects that are often found in these algorithms are exploration and exploitation. Exploration represents the diversification in the search space wherein the existing solutions are updated with the intention of exploring the search space. This helps in exploring the new solutions, prevents the stagnation problem, and responsible for achieving the global solution. The exploitation, which corresponds to the intensification of the current solution, performs the local search around the currently generated solutions. In this, the goal is to exploit the search space and responsible for convergence to the optimal solution. Generally, meta-heuristic algorithms may broadly be classified into two categories, namely evolutionary and swarm algorithms as depicted in Fig. 5. Evolutionary-based algorithms are based on evolution theories such as Darwin's evolutionary theory. The evolutionary algorithms work on the principle of generating better individuals with the course of generation by combining the best individuals of the current generation. Genetic algorithm (GA) [59], evolutionary strategy (ES), [6], differential evolution (DE) [64], biogeography-based optimization (BBO) [58], and probability-based incremental learning (PBIL) [25] are some examples of evolutionary algorithms. On the other side, swarm-based algorithms behave like the swarm of agents, such as fishes or birds, to achieve optimal results. Some algorithms of this category are particle swarm optimization (PSO) [42], ant colony optimization (ACO) [29], gravitational search algorithm (GSA) [54], spider monkey optimization (SMO) [7], grey-wolf optimizer (GWO) [75], cuckoo search (CS) [60], and military dog based optimizer (MDO) [76].

The meta-heuristic algorithms are able to provide promising results for unfolding the image clustering problem. Generally, these clustering-based methods are better than other clustering methods in terms of independence from the initial parameter settings and return global optimal solution [52]. As these methods ensure better solution, they have been widely used in clustering [38]. The basic approach of using meta-heuristics algorithms as clustering algorithm was introduced by Selim and Alsultan [69], using simulated annealing. Thereafter, Bezdek et al. [11] presented genetic algorithm based data clustering method that was basically the first evolutionary based method of data clustering. The first, swarm-based clustering algorithm was introduced by Lumer et al. [44] using ant colony optimization. A literature review of the existing metaheuristic-based clustering for image segmentation is discussed below. Moreover, the pseudo-code for a metaheuristic-based clustering method is presented in Algorithm 5.

Algorithm 5 Metaheuristic-based clustering method.

Input: Initialize the number of clusters to be formed.

Output: Clustered data.

Choose a set of random population where each individual of the population corresponds to a cluster centroid.

while clustering condition is not satisfied or centroids do not change **do**

 Compute the fitness value of each solution through the considered objective function;

 Update the solution according to the meta-heuristic algorithm;

 Assign each data point to the nearest cluster centroid;

end while

Evolutionary-based clustering Genetic algorithm with K-means was explored by Krishna et al. [43] in which the crossover operation of the genetic algorithm was performed by K-means. Subsequently, Maulik et al. [50] introduced the evolutionary-based clustering

method using the genetic algorithm. Moreover, Schwefel [9] presented the recent works done in the area of clustering using evolutionary strategy. It has been perceived that the hybrid evolutionary strategy based algorithms perform better as compared to standard evolutionary strategy on the clustering problem for the majority of the UCI benchmark datasets. In continuation, Sheng et al. [70] introduced K-medoids and genetic-based hybrid algorithm for the clustering of the large-scale dataset. However, it has been observed that hybrid evolutionary based algorithms, which are formed by merging the good features of two or more algorithms, had outperformed the parent algorithms. Lue et al. [46] proposed gene transposition based algorithm for the efficient clustering in which immune cells (population) have been initialized with a vector of K cluster centroids. Likewise, Ye et al. [73] presented GA and PSO based application for image segmentation. Jiao et al. [37] applied memetic based clustering algorithm for the image segmentation of remote sensing images. Further, Lu et al. [47] combined multiple clustering using various measures of the partitions and proposed a fast simulated annealing algorithm for efficient clustering. Agusti et al. [4] explored an algorithm named grouping genetic algorithm to solve clustering problems. The summation of intra-cluster distance was used as the fitness function for performing the partition based clustering, while FCM was the fitness function for the fuzzy-based clustering. Further, Pal et al. [58] used bio-geography based optimizer (BBO) for clustering.

Swarm-based clustering Merve et al. [81] proposed the swarm-based algorithm for partitional clustering using PSO. Chuang et al. [19] introduced a chaotic PSO clustering algorithm in which conventional parameters of the PSO were replaced with chaotic operators. On the same footprints, Tsai and Cao [77] presented a PSO based clustering algorithm with selective particle regeneration. Further, the shuffled frog leaping based clustering algorithm was successfully utilized for image segmentation [12]. Moreover, hybrid variants of PSO with K-means [81], K-harmonic means [90] and rough set theory [34] have also been introduced. Zhang et al. [93] introduced probabilistic c-means and PSO based variant for image segmentation. Furthermore, some researchers proposed swarm and evolutionary based hybrid algorithms for effective data clustering. Xu et al. [86] combined DE with PSO, and presented efficient results. Similarly, PSO was integrated with GA and simulated annealing for the improvements in the clustering as compared to the conventional PSO. Zou et al. [95] introduced cooperative ABC based clustering. Moreover, hybrid algorithms for clustering have also been proposed by a number of researchers and utilized for unfolding image segmentation. Moreover, a clustering algorithm based on bacterial foraging optimization was introduced by Lie et al. [83]. Senthilnath et al. [89] developed the firefly based algorithm for the efficient analysis of the clusters and tested on the UCI datasets. Besides this, the invasive weed optimization based clustering algorithm was presented by Chowdhury et al. [18]. Subsequently, Liu et al. [45] proposed multi-objective invasive weed optimization based algorithm for efficient clustering. Hatamlou et al. [33] introduced GSA based clustering algorithm where the cluster centroids are initialized with K-means.

