Image and Vision Computing 88 (2019) 29–40

Contents lists available at ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

EF-Index: Determining number of clusters (K) to estimate number of segments (S) in an image☆

Mrinal Kanti Bhowmik [a,]*, Tathagata Debnath [a], Debotosh Bhattacharjee [b], Paramartha Dutta [c]

[a] Department of Computer Science and Engineering, Tripura University (A Central University), Suryamaninagar, 799022 Agartala, India
[b] Department of Computer Science and Engineering, Jadavpur University, Kolkata 700032, West Bengal, India
[c] Department of Computer and System Sciences, Visva-Bharati University, Santiniketan 731235, West Bengal, India

ARTICLE INFO

Article history:
Received 20 February 2019
Accepted 20 April 2019
Available online 26 April 2019

Keywords:
Clustering
Cluster-indexing
Segmentation
Electrostatic Force Image (EF-Image)
Force Influence Image
Electrostatic Force Index (EF-Index)

ABSTRACT

Estimation of number of segments in an image attracts a formidable interest among the research community. The number of segments in an image is estimated by calculating the number of clusters present in the pixels of that image. The present work offers an unsupervised method, named "Electrostatic Force Index (EF-Index)", to estimate the number of clusters inherent in an image, reporting of which is very rare in literature. The proposed approach is inspired by Coulomb's law of electrostatics. The EF-Index explores the mutual influence of an arbitrary pixel on another, by considering them similar to point charges. Our proposed cluster indexing method, viz. EF-Index is capable of determining the number of clusters present in an image. It has a strong resemblance to the way the electrostatic force is operative between a pair of static point charges in a closed system as per Coulomb's principle. In order to justify the effectiveness of the proposed approach, we have compared EF-Index of a given image with *DB-Index*, *I-Index*, *CVNN-Index*, *DOE-AND-SCA* and *Sym-Index* of the same image. Experimental results show that EF-Index is same as other state-of-the-art indices, whereas EF-Index does not require any clustering algorithm. To establish the applicability of the EF-Index, the same is applied for image segmentation considering Berkeley Segmentation Dataset and Stanford Background Dataset. We observe the results obtained conform to the ground truth and results achieved by applying existing well-established segmentation techniques on the same datasets. The efficacy of the proposed approach is further substantiated in terms of its reduced computational overhead in comparison to the state-of-the-art segmentation algorithms.

1. Introduction

Determination of the number of segments (S) inherent in an image for image understanding is a challenging problem in computer vision. The lower bound of number of segments (S) can be determined by the number of clusters (K) present in an image. The proper value of the number of segment or number of clusters is particularly useful in clustering-based image segmentation [1–6] by assigning a proper label to an individual pixel in the image, resulting in the formation of disjoint, coherent and compact regions [36,37].

Determining number of clusters automatically in an optimal sense has been studied for decades. All popular algorithms for the number of clusters determination of in an image [10,11,13,14,23] are inoperative without completion of the clustering task. The information about the number of segments or number of clusters in an image can be used to segment the image. However, finding the number of clusters using any of the existing state-of-the-art algorithms viz. *DB-Index* [10], *I-Index* [11], *CVNN-Index* [13], *DOE-AND-SCA* [14], and *Sym-Index* [23] is not possible without applying a clustering algorithm multiple times, which is computationally expensive. Therefore, an automatic technique for identifying the number of clusters without applying any underlying clustering algorithm is highly desirable. Present literature lacks such a robust algorithm. This problem is the prime motivation behind the development of the EF-Index. The proposed EF-Index algorithm can determine the number of clusters present in an image independent of any clustering algorithm. The output of the EF-Index algorithm i.e. the information about the number of clusters in an image can be used as number of segments to further segment the image.

Image segmentation algorithms can be broadly divided into two categories viz. clustering-based segmentation algorithms like *K-Means* (*KM*) [7], and *Fuzzy-C-Means* (*FCM*) [8] and non-clustering-based segmentation algorithms like N-Cut (Normalized Cuts) [32], Felzenszwalb–Huttenlocher (FH) [33], Fractional-Order Darwinian Particle Swarm Optimization (FODPSO) [34], etc. Both of them utilize the information about the number of clusters directly or indirectly.

☆ This paper has been recommended for acceptance by S. Todorovic.

* Corresponding author.
E-mail addresses: mrinalkantibhowmik@tripurauniv.in (M.K. Bhowmik), tirtha.debnath@gmail.com (T. Debnath), debotosh@ieee.org (D. Bhattacharjee), paramartha@ieee.org (P. Dutta).

https://doi.org/10.1016/j.imavis.2019.04.009
0262-8856/

The advantage of a priori knowledge about the correct value of K and S is as follows:

(a) Direct benefit:

Many state-of-the-art clustering-based segmentation algorithms [7–9] require the appropriate value of K (number of clusters) as an input parameter which influences the quality of the segmentation.

(b) Indirect benefits:

(i). For non-clustering-based image segmentation algorithms, this information about the number of segments may help in the proper determination of input parameters, which eventually results in proper segmentation [44].

(ii). The number of clusters or number of segments may be used as a parameter for categorization or grouping of images for image understanding or content based image retrieval. Due to geometrical transformations like T, R, S, etc. member of clusters in an image remains almost same. If number of segments present in one image is not close to that of another image then it is ensured that the images are not similar. Moreover, if the images are to be similar then the number of clusters present in the concerned images should also have values close enough. Therefore, number of clusters can also be used as a parameter for coarse grouping of a set of given images.

Incidentally, the number of clusters (K) and the number of segments (S) in an image may or may not be equal. In most of the occasions, K is dominated above by S in an image. However, in some typical situations, it may be otherwise also. Therefore the non-clustering-based segmentation algorithms will be the indirect beneficiary of the information regarding the number of clusters and/or segments.

In most of the situations, it is tough to ascertain the correct value of number of clusters/segments directly from an image content. Therefore, an automated selection of the number of clusters K is highly desirable as well as challenging for any application, as it will eliminate the necessity of human intervention. Improper choice of this number K may lead to undesirable consequence like either under-segmentation or over-segmentation [37].

Over the years many algorithms capable of deciding the number of clusters in statistical datasets have been reported in the literature. Davies–Bouldin Index (*DB-Index*) [10], *I-Index* [11], Clustering Validation Index based on Nearest Neighbors (*CVNN-Index*) [13], Dynamic Nearest Neighbors (DOE-AND-SCA) [14] and Symmetry distance-based index (*Sym-Index*) [23] are some of the popular algorithms to this effect. Most of them are frequently used in data mining [29,46], voice mining [15], web mining [16], and text mining [17]. These are also subsequently used in images [22,23]. Indices mentioned here, measure the similarity between the constituent members of a cluster and dissimilarity among the members of two different clusters or some function of those indices. For any statistical dataset or image, these indices generate a value corresponding to a K (number of clusters). From the outputs, correct K (number of clusters) value is calculated by finding the corresponding optimum value. For example, DB Index provides the correct number of clusters for minimum value [10] and the *I*-Index corresponds to the maximum value [11].

Despite being effective and easy to use for images in specific and datasets in general, all the indexing based techniques have some inherent drawbacks. These are:

(a) Dependency on prior clustering or partitioning:

Any such index value is available only on execution of some clustering algorithms. Bypassing the execution of a clustering algorithm, determination of a value for cluster index is not yet reported.

(b) Performance of clustering algorithm

Due to the above mentioned dependency on clustering algorithms, the performance of the indexes is also largely dependent on the choice of the clustering algorithm. Application of different indexing techniques using same clustering technique may report different results. Same indexing techniques may report different values as an effect of applying different clustering algorithms on the same dataset.

From the above discussion, it is clear that a technique devoid of such dependencies would show a distinct advantage in respect of the automatic number of clusters or number of segments determination.

Some such non-Index based automatic techniques exist in the literature for statistical datasets. Some of them are visual methods for cluster tendency assessment [18–20]. These algorithms generate an intermediate representation of the given data, which assists in proper prediction of the number of clusters inherent therein. Bezdek and Hathaway proposed the VAT [21] algorithm, falling under this category. Many modifications of the VAT algorithm were subsequently supplemented to the literature such as the DBE [24], iVAT [25], bigVAT [26] and sVAT [27], coVAT [28], asiVAT [45], clusiVAT [47], etc. However, none of them are designed for images.