In general, meta-heuristic methods have a number of merits like, applicable to a wide set of real-world optimization problems, searching behaviour follows a random approach, and able to find the approximate solutions to the NP-complete problems. However, these algorithms suffer from a number of demerits like convergence to the global optima is probabilistic, trap into local optima sometimes, and computational complexity is comparatively high.

4 Performance evaluation parameters

The performance evaluation of a method is necessary to assure the validity of a method. This section lists out various performance measures that researchers have found useful for the quantitative evaluation of an image segmentation method [51]. Table 3 tabulates the formulation of various performance parameters.

1. Confusion matrix (CM): It is a widely used representation for the assessing the efficacy of a classification method [66]. However, it can be used to analyze the results of a clustering method too. The confusion matrix (CM) of size NxN represents that there are N classes (or clusters). In context of clustering, the number of correctly clustered patterns (true positive (TP), true negative (TN)) and wrongly clustered patterns (false positive (FP), false negative (FN)) can be easily identified from the confusion matrix. Table 4 illustrates an example of a typical confusion matrix for 2 clusters, i.e. positive and negative. In the table, TP and TN depict the number of data items which are correctly predicted in positive and negative clusters respectively. FP and FN correspond to the number of data items which are wrongly predicted as positive and negative clusters respectively. Based on Table 4, precision, recall and accuracy can also be computed using (18) – (20).

$$\text{Precision} = \frac{TP}{TP + FP} \quad (18)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (19)$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (20)$$

2. Intersection of Union (IoU): Intersection of Union is the ratio of the number of common pixels between X and Y to the total number of pixels in X and Y. Here, X and Y

Table 3 Various performance parameters for the quantitative evaluation of an image segmentation method [51]

Parameters	Formulation
Boundary Displacement Error (BDE)	$BDE = \begin{cases} \frac{u-v}{L-1}, & 0 < (u - v) \\ 0, & (u - v) \leq 0 \end{cases}$
Probability Rand Index (PRI)	$PRI = \frac{a+b}{a+b+c+d} = \frac{a+b}{L}$
Variation of Information (VoI)	$VoI = H(S_1) + H(S_2) - 2I(S_1, S_2)$
Global Consistency Error (GCE)	$GCE = \frac{1}{n} \left\{ \sum_i E(S_1, S_2, p_i), \sum_i E(S_2, S_1, p_i) \right\}$
Structural Similarity Index (SSIM)	$SSIM = \frac{(2 \times \bar{X} \times \bar{Y} + c_1)(2 \times \sigma_{XY} + c_2)}{(\sigma_X^2 + \sigma_Y^2 + c_2) \times ((\bar{X})^2 + (\bar{Y})^2 + c_1)}$
Feature Similarity Index (FSIM)	$FSIM = \frac{\sum_{x \in \Omega} S_L(x).PC_m(x)}{\sum_{x \in \Omega} PC_m(x)}$
Root Mean Squared Error (RMSE)	$RMSE = \sqrt{MSE(\hat{\theta})} = \sqrt{E((\hat{\theta} - \theta)^2)}$
Peak Signal to Noise Ratio (PSNR)	$PSNR = 10 \log_{10} \frac{(2^b - 1)^2}{\sqrt{MSE}}$
Normalized Cross-Correlation (NCC)	$NCC = \frac{1}{n} \sum_{x,y} \frac{1}{\sigma_X \sigma_Y} (X(x, y) - \bar{X})(Y(x, y) - \bar{Y})$
Average Difference (AD)	$AD = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N X(x, y) - Y(x, y) $
Maximum Difference (MD)	$MD = \max X(x, y) - Y(x, y) $
Normalized Absolute Error (NAE)	$NAE = \frac{\sum_{x=1}^M \sum_{y=1}^N X(x, y) - Y(x, y) }{\sum_{x=1}^M \sum_{y=1}^N X(x, y) }$

Table 4 Confusion matrix

	True Positive	True Negative
Predicted Positive	TP	FP
Predicted Negative	FN	TN

correspond to the segmented image and ground truth respectively. The formulation of IoU is depicted in (21).

$$IoU = \frac{|X \cap Y|}{|X| + |Y|} \quad (21)$$

3. Dice-Coefficient (DC): Dice-Coefficient is defined as twice the number of common pixels divided by total number of pixels in X and Y, where X corresponds to the segmented image and Y is the ground truth. DC is mathematically defined as (22).

$$DC = \frac{2|X \cap Y|}{|X| + |Y|} \quad (22)$$

4. Boundary Displacement Error (BDE): This parameter computes the average boundary pixels displacement error between two segmented images as depicted in (23). The error of one boundary pixel is defined as it's distance from the closest pixel in the other boundary image.

$$\mu_{LA}(u, v) = \begin{cases} \frac{u-v}{L-1} & 0 < u - v \\ 0 & u - v < 0 \end{cases} \quad (23)$$

5. Probability Rand Index (PRI): Probability Rand Index finds labelling consistency between the segmented image and its ground truth. It counts such fraction of pairs of pixels and average the result across all ground truths of a given image as shown in (24).

$$R = \frac{a+b}{a+b+c+d} = \frac{a+b}{n} \quad (24)$$

6. Variation of Information (VOI): Variation of Information as shown in (25) computes the randomness in one segmentation in terms of the distance from given segmentation.

$$VOI(X; Y) = H(X) + H(Y) - 2I(X, Y) \quad (25)$$

7. Global Consistency Error (GCE): A refinement in one segmentation over the other is depicted by the value of GCE as given in (26). If two segmentations are related this way then they are considered as consistent, i.e. both can represent the same natural image segmentation at different scales.

$$GCE = \frac{1}{n} \left\{ \sum_i E(s_1, s_2, p_i), \sum_i E(s_2, s_1, p_i) \right\} \quad (26)$$

8. Structural Similarity Index (SSIM): Structural Similarity Index measures the similarity between two images by taking initial uncompressed or distortion-free image as the reference and computed as (27). It incorporates important perceptual phenomena such as luminance masking and contrast masking.

$$SSIM = \frac{(2 \times \bar{x} \times \bar{y} + c_1)(2 \times \sigma_{xy} + c_2)}{(\sigma_x^2 + \sigma_y^2) \times ((\bar{x})^2 + (\bar{y})^2 + c_1)} \quad (27)$$

9. Feature Similarity Index (FSIM): Feature Similarity Index is a quality score that uses the phase congruency (PC), which is a dimensionless measure and shows the significance of a local structure. It is calculated by (28).

$$FSIM = \frac{\sum_{x \in \Omega} S_L(x) \cdot PC_m(x)}{\sum_{x \in \Omega} PC_m(x)} \quad (28)$$

10. Root Mean squared error (RMSE): The root-mean-squared error computes the difference between sample value predicted by a model or an estimator and actual value. The formulation is shown in (29).

$$RMSE(\hat{\theta}) = \sqrt{MSE(\hat{\theta})} = \sqrt{E((\hat{\theta} - \theta)^2)} \quad (29)$$

11. Peak Signal to Noise Ratio (PSNR in dB): Peak signal-to-noise ratio is defined as the ratio between the maximum possible power of a signal and the power of corrupting noise and is calculated using (30). In general, a higher value of PSNR represents high quality reconstruction and is defined via MSE.

$$PSNR = 10 \log_{10} \frac{(2^n - 1)^2}{\sqrt{MSE}} \quad (30)$$

12. Normalized Cross-Correlation (NCC): Normalized cross-correlation is used for template matching where images are first normalized due to lighting and exposure conditions. It is calculated by subtracting the mean of original and segmented images from the corresponding images, and divided by their standard deviations. Let $t(x, y)$ is the segmented image of $f(x, y)$, then NCC is calculated by (31).

$$\frac{1}{n} \sum_{x,y} \frac{1}{\sigma_f \sigma_t} (f(x, y) - \bar{f})(t(x, y) - \bar{t}) \quad (31)$$

13. Average Difference (AD): This parameter represents the average difference between the pixel values and is computed by (32).

$$AD = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N |x(i, j) - y(i, j)| \quad (32)$$

14. Maximum Difference (MD): This parameter finds the maximum of error signal by taking the difference between original image and segmented image and defined in (33).

$$MD = \max |x(i, j) - y(i, j)| \quad (33)$$

15. Normalized Absolute Error (NAE): The normalized absolute difference between the original and corresponding segmented image gives NAE and is calculated by (34).

$$NAE = \frac{\sum_{i=1}^M \sum_{j=1}^N |x(i, j) - y(i, j)|}{\sum_{i=1}^M \sum_{j=1}^N |x(i, j)|} \quad (34)$$

In the above mentioned parameters, IoU, DC, SSIM, FSIM, PRI, PSNR, and NCC show better segmentation on high values. A high value indicates that the segmented image is more close to the ground truth. To measure the same, these parameters compute values by considering the region of intersection between the segmented image and the ground truth which corresponds to matching the number of similar pixels. On the contrary, other indices prefer lower values for better segmentation as these measures compute error between the segmented image and the ground truth and aim at reducing the difference between them.

5 Image segmentation benchmark datasets

The considered dataset is the key for fair analysis of an image segmentation method. The validation of segmentation methods against benchmark datasets tests its performance against the challenges posed by the considered dataset. Generally, a benchmark dataset consists of a variety of images that varies in number of different aspects. Some of the common challenges which segmentation methods need to handle are illumination variation, intra-class variation, and background complexity. This paper lists some of the popular benchmark datasets in Table 5 which are publicly available for image segmentation task and are briefed below.

- **Aberystwyth Leaf Evaluation Dataset:** It is dataset of timelapse images of *Arabidopsis thaliana* (*Arabidopsis*) plant to perform leaf-level segmentation. It consists of original *arabidopsis* plant images with 56 annotated ground truth images [1].
- **ADE20K:** It is a scene parsing segmentation dataset with around 22K hierarchically segmented images. For each image, there is a mask to segment objects and different parts of the objects [3].