Despite the presence of all such algorithms determining the number of clusters, in the literature, there is hardly any non-index based algorithm, available for application to images, containing multiple clusters within. Such algorithms if available will not only be time efficient but also be helpful to perform efficient image segmentation based on clustering.

This article proposes a novel method to compute the Electrostatic Force Index (EF-Index), which is the number of pixel clusters within a given image. EF-Index is computed based on influence of electrostatic force on a pixel Intensity exerted by all the pixels of the input image. The proposed method does not have any pre-clustering requirement to find optimum number of clusters in an image, which is an important prerequisite of *DB-Index* [10], *I-Index* [11], *CVNN-Index* [13], *DOE-AND-SCA* [14], *Sym-Index* [23], etc. Hence, the task of clustering or segmentation becomes more effective. The proposed algorithm can act as a preprocessing step of segmentation algorithms in general, and especially for the clustering-based image segmentation algorithms.

To sum up, the major contributions of this article are:

1. Our proposed Electrostatic Force Index (EF-Index) is an unsupervised approach, capable of automatically determining number of clusters inherent in an image.
2. EF-Index is independent of any clustering algorithm.
3. The proposed Index value, which is comparable to that of the number of clusters determined by other state-of-the-art Indexing techniques such as *DB-Index* [10], *I-Index* [11], *CVNN-Index* [13], *DOE-AND-SCA* [14], *Sym-Index* [23].
4. Estimation of Number of Segments from Number of Clusters.

Application of the proposed method is as follows:

1. The present method clubbed with clustering-based image segmentation method produces better segmentation results than the existing state-of-the-art segmentation algorithms [31–34], where K is not needed a priori.
2. Noting the results of experiments conducted on the images used in traditional image processing methods, twenty five in number and collected from different sources, as well as images available in benchmark segmentation image databases viz. the Berkeley Segmentation Dataset (BSDS300) and (BSDS500) [30], and Stanford Background Dataset (SBD) [31].

The rest of the paper is arranged as follows. In Section 2 the problem is defined that our proposed algorithm solves, Section 3 describes our proposed EF-Index algorithm. The Section 4 compares the outputs of the proposed EF-Index algorithm with other state-of-the-art algorithms for calculating the number of clusters. Then we compare the segmentation results yielded by existing clustering algorithms based on the values of K reported by our proposed EF-Index algorithm, with other competing state-of-the-art segmentation approaches on BSDS300, BSDS500, and SBD datasets. Section 5 sums up the proposed contribution and indicates some appropriate future directions.

2. Problem definition

In this section, we shall analyze the relationship between the number of segments and the number of clusters from an image context and subsequently justify the significance of identifying the number of clusters in clustering-based image segmentation with appropriate examples.

The atmosphere has an influence on how far one can see through aerosols, the type of infrared camera used, and especially the waveband in which the camera operates are also of importance. Because the particle size is much larger than the wavelength in the visible portion of the EM spectrum (0.4 to 0.74 μm), attenuation by atmospheric aerosols is independent of wavelength. That means attenuation is worst in case of the visible wavelength. As wavelength increases, attenuation becomes less of an issue. Since wavelengths of the far-infrared are larger than other infrared wave bands, impact of particles on far-infrared is relatively insignificant. Far-infrared provides the advantage of 'seeing' not only at night but also through many atmosphere aerosols such as dust, fog, rain, etc. In Fig. 2, there is some visual and corresponding thermal sample frames have shown in night time at several atmospheric conditions. To characterize the texture, we have estimated entropy value where the visual frames shown less entropy value than thermal frames. Being able to see through low light and atmospheric particles is useful for security and surveillance applications can all benefit from the power of thermal imaging.

2.1. Relationship of the number of segments (S) with the number of clusters (K) in an image

For an image, the number of clusters K may or may not be equal to the number of segments S. As some pixels, belonging to the same cluster, may be associated with more than one segments and vice-versa.

Accordingly, K and S can have three possible relationships — (i) K and S are equal ($K = S$), (ii) K is less than S ($K < S$), or (iii) K is greater than S ($K > S$). Different possibilities are illustrated in Fig. 1. (a) and (c).

Out of these three possibilities, two are often found most frequently viz. $K = S$ and $K < S$. This is primarily because the clusters themselves will form individual segments if not divided into multiple segments. In the case (iii), mentioned above, $K > S$ holds when a whole object is considered as one segment although it has numerous constituent parts. During initial segmentation always $K < S$, but after post processing or refinement, K becomes greater than S ($K > S$). Therefore, it is evident that the number of clusters (K) obtained is a lower bound on the number of segments (S) in an image. Hence, S may be estimated from K. It is worth to note that, since segments are spatially dis-integrated over the space (2D in this case), number of segments (S) is essentially greater than or equal to the number of clusters (K). In this work estimation of number of segments is done based on the principle that $K \leq S$.

In the segmented image, the members of two or more clusters must maintain distinct intensity thereby resulting in the formation of non-coherent regions. For these reasons, the value of K will usually be bounded above by S.

2.2. Influence of number of clusters (K) on clustering-based image segmentation

Proper determination of the value of K is critical for clustering-based image segmentation. Fig. 1. (b) and (d) illustrates this on one synthetic and another natural image. For ready understanding, we may utilize a synthetic image Fig. 1. (b). (i) from which one may easily determine the proper value of K in the image.

Although it is effortless to determine the number of clusters present in such a synthetic image, this is not same for natural images, just by looking at them as evident from Fig. 1. (d). Fig. 1. (c) offers glaring examples to indicate as to how the human vision is inadequate to calculate the correct number of segments and/or clusters present in the image.