Table 5 Benchmark datasets for image segmentation [23]

S.No.	Dataset	Purpose	Links
1	Aberystwyth Leaf Evaluation Dataset	Segmentation of Leaf	[1]
2	ADE20K Dataset	Segmentation of object and different parts of the objects	[3]
3	Berkeley Segmentation Dataset and Benchmark (BSDS)	Segmentation of objects	[74, 78]
4	Brain MRI Dataset	Segmentation of FLAIR abnormality	[14]
5	CAD120 Affordance Dataset	Segmentation of Affordance of objects	[15]
6	CoVID-19 CT-images Segmentation Dataset	Segmentation of infected region	[21]
7	Crack Detection Dataset	Segmentation of cracks	[5]
8	Daimler Pedestrian Segmentation Benchmark	Segmentation of pedestrian	[24]
9	Epithelium Segmentation Dataset	Segmentation of epithelium region	[80]
10	EVIMO Dataset	Motion segmentation	[30]
11	Liver Tumor Segmentation Dataset	Segmentation of Liver and Liver Tumor	[39]
12	Materials in Context Dataset	Material segmentation	[57]
13	Nuclei Segmentation Dataset	Segmentation of breast cancer nuclei	[79]
14	Objects with Thin and Elongated parts	Segmentation of objects with thin and elongated parts	[20]
15	OpenSurfaces Dataset	Surface segmentation	[56]
16	Oxford-IIIT Pet	Segmentation of pet animals	[82]
17	PetroSurf3D Dataset	Segmentation of petroglyphs	[35]
18	Segmentation Evaluation Database	Segmentation of single or two objects	[67]
19	Sky Dataset	Segmentation of sky	[71]
20	TB-roses-v1 Dataset	Segmentation of rose stems	[26]
21	Tsinghua Road Markings Dataset	Segmentation of road markings	[63]

- **Berkeley Segmentation Dataset and Benchmark (BSDS):** This is a benchmark dataset for evaluation of image segmentation method. It is a collection of 12K manually-labelled segmentations performed on 1K images from the Corel dataset by 30 human subjects. There are two versions of this dataset. BSDS300 [74] and BSDS500 [78] consist of 300 and 500 number of images respectively.
- **Brain MRI dataset:** It is segmentation dataset of MRI images along with manual fluid-attenuated inversion recovery (FLAIR) abnormality segmentation masks [14].
- **CAD120 Affordance Dataset:** This segmentation dataset provides binary masks for each the affordance in the RGB image captured from the Cornell Activity Dataset (CAD) 120. The affordance annotation is performed in the context of human [15].
- **CoVID-19 CT-images Segmentation Dataset:** This dataset consists of 100 CT-scan images of more than 40 CoVID-19 patients along with segmented ground truths. The segmented images were annotated by an expert radiologist [21].
- **Crack Detection Dataset:** It consists of five datasets where each dataset consists of pavement images and associated segmented images for crack detection [5].
- **Daimler Pedestrian Segmentation Benchmark:** This dataset contains pedestrian images along with corresponding segmented ground truth to perform pedestrian segmentation [24].
- **Epithelium Segmentation dataset:** There are 42 ER+ breast cancer images scanned at the magnification of 20x. The ground truth contains manually annotated epithelium region by an expert pathologist [80].
- **EVIMO Dataset:** This is a motion segmentation dataset along with egomotion estimation and tracking captured through event camera. It provides pixel-level masks for the motion segmentation [30].
- **Liver Tumor Segmentation (LITS):** This is a public available benchmark dataset provided by CodeLab for segmentation of liver tumor. It contains 3D-CT scans of liver along with segmented liver and liver tumor [39].
- **Materials in Context (MINC):** This dataset is intended for the segmentation and recognition of materials from the image. It consists of point annotations for 23 different types of materials [57].
- **Nuclei Segmentation dataset:** This dataset contains 143 ER+ breast cancer images scanned at 40x. In total, there are around 12,000 nuclei segmented manually. [79]
- **Objects with Thin and Elongated Parts:** It is a database of three datasets where each image consists of objects having thin and elongated parts. In total, there are 280 images of birds and insects with corresponding ground truth images [20].
- **OpenSurfaces:** This dataset includes tens of thousands of segmented surfaces. The dataset is constructed from interior photographs and annotated with information like material type, texture, and contextual [56].
- **Oxford-IIIT Pet Dataset:** This dataset consists of pet images with their pixel-level trimap segmentation. There are 37 categories of pets and each category contains 200 images [82].
- **PetroSurf3D Dataset:** This is a dataset which contributes in the area of rock art. There are total 26 high-resolution 3D-scans images with pixel-wise labelling for the segmentation of petroglyphs [35].
- **Segmentation Evaluation Database:** This dataset contains 200 gray-level images along with manually generated ground-truth images by three human subjects. Each image consists of either one or two objects in the foreground and the ground truth images are annotated into two or three classes [67].

- **Sky Dataset:** This dataset consists of images taken from the Caltech Airplanes Side dataset. The images, containing sky regions, were selected and their ground truths were prepared. In total, there are 60 images along with ground truth to perform the segmentation of sky [71].
- **TB-roses-v1 Dataset:** This is a small dataset consisting of 35 images of rose bushes and corresponding manually segmented rose stems [26].
- **Tsinghua Road Markings (TRoM):** It is a dataset containing road images along with ground truths for the purpose of segmentation of road markings. There are 19 categories of road markings [63].