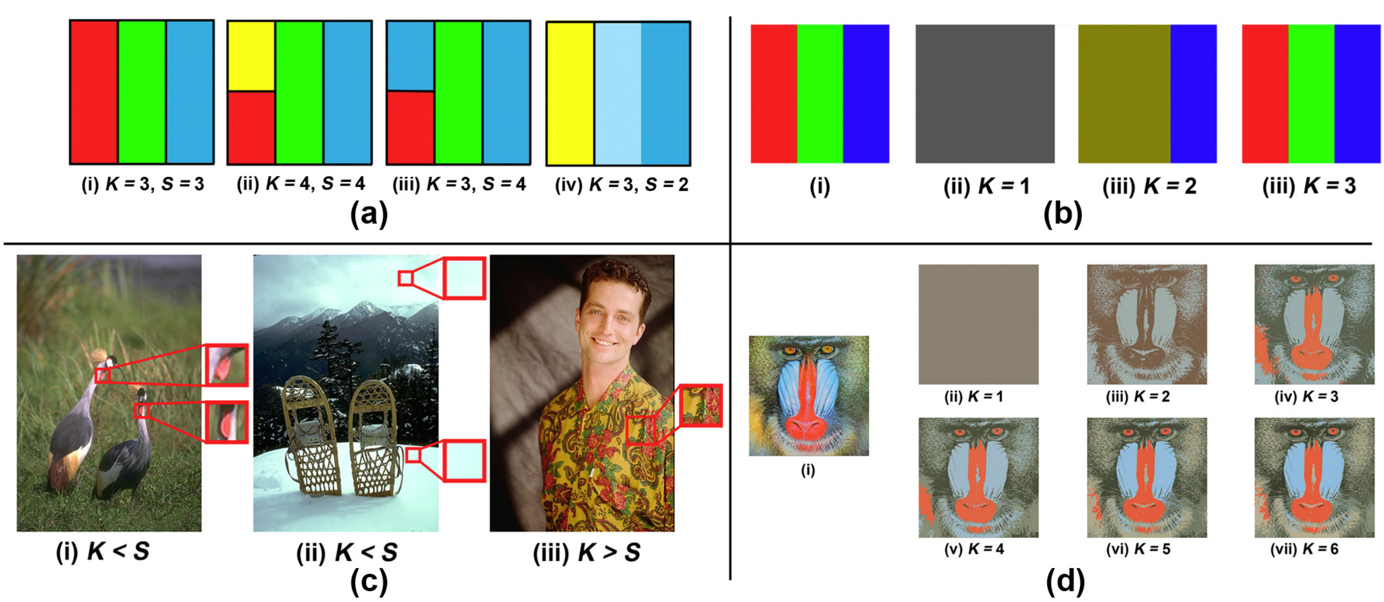

Fig. 1. (a) Synthetic images with different number of clusters (K) and number of segments (S). (a). (i) and (a). (ii) exhibit the situation, $K = S$, (a). (iii) exhibits $K < S$, and (a). (iv) indicates $K > S$. In (a). (i) $K = 3, S = 3$; (a). (ii) $K = 4, S = 4$ but for (a). (iii) $K = 3, S = 4$, and in (a). (iv) $K = 3, S = 2$. (b) The effect of number of clusters (K) on Clustering-based Image Segmentation is explored here. (b). (i) Corresponds to the original image, comprising three segments viz. red, green, and blue. Clustering-based Image Segmentation outcomes with different K as depicted in (b). (ii) $K = 1$, (b). (iii) $K = 2$, (b). (iv) $K = 3$. It is evident that, although the original image is segmented for different values of K, proper segmentation is arrived at $K = 3$ only and not $K = 1$ or 2. On the contrary for $K > 3$, empty clusters are formed, leading thereby to inappropriate and/or no segmentation. (c) Deals with some real life images. (c). (i) and (c). (ii) are real life examples of $K < S$, where the members of a single cluster are found belonging to two different segments. On the contrary, (c). (iii) is an example of $K > S$. In (c). (i) the snoods of the two birds are member of the same cluster but part of two different segments. Similarly in (c). (ii) the sky and the snow have pixels from the same cluster belonging to different segments. However in (c). (iii) the pixels of the shirt belong to multiple clusters being in the same segment. (d) Is a traditional benchmark image. (d). (i) is the input baboon image. Its Clustering-based image Segmentation is done for different K as depicted in (d). (ii) $K = 1$, (d). (iii) $K = 2$, (d). (iv) $K = 3$, (d). (v) $K = 4$, (d). (vi) $K = 5$, and (d). (vii) $K = 6$.

Considering the challenge associated with decide the actual value of K for an arbitrary image and also to ensure its proper segmentation based on clustering, we offer a technique that fulfills the requirements in an automatic manner.

3. Proposed method

In this article, we propose a new Index, named the Electrostatic Force Index (EF-Index) which will directly yield the number of clusters for any input image. It is defined as:

$$K = COUNT\left[DISTINCT\{\ \min(\Phi_{(i,j)}|I_{(i,j)} = p)\}_{P=0,\ldots\ldots,255}\right] \tag{1}$$

Here, K is the Electrostatic Force Index. I is the input image and p is any intensity value of the input image, $i = 1, \ldots\ \ldots, M$ and $j = 1, \ldots \ldots, N$. The size of the input Image I is $(M \times N)$, and Φ is the Electrostatic Force Image of the same size i.e. $(M \times N)$. The method to generate Electrostatic Force Image, Φ, form the input Image, I, is elaborated in Algorithm 1.

To explain the concept of Electrostatic Force, we begin the discussion with the preliminary concept of Coulomb's law of electrostatics. Coulomb's law of electrostatics is applicable in respect of two static point charges. In the present scope, we postulate that such a framework may be modeled in a similar way in a digital image. The influence of an arbitrary pair of pixels participating in the image formation is reflected in their respective intensity values in a very similar way Coulomb's principle of electrostatics works. It is understandable that closer the distance the concerned pixels maintain between themselves, the influence of one pixel on another will be more for any appropriate distance metric.

For the sake of completeness, we shall briefly provide Coulomb's law of electrostatics, which is the building block of the proposed Electrostatic Force Index.

3.1. Coulomb's law of electrostatics

Coulomb's law of electrostatics [12] provides the mathematical foundation for the interaction between two static point charges. In Fig. 2. (a) two point charges A and B with charges, q_A and q_B respectively situated at distance r_{AB} apart interact via Electrostatic force, represented by:

$$\vec{F}_{AB} = \frac{C \times q_A \times q_B}{r_{AB}^2} \tag{2}$$

where, C is known as Coulomb's constant, originally represented by K, we have replaced it with C in order to avoid confusion with K that is representing the Electrostatic Force Index.

Now, we shall discuss how different pixels with their associated intensities in an image show resemblance with Coulomb's principle.

Fig. 2. (b) explains how two pixels P and Q located at (i,j) and (x,y) positions in the input image, I influence one another, maintaining a Manhattan distance r_{PQ}. They have respective intensities of I_P and I_Q, intensities being scalar [44] we consider their magnitudes only in the present scope. The Electrostatic Force is represented by Eq. (3).

$$F_{PQ} = \frac{C \times I_P \times I_Q}{r_{PQ}^2} \tag{3}$$

where, C is the proportionality constant like Coulomb's constant. Later we shall see that C may assume any constant value and hence in the present scope the value of C is taken as unity for computational simplicity.

Then the input image I is normalized to [0,1] range from [0,255] range by dividing every pixel with 255, the maximum gray-level value [39] of any image, as per Eq. (4). The new image is represented as Γ.

$$\Gamma_{(i,j)} = \frac{I_{(i,j)}}{255} \tag{4}$$

It is assumed that a pixel does not have any force contribution on itself, behaviorally similar to a point charge. Accordingly, the cumulative force viz. the Electrostatic Force $F_{(i,j)}$ on any $(i,j)th$ pixel of an image, Γ, having $(M \times N)$ pixels will be:

$$F_{(i,j)} = \left[\sum_{\substack{x=1\\(i,j)}}^{M} \sum_{\substack{y=1\\ \neq(x,y)}}^{N} \frac{C \times \Gamma_{(i,j)} \times \Gamma_{(x,y)}}{r^2_{(i,j)(x,y)}} \right] \tag{5}$$

where, $[x]$ is the integer value closest to x. $\Gamma_{(i,j)}$ and $\Gamma_{(x,y)}$ are the normalized intensity values stored at the coordinates (i,j) and (x,y) of Γ. $r_{(i,j)(x,y)}$ is the Manhattan distance between the locations (i,j) and (x,y).

3.2. The Electrostatic Force Index (EF-Index)

To compute Electrostatic Force Index (EF-Index), an Electrostatic Force Image (EF-Image), Φ, is constructed, using Eq. (6), from the corresponding Force Matrix, F, obtained from the given image, I.

$$\Phi_{(i,j)} = \left[\frac{255}{(F_{\max} - F_{\min})} \times \left(F_{(i,j)} - F_{\min}\right)\right] \tag{6}$$

where, $F_{\max}$ and $F_{\min}$ are the maximum and minimum values of the Force matrix, F respectively. By this operation the force values are converted to [0,255] range. Due to the scale fitting operation in Eq. (6), the intensities stored in Φ will be invariant of any choice of C. The influence of the Electrostatic Force Image, Φ, on the corresponding input image I is represented as the Force Influence Image, Ψ, given by Eqs. (7) to (9):

$$\Theta_p = \Theta_p \cup \{\Phi_{(i,j)}\}, \quad \forall I_{(i,j)} = p \tag{7}$$

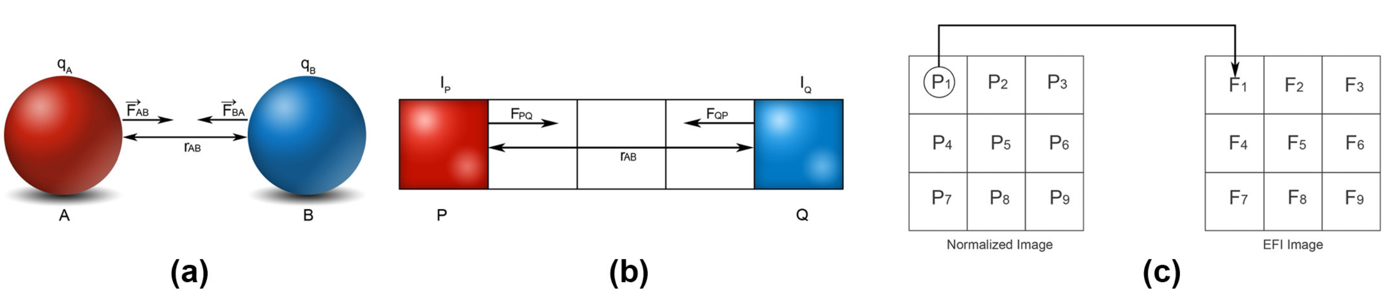

Fig. 2. (a) Two static point charges A and B, having distance r_{AB} exerting forces $(F_{(A,B)}, F_{(B,A)})$ on each other obeying Coulomb's law of electrostatics. (b) Two pixels P and Q are in similar situation as of the two point charges A and B, respectively. (c) The total force on P_1 in the normalized image is stored in the corresponding location of the Electrostatic Force image.

The set Θ_p, which is initially *NULL*, stores the influences of Electrostatic Force on all the pixels of input image *I* having the intensity value *p*.

$$\theta_p = \min\{\Theta_p\} \mid |\Theta_p|>0 \tag{8}$$

For any intensity level, *p*, present in *I*, the minimum of all location-corresponding intensity values from Θ_p is stored in θ_p.

All the corresponding pixels of *I* having intensity value *p* are replaced by θ_p and stored as the Force-Influence Image, Ψ, as shown in Eq. (9).

$$\Psi_{(i,j)} = \theta_p, \forall I_{(i,j)} = p \tag{9}$$

To compute Electrostatic Force Index (EF-Index) all the existing values of θ_p, are stored in set, Ω, given in Eq. (10).

$$\Omega = \{x | x = \theta_p 0 \le p, x \le 255\} \tag{10}$$

The Electrostatic Force Index (*K*) of the input image *I* is calculated as the number of distinct non-negative intensity values present in θ, which is computed as the cardinality of the set Ω, given by Eq. (11).

$$K = |\Omega| \tag{11}$$

Next, we analyze the behavioral aspect of Electrostatic Force Image and its influence on the input image.

To compile together, the final intensity $\Phi_{(i,j)}$ corresponding to any initial intensity $I_{(i,j)}$ is set preferably at the minimum final intensity level, without any alteration, to neutralize the effects of the surrounding pixel intensities and the location of the pixel under consideration.

Lemma 1. The Electrostatic Force on pixels having same intensity values is constant and independent of location.

Proof. The total force on any pixel is represented by Eq. (5). The effects of surrounding intensities (case-2) and the location (case-3) of any pixel in *I* are ignored by setting the value of a set of pixels present in Φ to the minimum and stored in Ψ, where all these pixels correspond to pixels having the same intensity value in the input Image *I*. As a result, the total force exerted on any arbitrary pixel from a set of pixels having same intensity value is independent of its location.

Lemma 2. The contribution of other pixels to the computation of force on any pixel is always constant.

Proof. The total force on any pixel is obtained by Eq. (5).

This equation can be modified as Eq. (13)

$$F_{(i,j)} = C \times \Gamma_{(i,j)} \times \sum_{\substack{x=1 \\ (i,j)}}^{M} \sum_{\substack{y=1 \\ \neq (x,y)}}^{N} \frac{\Gamma_{(x,y)}}{r^2_{(i,j)(x,y)}} \tag{13}$$

We can assume, Eq. (14).

$$S = \sum_{\substack{x=1 \\ (x,y)}}^{M} \sum_{\substack{y=1 \\ \neq (i,j)}}^{N} \Gamma_{(i,j)} \tag{14}$$

which is the cumulated intensity of the image.

Now considering Eq. (15).

$$d_{(i,j)} = \sum_{\substack{x=1 \\ (i,j)}}^{M} \sum_{\substack{j=1 \\ \neq (x,y)}}^{N} \frac{1}{r^2_{(i,j)(x,y)}} \tag{15}$$

So, we can rewrite Eq. (13) as Eq. (16).

$$F_{(i,j)} = C \times \Gamma_{(i,j)} \times \left(S - \Gamma_{(i,j)}\right) \times d_{(i,j)} \tag{16}$$

So, we can modify Eq. (16) as Eq. (17).

$$F_{(i,j)} = f\left(\Gamma_{(i,j)}, S, d_{(i,j)}\right) \tag{17}$$

i.e. $F_{(i,j)}$ is a function of normalized pixel intensity $\Gamma_{(i,j)}$ of the $(i,j)th$ pixel, total intensity of the image *S*, which will be constant for any image and also the position of the $(i,j)th$ pixel $d_{(i,j)}$.

From Lemma 1 we can say that, $F_{(i,j)}$ is independent of $\Gamma_{(x,y)}$ and $r^2_{(i,j)(x,y)}$. So, by using Eqs. (14) and (15), we can say $F_{(i,j)}$ is independent of *S* and $d_{(i,j)}$. So, the Eq. (17) can be written as Eq. (18).

$$F_{(i,j)} = f_1\left(\Gamma_{(i,j)}\right) \tag{18}$$

The highest degree of $\Gamma_{(i,j)}$ is one in Eq. (16). It is also visible in case-1 of the Electrostatic Force analysis that the graph between force and pixel intensity is linear. By expressing the Eq. (18) in standard linear equation form, we get Eq. (19).

$$F_{(i,j)} = \left(const_1 \times \Gamma_{(i,j)}\right) + const_2 \tag{19}$$

Now, from Eq. (16) we can say that there is no constant in the sum form. So, in Eq. (19) the value of $const_2$ will become zero. By expressing $const_1$ as just *const*, we get Eq. (20).

$$F_{(i,j)} = const \times \Gamma_{(i,j)} \tag{20}$$

Therefore we can conclude that the force $F_{(i,j)}$ linearly depends on $\Gamma_{(i,j)}$ exclusively.

Theorem 1. Forces on pixels with similar intensity values and placed in different locations will have insignificant difference in magnitude.

Proof. For a change $\Delta\Gamma_{(i,j)}$ in the intensity of the $(i,j)th$ pixel $\Gamma_{(i,j)}$, it will become $(\Gamma_{(i,j)} \pm \Delta\Gamma_{(i,j)})$. We assume that the change, $\Delta\Gamma_{(i,j)}$ is insignificant as compared to the original intensity, $\Gamma_{(i,j)}$, as given in Eq. (21):

$$\Gamma_{(i,j)} >> \Delta\Gamma_{(i,j)} \tag{21}$$

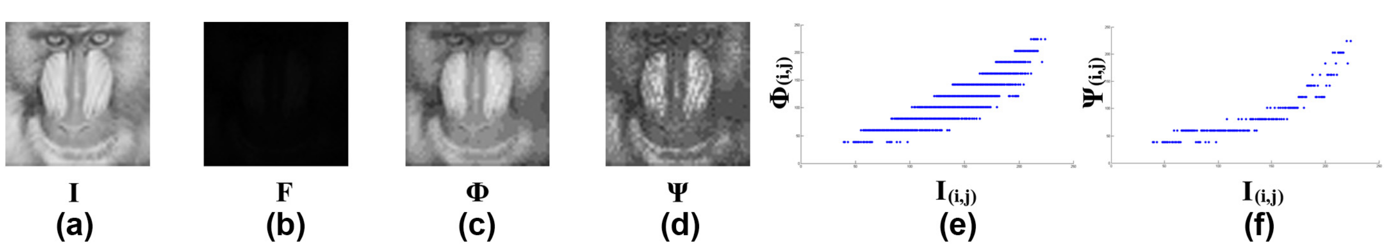

Fig. 3. (a) The original Image *I*, (b) its corresponding Force Matrix *F*, (c) Electrostatic Force Image (EF-Image) Φ of *I*, (d) Force Influence Image, Ψ, derived from *I* and Φ, (e) the graph between the intensities of Φ vs. *I*, (f) the graph between the intensities of Ψ vs. *I*.