6 Findings and conclusion

This paper reviews different clustering-based methods in the field of image segmentation. The clustering methods may be categorized in two broad classes, namely hierarchical and partitional based clustering. Hierarchical clustering methods perform data grouping at different similarity levels. They are suitable for both convex and arbitrary data since these algorithms construct a hierarchy of data elements. However, these methods fail when clusters are overlapping and are quite computationally expensive, especially on high-dimensional dataset like images as these methods scan the $N \times N$ matrix for the lowest distance in each of $N-1$ iterations. On such dataset, partitional clustering methods have performed better and partitions the data into required number of clusters based on a criteria and preferable for convex shape data. Therefore, further study includes an extensive survey in the perspective of partitional based clustering methods. In literature, there exists a number of partitional-based clustering methods, which belong to either soft or hard clustering approaches. Further, the hard partitional clustering methods are categorized into three broad classes, namely Kmeans-based methods, histogram-based methods, and metaheuristic-based methods. A concise overview of the different clustering-based image segmentation methods is presented in Table 6. From the table, following guidelines are extracted.

- The selection of an appropriate clustering method generally depends on certain parameters such as type of data, sensitivity towards noise, scalability, and number of clusters.
- Hierarchical clustering algorithms are suitable for the data set with arbitrary shape and attribute of arbitrary type. There are a number of applications in which hierarchical clustering algorithms are used. For example, all files and folders on the hard disk are organized in a hierarchy and it can be easily managed with the help of hierarchical clustering. In hierarchical clustering, storage and time requirements grow faster than linear rate, Therefore, these methods cannot be directly applied to large datasets like image, micro-arrays, etc. The BIRCH clustering method is computationally efficient hierarchical clustering method; however, it generates low-quality clusters when applied on large datasets. Among all the hierarchical methods, the CURE clustering method is preferred for clustering high dimensional datasets. The computational complexity of the CURE method is $O(n^2 \log n)$, where n corresponds to the number of data points.
- Partitional clustering methods generate disjoint clusters according to some predefined objective functions. The partitional clustering methods are preferred due to their high computing efficiency and low time complexity [66]. For the large datasets, partitional clustering methods are most preferred than other clustering methods. The partitional clustering algorithms are used when number of clusters to be created are known in

Table 6 An overview of the different clustering-based image segmentation methods [85]

Category	Method	Time complexity	Suitable data-type	Suitable data-shape	Noise/outlier sensitive	Scalability	Suitable for high-dimension data	Suitable for large-scale data
Hierarchy	BIRCH	$O(n)$	Numerical	Convex	Low	High	No	Yes
	CURE	$O(s^2 \times \log s)$	Numerical	Arbitrary	Low	High	Yes	No
	ROCK	$O(n^2 \times \log n)$	Categorical	Arbitrary	Low	Average	Yes	No
	Chameleon	$O(n^2)$	Numerical/ Categorical	Arbitrary	Low	High	No	Yes
Partition	FCM	$O(n)$	Numerical	Convex	High	Average	No	No
	FCS	Kernal	Numerical	Arbitrary	High	Low	No	No
	MM	$O(v^2 \times n)$	Numerical	Arbitrary	Low	Low	No	No
	K-means	$O(knt)$	Numerical	Convex	High	Average	No	Yes
	K-medoids	$O(k(n - k)^2)$	Numerical	Convex	Low	Low	No	No
	PAM	$O(k^3 \times n^2)$	Numerical	Convex	Low	Low	No	No
	CLARA	$O(k s^2 + k(n-k))$	Numerical	Convex	Low	High	No	Yes
	CLARANS	$O(n^2)$	Numerical	Convex	Low	Average	No	Yes
	Histogram	$O(ML^2)$	Numerical	Arbitrary	Average	Low	Yes	No
	Metaheuristic	$O(P \times t \times C)$	Numerical/ Categorical	Arbitrary	Low	Low	Yes	Yes

n: Number of data items; k: number of formed clusters; t: number of iterations; s: number of sample data items

P: Population-size; C: complexity of the objective function; L: number of levels; M: number of partitions

advance. For example, partitional clustering is used to group the visitors to a website using their age. Though partitional clustering is computationally efficient, the number of clusters need to be priorly defined. Moreover, partitional clustering methods are sensitive to the outliers and sometimes also trap to local optima. Particularly, FCM and K-means are preferable to perform clustering on spherical-shaped data while histogram-based methods may be advantageous when clustering is to be performed irrespective of the data distribution.

- Metaheuristic-based methods are scalable for clustering on large data, especially when the data distribution is complex. The meta-heuristic clustering mitigate the issue of trapping to local optima by using the exploration and exploitation property. However, existing meta-heuristic methods fail to maintain an efficient balance between the exploitation and exploration process. This will greatly affect the performance and the convergence behaviour of the meta-heuristic method.

Apart from the complete statistics of clustering-based image segmentation methods as tabulated in Table 6, the different performance parameters required for the quantitative evaluation of a segmentation method are discussed. Various performance parameters are briefed along with their formulation. In addition, the publicly available image-segmentation datasets along with active links (till date) are listed out to facilitate the research in the domain of image-segmentation. A dataset consists of a variety of images that varies in number of different aspects like different region-of-interest, image size, applicability of image, and many more. This results in a number of challenges for the segmentation method such as illumination variation, intra-class variation, and background complexity.

In future, the clustering algorithms can be compared on different factors such as stability, accuracy, normalization, and volume of the dataset. The analysis on these factors may reveal quality and performance aspects of the clustering algorithms. Secondly, the applicability of the compared clustering methods on real-time applications needs to be analysed. Thirdly, different cluster validity indices [53] can be considered for the comparison. An efficient variant of existing meta-heuristic methods may improve the performance. Further, based on the findings concluded in this paper, similar work on remaining issues can be considered in continuation and can be presented in future.

References

1. Aberystwyth leaf evaluation dataset — zenodo (2020) <https://zenodo.org/record/168158>, (Accessed 08 Jun 2020)
2. Abutaleb AS (1989) Automatic thresholding of gray-level pictures using two-dimensional entropy. *Comput Vision Graphics Image Process* 47:22–32
3. Ade20k dataset (2020) <http://groups.csail.mit.edu/vision/datasets/ADE20K/>, (Accessed 08 Jun 2020)
4. Agustí L, Salcedo-Sanz S, Jiménez-Fernández S, Carro-Calvo L, Del Ser J, Portilla-Figueras JA et al (2012) A new grouping genetic algorithm for clustering problems. *Expert Syst Appl* 39:9695–9703
5. Amhaz R, Chambon S, Idier J, Baltazar V (2020) Automatic crack detection on 2d pavement images: An algorithm based on minimal path selection. https://www.irit.fr/Sylvie.Chambon/Crack_Detection_Database.html, (Accessed 08 Jun 2020)
6. Babu GP, Murty MN (1994) Clustering with evolution strategies. *Pattern Recognit* 27:321–329
7. Bansal JC, Sharma H, Jadon SS, Clerc M (2014) Spider monkey optimization algorithm for numerical optimization. *Memetic Comput* 6:31–47
8. Bansal JC, Singh P, Saraswat M, Verma A, Jadon SS, Abraham A (2011) Inertia weight strategies in particle swarm optimization. In: 2011 Third world congress on nature and biologically inspired computing. IEEE, pp 633–640

9. Beyer H-G, Schwefel H-P (2002) Evolution strategies—a comprehensive introduction. *Nat Comput* 1:3–52
10. Bezdek JC (1973) Cluster validity with fuzzy sets. *J Cybern* 3:58–73
11. Bezdek JC, Boggavarapu S, Hall LO, Bensaid A (1994) Genetic algorithm guided clustering. In: Proc. of IEEE conference on world congress on computational intelligence. USA, pp 34–39
12. Bhaduri A, Bhaduri A (2009) Color image segmentation using clonal selection-based shuffled frog leaping algorithm. In: Proc. of IEEE international conference on advances in recent technologies in communication and computing. India, pp 517–520
13. Bouguettaya A, Yu Q, Liu X, Zhou X, Song A (2015) Efficient agglomerative hierarchical clustering. *Expert Syst Appl* 42:2785–2797
14. Brain mri segmentation — kaggle (2020) <https://www.kaggle.com/mateusbuda/lgg-mri-segmentation>, (Accessed 08 Jun 2020)
15. Cad 120 affordance dataset — zenodo (2020) <https://zenodo.org/record/495570>, (Accessed 08 Jun 2020)
16. Chaturvedi A, Green PE, Caroll JD (2001) K-modes clustering. *J Classif* 18:35–55
17. Chavent M, Lechevallier Y, Briant O (2007) Divclus-t: A monothetic divisive hierarchical clustering method. *Comput Stat Data Anal* 52:687–701
18. Chowdhury A, Bose S, Das S (2011) Automatic clustering based on invasive weed optimization algorithm. In: Proc. of springer international conference on swarm, evolutionary, and memetic computing. India, pp 105–112
19. Chuang L-Y, Hsiao C-J, Yang C-H (2011) Chaotic particle swarm optimization for data clustering. *Expert Syst Appl* 38:14555–14563
20. Coift (2020) <http://www.vision.ime.usp.br/lucyacm/thesis/coift.html>, (Accessed 08 Jun 2020)
21. Covid-19 - medical segmentation (2020) <https://medicalsegmentation.com/covid19/>, (Accessed 10 Jun 2020)
22. Cs231n convolutional neural networks for visual recognition (2018) <http://cs231n.github.io/classification/>, (Accessed 28 Dec 2018)
23. Cvonline: Image databases (2020) <http://homepages.inf.ed.ac.uk/rbf/CVonline/Imagedbase.htm>, (Accessed 10 Jun 2020)
24. Daimler pedestrian segmentation benchmark (2020) http://www.gavrila.net/Datasets/Daimler_Pedestrian_Benchmark_D/Daimler_Pedestrian_Segmentatio/daimler_pedestrian_segmentatio.html, (Accessed 08 Jun 2020)
25. Dasgupta D, Michalewicz Z (2013) Evolutionary algorithms in engineering applications. Springer Science & Business Media, Berlin
26. Data master nicola strisciuglio / rustico gitlab (2020) <https://gitlab.com/nicstrisc/RUSTICO/tree/master/data>, (Accessed 08 Jun 2020)
27. Dave RN, Bhaswan K (1992) Adaptive fuzzy c-shells clustering and detection of ellipses. *IEEE Trans Neur Netw* 3:643–662
28. Dhillon IS, Mallela S, Kumar R (2003) A divisive information-theoretic feature clustering algorithm for text classification. *J Mach Learn Res* 3:1265–1287
29. Dorigo M, Blum C (2005) Ant colony optimization theory: A survey. *Theor Comput Sci* 344:243–278
30. Evimo - motion segmentation with event cameras (2020) <https://better-flow.github.io/evimo/index.html>, (Accessed 08 Jun 2020)
31. Guénocque A., Hansen P, Jaumard B (1991) Efficient algorithms for divisive hierarchical clustering with the diameter criterion. *J Classif* 8:5–30
32. Guha S, Rastogi R, Shim K (2001) Cure: An efficient clustering algorithm for large databases. 26:35–58
33. Hatamlou A, Abdullah S, Nezamabadi-Pour H (2012) A combined approach for clustering based on k-means and gravitational search algorithms. *Swarm Evolution Comput* 6:47–52
34. Huang KY (2011) A hybrid particle swarm optimization approach for clustering and classification of datasets. *Knowl-Based Syst* 24:420–426
35. Icg - 3dpitotidataset (2020) <https://www.tugraz.at/institute/icg/research/team-bischof/lrs/downloads/3dpitotidataset/>, (Accessed 08 Jun 2020)
36. Janowczyk A, Madabhushi A (2016) Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. *J Pathology Inform* 7:17–42
37. Jiao L, Gong M, Wang S, Hou B, Zheng Z, Wu Q (2010) Natural and remote sensing image segmentation using memetic computing. *IEEE Comput Intell Mag* 5:78–91
38. Jose-Garcia A, Gómez-Flores W. (2016) Automatic clustering using nature-inspired metaheuristics: A survey. *Appl Soft Comput* 41:192–213
39. Junqiangchen/lits-liver-tumor-segmentation-challenge: Lits - liver tumor segmentation challenge (2020) <https://github.com/junqiangchen/LiTS---Liver-Tumor-Segmentation-Challenge>, (Accessed 08 Jun 2020)