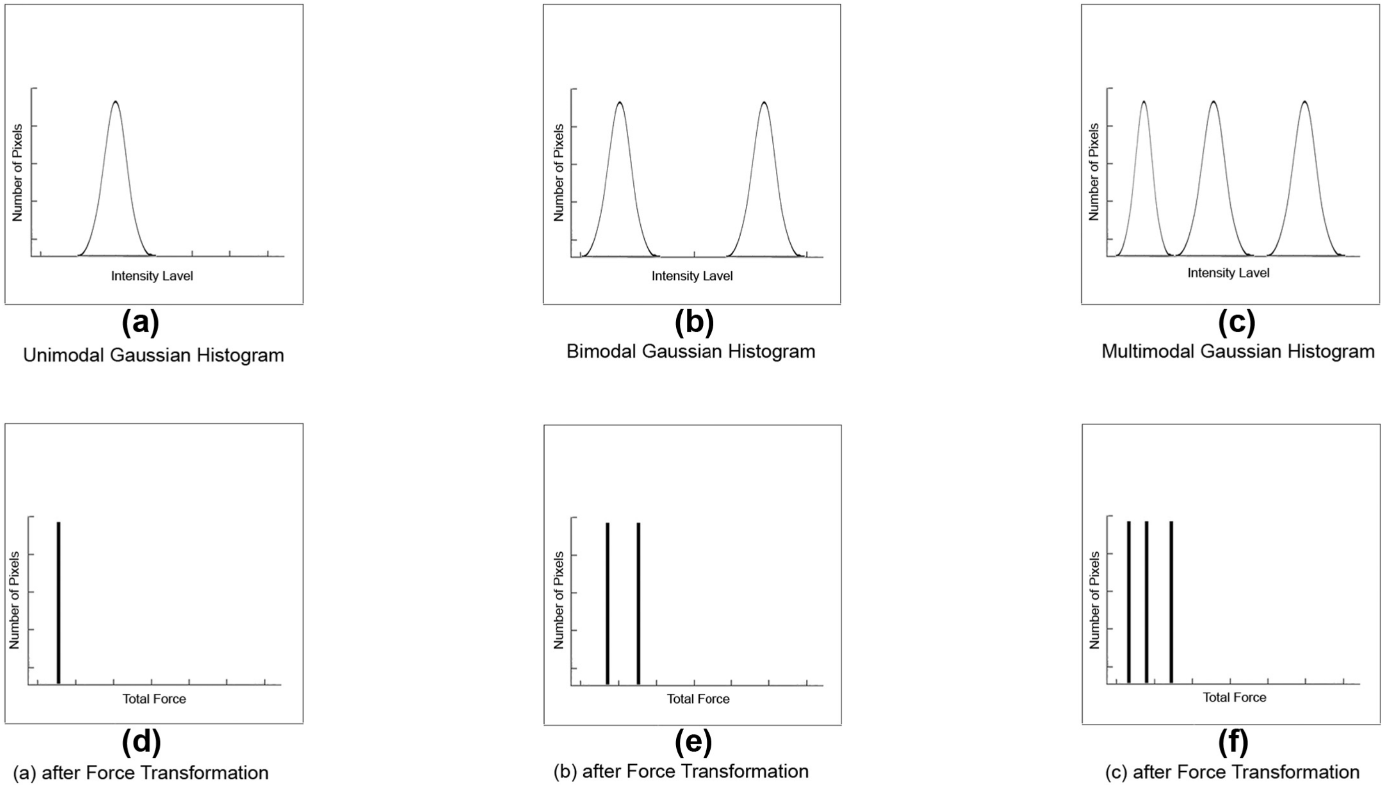

Fig. 4. The effect of EF-Index algorithm on images having unimodal, bimodal, and multimodal Gaussian histograms.

Now, from Lemma 2 we can infer that the total force $F_{(i,j)}$ on the (i,j) *th* pixel is linearly dependent on its intensity, $\Gamma_{(i,j)}$ as per Eq. (20).

Therefore, the updated force $F_{(i,j)} \pm \Delta F_{(i,j)}$ on the $(i,j)th$ pixel due to change in $\Gamma_{(i,j)}$ of Eq. (21) becomes Eq. (22).

$$\begin{aligned} F_{(i,j)} \pm \Delta F_{(i,j)} &= \left(\Gamma_{(i,j)} \pm \Delta\Gamma_{(i,j)}\right) \times const \\ &= \left(\Gamma_{(i,j)} \times const\right) \pm \left(\Delta\Gamma_{(i,j)} \times const\right) \end{aligned} \tag{22}$$

From Eqs. (20) and (22), we arrive at Eq. (23).

$$\Delta F_{(i,j)} = \pm\left(\Delta\Gamma_{(i,j)} \times const\right) \tag{23}$$

Now by dividing both sides of Eq. (23) with $F_{(i,j)}$ we arrive at Eq. (24).

$$\frac{\Delta F_{(i,j)}}{F_{(i,j)}} = \frac{\Delta\Gamma_{(i,j)} \times const}{F_{(i,j)}} \tag{24}$$

Using Eqs. (20) and (24), we arrive at Eq. (25)

$$= \frac{\Delta\Gamma_{(i,j)} \times const}{\Gamma_{(i,j)} \times const} \tag{25}$$

$$= \frac{\Delta\Gamma_{(i,j)}}{\Gamma_{(i,j)}} \tag{26}$$

Therefore, by using Eqs. (21) and (26), we get Eq. (27):

$$F_{(i,j)} >> \Delta F_{(i,j)} \tag{27}$$

The above derivation justifies that force differential at the $(i,j)th$ pixel will be insignificant with respect to the Force there when the intensity differential at the same location is insignificant to the corresponding intensity contribution.

Theorem 2. The EF-Index algorithm can correctly detect the number of cluster K present in an image.

Proof. Assuming the input image is uniformly distributed, i.e. all the neighborhood of the input image are similar. Therefore, the effect of the factors $r^2_{(i,j)(x,y)}$ in the denominator can be ignored, as it will be similar for all the pixels in the image.

As a result Eq. (5) will be modified as:

$$F_{(i,j)} = \left[\sum_{\substack{x=1 \\ (i,j)}}^{M} \sum_{\substack{y=1 \\ \neq(x,y)}}^{N} C \times \Gamma_{(i,j)} \times \Gamma_{(x,y)} \right] \tag{28}$$

Table 1
Tuning parameters of the existing algorithms.

Algorithm	Parameters
DB-Index [10]	p = 2, denotes L-2 norm i.e., Euclidian distance q = 2, denotes 2nd central moment i.e., standard deviation
I-Index [11]	Power p = 2
CVNN-Index [13]	Nearest Neighbor k = 10
DOE-AND-SCA [14]	Filter ratio ω = 3
Sym-Index [23]	knear = 2
Mean shift [31]	Bandwidth Spatial {2 h 1} = 7, Color {2 h} = 6.5
N-Cut [32]	compactness = 300
FH [33]	alpha = 4
FODPSO [34]	alpha = 0.6, current population of the swarm = 30, minimum population = 10, maximum population = 50, current number of swarms = 4, minimum number of swamps = 2, maximum number of swamps = 6

Table 2
Number of clusters determination using *EF-Index* and comparison with *DB*, *I*, *CVNN*, *Sym Indexes* over traditional images and benchmark image datasets.

Traditional images

Name		Index *DB*	*I*	*CVNN*	*Sym*	*AND*	EFI No.
Lena	KM	5	5	5	5	5	5
	FCM	5	5	5	5		
Camera man	KM	7	5	6	7	7	7
	FCM	5	5	7	7		
Barbara	KM	10	7	10	9	7	8
	FCM	9	9	7	6		
Boat	KM	11	8	7	9	8	9
	FCM	7	8	9	9		
Peppers	KM	7	7	7	7	7	7
	FCM	7	7	7	7		
House	KM	7	7	7	7	7	7
	FCM	7	7	7	7		
Airplane	KM	7	8	9	10	8	9
	FCM	11	10	10	8		
Arctichare	KM	6	7	9	9	7	8
	FCM	8	6	9	7		
Baboon	KM	7	8	10	10	7	8
	FCM	7	7	10	9		
Boy	KM	5	6	3	5	5	5
	FCM	5	7	5	6		
Flinstones	KM	9	10	9	9	9	9
	FCM	9	7	9	9		
Fruits	KM	13	13	13	13	13	13
	FCM	13	12	13	13		
Frymire	KM	7	7	7	7	7	7
	FCM	7	7	7	7		
Girl	KM	8	8	8	8	8	8
	FCM	8	8	8	8		
Goldhill	KM	9	9	9	9	9	9
	FCM	9	9	9	9		
Monarch	KM	7	7	7	7	7	7
	FCM	7	7	7	7		
Mountain	KM	9	9	9	9	9	9
	FCM	9	9	9	9		
Pool	KM	6	6	6	6	6	6
	FCM	6	6	6	6		
Sails	KM	5	5	5	5	5	5
	FCM	5	5	5	5		
Zelda	KM	5	5	5	5	5	5
	FCM	5	5	5	5		

BSDS300

Name		Index *DB*	*I*	*CVNN*	*Sym*	*AND*	EFI No
12,084	KM	6	6	6	6	6	6
	FCM	6	6	6	6		
41,033	KM	5	5	5	5	5	5
	FCM	5	5	5	4		
86,016	KM	5	7	7	5	5	5
	FCM	5	5	5	5		
101,087	KM	8	10	8	8	8	8
	FCM	8	8	8	8		
145,086	KM	8	9	8	8	8	8
	FCM	9	6	8	6		
229,036	KM	9	9	9	7	7	7
	FCM	7	7	7	7		
302,008	KM	6	6	6	6	6	6
	FCM	6	6	7	6		
385,039	KM	9	8	9	9	9	9
	FCM	11	11	9	11		
76,002	KM	7	7	7	8	7	7
	FCM	7	8	7	7		
80,099	KM	4	4	3	4	4	4
	FCM	4	4	4	4		
100,080	KM	5	4	5	7	5	5
	FCM	5	5	5	5		
118,020	KM	5	6	6	6	6	6
	FCM	6	5	6	6		
151,087	KM	8	9	8	8	9	9
	FCM	8	7	8	8		
159,045	KM	5	5	5	5	3	5
	FCM	5	5	5	5		
181,018	KM	6	6	6	6	6	6
	FCM	6	7	5	6		
198,023	KM	8	8	7	8	8	8
	FCM	8	8	8	10		
274,007	KM	7	7	7	7	7	7
	FCM	7	7	7	7		
301,007	KM	6	5	6	6	6	6
	FCM	6	6	6	6		
361,084	KM	7	6	7	6	7	7
	FCM	9	8	7	7		
374,067	KM	7	6	6	6	6	6
	FCM	4	6	6	6		

BSDS500

Name		Index *DB*	*I*	*CVNN*	*Sym*	*AND*	EFI No.
5096	KM	7	5	5	5	5	5
	FCM	5	5	5	5		
16,004	KM	6	10	8	8	8	8
	FCM	8	8	8	8		
23,050	KM	9	9	9	9	9	9
	FCM	8	9	9	10		
35,028	KM	4	4	4	4	4	4
	FCM	4	4	3	4		
41,096	KM	6	6	6	6	6	6
	FCM	6	6	6	6		
45,000	KM	4	4	3	4	4	4
	FCM	4	4	4	4		
80,085	KM	7	9	9	7	7	7
	FCM	7	9	8	7		
140,006	KM	7	7	7	7	7	7
	FCM	9	6	7	7		
306,051	KM	10	10	10	10	10	10
	FCM	10	10	11	10		
384,089	KM	8	6	8	8	10	8
	FCM	8	8	8	9		
25,098	KM	8	8	9	8	8	8
	FCM	8	8	10	8		
35,070	KM	4	4	4	4	4	4
	FCM	4	4	3	4		
71,046	KM	6	6	6	6	6	6
	FCM	8	4	6	6		
170,054	KM	9	9	9	9	9	9
	FCM	9	9	9	9		
181,079	KM	9	8	9	9	9	9
	FCM	9	9	9	9		
3096	KM	5	5	5	5	5	5
	FCM	5	5	5	4		
126,007	KM	9	9	9	9	8	9
	FCM	9	9	9	9		
253,055	KM	6	6	4	6	6	6
	FCM	8	5	6	6		
300,091	KM	9	11	9	9	9	9
	FCM	9	9	9	9		
361,010	KM	6	6	6	6	6	6
	FCM	6	6	6	6		

SBD

Name		Index *DB*	*I*	*CVNN*	*Sym*	*AND*	EFI No.
0000087	KM	7	7	7	7	7	7
	FCM	7	5	6	7		
0002395	KM	7	7	7	7	7	7
	FCM	7	7	6	7		
0003178	KM	9	9	9	9	9	9
	FCM	9	11	9	9		
0007932	KM	7	5	5	7	7	7
	FCM	9	7	7	7		
0101434	KM	6	5	5	5	5	5
	FCM	5	5	6	5		
1,000,731	KM	8	7	7	8	8	8
	FCM	8	7	9	8		
1,001,685	KM	6	6	6	6	6	6
	FCM	6	6	4	5		
2,000,042	KM	7	5	5	5	8	7
	FCM	7	8	7	7		
3,000,148	KM	5	6	6	5	6	5
	FCM	5	5	5	5		
4,100,246	KM	8	8	8	7	8	8
	FCM	6	10	8	8		
4,100,280	KM	7	9	9	10	9	9
	FCM	9	11	9	9		
5,000,119	KM	7	7	7	7	9	7
	FCM	8	7	5	7		
5,000,125	KM	6	6	6	6	6	6
	FCM	6	6	6	6		
6,000,015	KM	8	8	8	8	7	8
	FCM	6	8	8	6		
6,000,036	KM	6	7	7	5	7	7
	FCM	7	9	7	8		
8,001,155	KM	4	5	5	4	5	5
	FCM	5	5	5	5		
8,003,131	KM	9	11	11	9	9	9
	FCM	9	9	9	9		
9,002,090	KM	6	6	6	6	6	6
	FCM	6	6	6	6		
9,003,635	KM	10	10	10	10	10	10
	FCM	10	10	10	10		
9,005,105	KM	12	10	10	10	10	10
	FCM	10	10	10	10		

Table 3
Comparison of traditional BSDS300, BSDS500, and SBD datasets.

Data set		Similar results with EF-Index				
		DB Index	*I Index*	*CVNN Index*	*Sym Index*	*DOE-AND-SCA*
Traditional images	KM	80.77%	65.38%	76.92%	84.62%	80.77%
	FCM	76.92%	61.54%	84.62%	80.77%	
BSDS300	KM	85%	70%	82%	82%	84%
	FCM	81%	73%	79%	81%	
BSDS500	KM	86%	74%	82%	81%	85%
	FCM	82%	75%	80%	83%	
SBD	KM	82.24%	75.10%	80%	78.04%	79.16%
	FCM	84.06%	77.20%	75.10%	84.06%	

Table 4
Quantitative segmentation evaluation for BSDS300.

	PRI	VOI	GCE	BDE
Mean shift	0.685	2.720	0.308	17.968
N-Cut	0.554	**2.570**	**0.181**	35.449
FH	0.686	3.405	0.307	20.814
FODPSO	0.631	3.746	0.491	20.089
EF-Index (Fuzzy C-means)	**0.763**	3.484	0.348	**11.403**
EF-Index (*K*-means)	**0.762**	3.616	0.359	**11.403**

Table 5
Quantitative segmentation evaluation for BSDS500.