40. Kanungo T, Mount DM, Netanyahu NS, Piatko CD, Silverman R, Wu AY (2002) An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Trans Pattern Anal Machine Intell* 881–892
41. Karypis G, Han E.-H., Kumar V (1999) Chameleon: Hierarchical clustering using dynamic modeling. *Computer* 32:68–75
42. Kennedy J (2011) Particle swarm optimization. In: Encyclopedia of machine learning. Springer
43. Krishna K, Murty MN (1999) Genetic k-means algorithm. *IEEE Trans Syst Man Cybern Part B (Cybern)* 29:433–439
44. Langham AE, Grant P (1999) Using competing ant colonies to solve k-way partitioning problems with foraging and raiding strategies. In: Proc. of springer european conference on artificial life. Switzerland, pp 621–625
45. Liu R, Wang X, Li Y, Zhang X (2012) Multi-objective invasive weed optimization algortihm for clustering. In: Proc. of IEEE congress on evolutionary computation. Australia, pp 1–8
46. Liu T, Zhou Y, Hu Z, Wang Z (2008) A new clustering algorithm based on artificial immune system. In: Proc. of international conference on fuzzy systems and knowledge discovery. USA, pp 347–351
47. Lu Z, Peng Y, Ip HH (2011) Combining multiple clusterings using fast simulated annealing. *Pattern Recogn Lett* 32:1956–1961
48. MacQueen J et al (1967) Some methods for classification and analysis of multivariate observations. In: Proc. of berkeley symposium on mathematical statistics and probability. USA pp 281–297
49. 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: Proc. of IEEE international conference on computer vision. Canada, pp 1–11
50. Maulik U, Bandyopadhyay S (2000) Genetic algorithm-based clustering technique. *Pattern Recogn* 33:1455–1465
51. Mittal H, Saraswat M (2018) An optimum multi-level image thresholding segmentation using non-local means 2d histogram and exponential kbest gravitational search algorithm. *Eng Appl Artif Intell* 71:226–235
52. Mittal H, Saraswat M (2019) An automatic nuclei segmentation method using intelligent gravitational search algorithm based superpixel clustering. *Swarm Evolution Comput* 45:15–32
53. Mittal H, Saraswat M (2020) A new fuzzy cluster validity index for hyper-ellipsoid or hyper-spherical shape close clusters with distant centroids, *IEEE Trans Fuzzy Syst*
54. Mittal H, Tripathi A, Pandey AC, Pal R (2020) Gravitational search algorithm: A comprehensive analysis of recent variants. *Multimed Tools Appl* 1–28
55. Ng RT, Han J (2002) Clarans: A method for clustering objects for spatial data mining. *IEEE Trans Knowl Data Eng* 5:1003–1016
56. Opensurfaces - a richly annotated catalog of surface appearance (2020) <http://opensurfaces.cs.cornell.edu/publications/opensurfaces/>, (Accessed 08 Jun 2020)
57. Opensurfaces - a richly annotated catalog of surface appearance (2020) <http://opensurfaces.cs.cornell.edu/publications/minc/>, (Accessed 08 Jun 2020)
58. Pal R, Saraswat M (2019) Histopathological image classification using enhanced bag-of-feature with spiral biogeography-based optimization. *Appl Intell* 1–19
59. Pal R, Yadav S, Karnwal R et al (2020) Eewc: Energy-efficient weighted clustering method based on genetic algorithm for hwsns. *Complex Intell Syst* 1–10
60. Pandey AC, Rajpoot DS, Saraswat M (2020) Feature selection method based on hybrid data transformation and binary binomial cuckoo search. *J Ambient Intell Human Comput* 11(2):719–738
61. Park HS, Jun CH (2009) A simple and fast algorithm for k-medoids clustering. *Expert Syst Appl* 36:3336–3341
62. Phillips SJ (2002) Acceleration of k-means and related clustering algorithms. In: Lecture notes of springer workshop on algorithm engineering and experimentation. USA, pp 166–177
63. roadmarking (2020) <http://www.tromai.icoc.me/>, (Accessed 08 Jun 2020)
64. Saraswat M, Arya K, Sharma H (2013) Leukocyte segmentation in tissue images using differential evolution algorithm. *Swarm Evolution Comput* 11:46–54
65. Sarkar S, Das S (2013) Multilevel image thresholding based on 2d histogram and maximum tsallis entropy-a differential evolution approach. *IEEE Trans Image Process* 22:4788–4797
66. Saxena A, Prasad M, Gupta A, Bharill N, Patel OP, Tiwari A, Er MJ, Ding W, Lin CT (2017) A review of clustering techniques and developments. *Neurocomputing* 267:664–681
67. Segmentation evaluation database (2020) http://www.wisdom.weizmann.ac.il/vision/Seg_Evaluation_DB/index.html, (Accessed 08 Jun 2020)
68. Seifoddini HK (1989) Single linkage versus average linkage clustering in machine cells formation applications. *Comput Indust Eng* 16:419–426