	PRI	VOI	GCE	BDE
Mean shift	0.693	3.248	0.311	17.146
N-Cut	0.553	**2.577**	**0.181**	34.063
FH	0.692	3.441	0.310	19.857
FODPSO	0.638	3.787	0.500	19.187
EF-Index (Fuzzy C-means)	**0.770**	3.518	0.362	**10.988**
EF-Index (*K*-means)	**0.770**	3.639	0.371	**10.988**

The force on the $(i,j)th$ pixel will be:

$$F_{(i,j)} = \left[\Gamma_{(i,j)} \times \left(\sum_{\substack{x=1\\(i,j)}}^{M} \sum_{\substack{y=1\\ \neq (x,y)}}^{N} C \times \Gamma_{(x,y)}\right)\right] \tag{29}$$

For any $(M \times N)$ image the quantity $\left(\sum_{\substack{x=1\\(i,j)}}^{M} \sum_{\substack{y=1\\ \neq (x,y)}}^{N} C \times \Gamma_{(x,y)}\right)$ will be similar for all the pixels in the image. Therefore, considering $\left(\sum_{\substack{x=1\\(i,j)}}^{M} \sum_{\substack{y=1\\ \neq (x,y)}}^{N} C \times \Gamma_{(x,y)}\right) = const_1$, we get:

$$F_{(i,j)} = \left[\Gamma_{(i,j)} \times const_1\right] \tag{30}$$

Therefore, after force transformation the force on the $(i,j)th$ pixel will be proportional to its original intensity $\Gamma_{(i,j)}$. After converting the force $F_{(i,j)}$ to the nearest integer, it will go into the group of the pixels having similar intensity as the $(i,j)th$ pixel.

Now if we consider the histogram of the input image as a single modal Gaussian curve, i.e. the histogram have only one group or cluster of pixels. Then after force transformation all the pixels will come into one single value. Let's consider that value to be μ, representing mean of the Gaussian curve. After force transformation all original intensity the values in the 6σ range of the Gaussian curve will be represented by the final value μ.

Similarly for an image having multi Gaussian histogram with K modes, which are separated by a distance $d > 6\sigma$, will generate K different μs representing the K modes of the original input image.

Therefore, we can say that the proposed EF-Index algorithm can correctly detect the number of clusters K present in an image.

Force exerted on similar pixels should be very close barring two exceptions (i) effect due to difference in the values of the surrounding pixels, and (ii) distance between the pixels. Effect due to dissimilarities in pixels is mitigated to a large extent by rounding off the force computed for any pixel intensity. Effect due to distance between two pixels is eliminated by taking the corresponding minimum force value for a set of similar pixels. Clustering is occurring as a result of compaction of corresponding pixels into same force value.

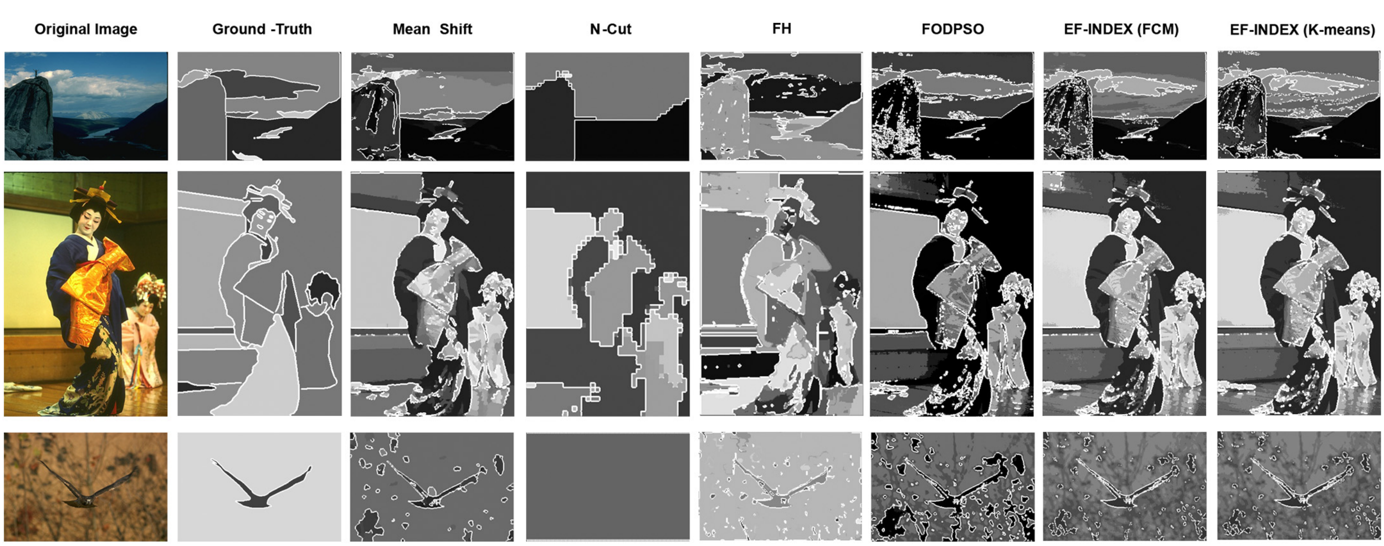

Fig. 5. Image segmentation results from Berkeley Segmentation Data Set (BSDS300 and BSDS500).

Table 6
Quantitative segmentation evaluation for SBD.

	PRI	VOI	GCE	BDE
Mean shift	0.642	2.318	**0.169**	16.430
N-Cut	0.746	**2.149**	0.197	11.706
FH	0.755	6.575	0.431	12.184
FODPSO	0.743	3.292	0.422	**9.416**
EF-Index (Fuzzy C-means)	**0.764**	3.885	0.481	10.031
EF-Index (*K*-means)	**0.762**	4.012	0.491	10.075

3.3. Determination of appropriate number of quantization levels (EF-Index) of input image

Force Influence Image, Ψ, discussed in Section 3.2, will contain number of quantization levels. The process to construct Ψ and finding out of quantization levels from input image I, is shown in Fig. 3. The Input image, I, is shown in Fig. 4. (a); the image shown in Fig. 4. (b) is the Force matrix, F, obtained from I; and Fig. 4. (c) is the *EF-Image*, Φ of I. The generation of EF-Image from input image is given in Algorithm 1.

The Force Influence Image, Ψ, is depicted in Fig. 4. (d). The graph of intensities of Φ against those of I is shown in Fig. 4. (e). It may be noted that, separate but close intensities in I concentrate into a single intensity value in Φ due to the effect of rounding-off the force values as per Eq. (5). As a result, similar intensities are assigned with same force value, as apparent in the graph Fig. 4. (e). Moreover, the phenomenon of computing the influence of EF-Image, as discussed earlier, has got close resemblance to *quantization* of intensities of any arbitrary input gray image. For any single intensity value in I, there could be several location-corresponding different intensity values in Φ. To avoid redundancies thus obtained, the minimum of all such intensities is stored in corresponding locations of a new image, named as Force Influence image, Ψ. Due to removal of redundancies some of the influence points present in Fig. 4. (e) will be mapped to other lower levels, which is shown in the Ψ vs. I graph of Fig. 4. (f). As a result, there is a possibility that the number of levels present in Φ vs. I graph may decrease in the Ψ vs. I graph, which will eventually produce optimum number of quantization levels i.e. EF-Index. The process to generate EF-Index from Input Image and its corresponding EF-Image is detailed in Algorithm 2.

The count of such different intensity levels in the Force Influence image, Ψ will eventually turn-out to be the number of clusters, K, as per Eq. (11) inherent in the original image I.

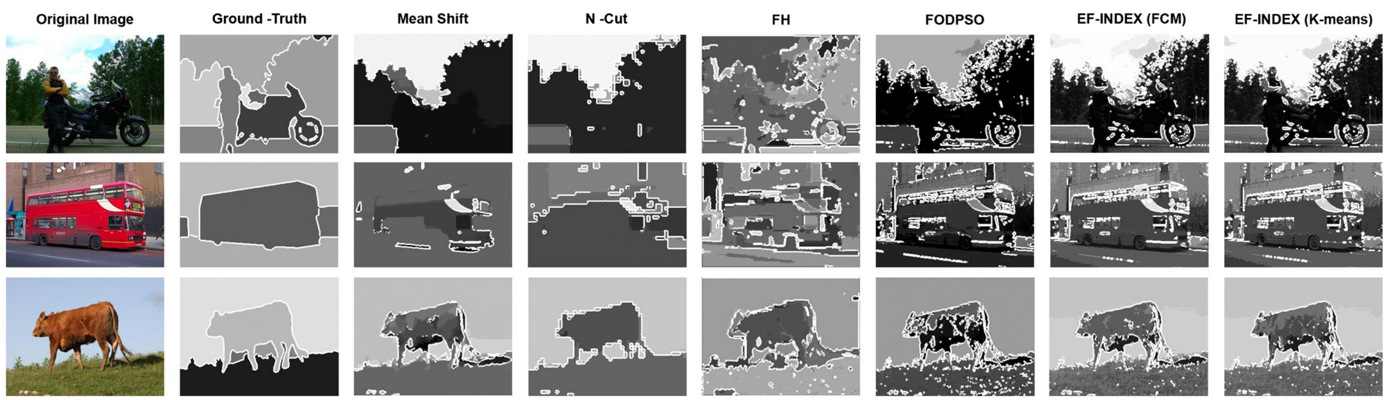

Fig. 6. Image segmentation results from Stanford Background Data Set (SBD).