69. Selim SZ, Alsultan K (1991) A simulated annealing algorithm for the clustering problem. *Pattern Recogn* 24:1003–1008
70. Sheng W, Liu X (2004) A hybrid algorithm for k-medoid clustering of large data sets. In: Proc. of IEEE congress on evolutionary computation. USA, pp 77–82
71. Sky dataset (2020) <https://www.ime.usp.br/eduardob/datasets/sky/>, (Accessed 08 Jun 2020)
72. Steinbach M, Karypis G, Kumar V et al (2000) A comparison of document clustering techniques. In: Proc. of ACM international conference on knowledge discovery and data mining workshop on text mining. USA, pp 1–20
73. Sun F-J, Tian Y (2010) Transmission line image segmentation based ga and pso hybrid algorithm. In: Proc. of IEEE international conference on computational and information sciences. China, pp 677–680
74. The berkeley segmentation dataset and benchmark (2020) <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bbsd/>, (Accessed 08 Jun 2020)
75. Tripathi AK, Sharma K, Bala M (2018) A novel clustering method using enhanced grey wolf optimizer and mapreduce. *Big Data Res* 14:93–100
76. Tripathi AK, Sharma K, Bala M, Kumar A, Menon VG, Bashir AK (2020) A parallel military dog based algorithm for clustering big data in cognitive industrial internet of things. *IEEE Trans Indust Inform*
77. Tsai C-Y, Kao I-W (2011) Particle swarm optimization with selective particle regeneration for data clustering. *Expert Syst Appl* 38:6565–6576
78. Uc berkeley computer vision group - contour detection and image segmentation - resources (2020) <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/resources.html>, (Accessed 08 Jun 2020)
79. Use case 1: Nuclei segmentation - Andrew Janowczyk (2020) <http://www.andrewjanowczyk.com/use-case-1-nuclei-segmentation/>, (Accessed 08 Jun 2020)
80. Use case 2: Epithelium segmentation - Andrew Janowczyk (2020) <http://www.andrewjanowczyk.com/use-case-2-epithelium-segmentation/>, (Accessed 08 Jun 2020)
81. Van der Merwe D, Engelbrecht AP (2003) Data clustering using particle swarm optimization. In: Proc. of IEEE congress on evolutionary computation. Australia, pp 215–220
82. Visual geometry group - university of oxford (2020) <https://www.robots.ox.ac.uk/vgg/data/pets/>, (Accessed 08 Jun 2020)
83. Wan M, Li L, Xiao J, Wang C, Yang Y (2012) Data clustering using bacterial foraging optimization. *J Intell Inform Syst* 38:321–341
84. Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. *IEEE Trans Evol Comput* 1:67–82
85. Xu D, Tian Y (2015) A comprehensive survey of clustering algorithms. *Annals Data Sci* 2:165–193
86. Xu R, Xu J, Wunsch DC (2010) Clustering with differential evolution particle swarm optimization. In: Proc. of IEEE congress on evolutionary computation. Spain, pp 1–8
87. Xue-guang W, Shu-hong C (2012) An improved image segmentation algorithm based on two-dimensional otsu method. *Inform Sci Lett* 1:77–83
88. Yager RR, Filev DP (1994) Approximate clustering via the mountain method. *IEEE Trans Syst Man Cybern* 24:1279–1284
89. Yang X-S (2010) Firefly algorithm, stochastic test functions and design optimisation. *Int J Bio-Inspired Comput* 2:78–84
90. Yang F, Sun T, Zhang C (2009) An efficient hybrid data clustering method based on k-harmonic means and particle swarm optimization. *Expert Syst Appl* 36:9847–9852
91. Zaitoun NM, Aqel MJ (2015) Survey on image segmentation techniques. *Procedia Comput Sci* 65:797–806
92. Zhang B, Hsu M, Dayal U (2001) K-harmonic means-a spatial clustering algorithm with boosting. In: Lecture notes of springer workshop on temporal, spatial, and spatio-temporal data mining. France, pp 31–45
93. Zhang Y, Huang D, Ji M, Xie F (2011) Image segmentation using pso and pcm with mahalanobis distance. *Expert Syst Appl* 38:9036–9040
94. Zhang T, Ramakrishnan R, Livny M (1996) Birch: An efficient data clustering method for very large databases. In: Proc. of ACM sigmod record, pp 103–114
95. Zou W, Zhu Y, Chen H, Sui X (2010) A clustering approach using cooperative artificial bee colony algorithm. *Discret Dyn Nat Soc* 2010:22–37

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.