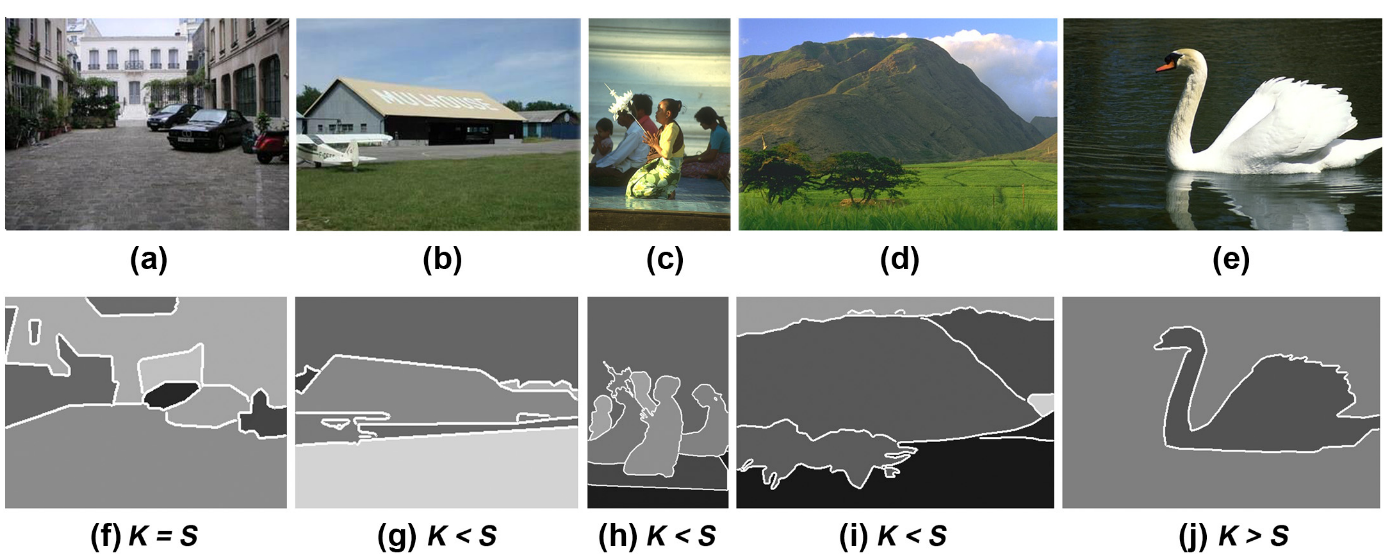

Fig. 7. Comparison between the number of segments (S) in the Ground-Truth and number of clusters (K) determined by the EF-Index for natural images. (a) Natural Image from SBD dataset and (f) its Ground-Truth. Here, $K = 10$; $S = 10$. (b) Another Natural Image from SBD dataset with Ground-Truth (g). Here, $K = 6$; $S = 10$. (c) Natural Image and Ground-Truth (h) from BSDS500 dataset. Here, $K = 16$; $S = 17$. (d) Natural Image and Ground-Truth (i) from BSDS500 dataset. Here, $K = 7$; $S = 9$. (e) Natural Image and Ground-Truth (j) from BSDS500 dataset. Here, $K = 10$; $S = 2$. Therefore, in (a) $K = S$, then in (b), (c), and (d) $K < S$, and in (e) $K > S$.

Algorithm 1 is used to compute EF-Image, Φ. This EF-Image is used to generate the Force Influence Image, Ψ. The number of gray levels present in Ψ is calculated and reported as *K*. Incidentally, *K* corresponds to the number of clusters present in the original image *I*, which is the final output in the form of EF-Index produced by Algorithm 2. It is noteworthy that the above algorithms are free of user intervention in any form.

4. Experimental results

We conducted experiments on twenty five traditional images as they are considered ideal by the image processing community [43] to analyze the performance of any algorithm and benchmark datasets such as Berkeley Segmentation Data Sets viz. BSDS300 and BSDS500 [30], and the Stanford Background Data Set SBD [31] containing 300, 500, and 715 images respectively.

The experiments conducted to evaluate the EF-Index algorithm are divided into two parts. First, we analyzed the number of clusters yielded by the proposed algorithm. For that we matched the results produced by the EF-Index algorithm with the existing state-of-the-art algorithms viz. *DB-Index* [10], *I-Index* [11], *CVNN-Index* [13], DOE-AND-SCA [14], and *Sym-Index* [23] for computing the correct value of *K*.

In the second part of the experiment, we evaluated clustering-based image segmentation results for different algorithms. The state-of-the-art clustering algorithms like the *K-means* [7], and *Fuzzy C-means* [8] are applied to perform the clustering-based image segmentation on the gray level version of the images, taking the EF-Index as requisite input i.e. the number of clusters. The results of segmentation using EF-Index are compared with existing state-of-the-art segmentation algorithms viz. Mean-shift [31], N-Cut [32], FH [33], and FODPSO [34]. Their performances are measured using the standard segmentation performance measures, such as *Probabilistic Rand Index* (*PRI*) [40], *Variation of Information* (*VOI*) [41], *Global Consistency measure* (*GCE*) [44] and *Boundary Displacement Error* (*BDE*) [42]. Higher value of *PRI* and lower value of each of *VOI*, *GCE*, and *BDE* indicates better segmentation. The MATLAB source code for *PRI* [40], *VOI* [41], *GCE* [44], and *BDE* [42] is kindly given on-line at http://www.eecs.berkeley.edu/~yang/software/lossy_segmentation/.

Algorithm 1. EF-Image generation algorithm.

Input: I_{MN}

Initialize: F_{MN} , Φ_{MN}

for x = *1……M*
  for y = *1……N*
    $\Gamma_{(i,j)} \leftarrow I_{(i,j)} / 255$
  end for
end for
for x = *1……M*
  for y = *1……N*
    $F_{(i,j)} \leftarrow$ *Electrostatic Force* on $\Gamma_{(i,j)}$ using equation (10)
  end for
end for
$\Phi_{MN} \leftarrow$ Adjusted values of F_{MN} applying equation (11)

Output: Φ_{MN}, Γ_{MN}

Algorithm 2. Electrostatic Force Index determination algorithm.

Input: Γ_{MN} , Φ_{MN}

Initialize: $\eta \leftarrow$ *NULL set* , $\sigma \leftarrow$array,
$\theta \leftarrow NULL$, $\Omega_{MN} \leftarrow$ 2D array, $K \leftarrow NULL$, $\Psi_{MN} \leftarrow$ 2D array.
// η stores the coordinate values from Γ
// σ stores the intensity values from Φ
// K preserves the *Electrostatic Force – Index (EF-Index)* as the final output.

for *i = 0………255*
  $\theta \leftarrow NULL$
  $\eta \leftarrow$ Coordinate of all the pixels in Γ having intensity = *i*
  $\sigma \leftarrow \Phi_{(x,y)}, \forall (x,y) \in \eta$
  $\theta \leftarrow$ minimum of σ
  if $\theta \neq NULL$
    $\Psi_{(x,y)} \leftarrow \theta, \forall (x,y) \in \eta$
    $\Omega \leftarrow \Omega \cup \{\theta\}$
  end if
end for
$K \leftarrow$ Number of gray-levels present in Ω

Output: K

As shown by Yang et al. in [35], the segmentation performance measure of PRI is highly correlated with the Ground Truths produced by experts. The performance measure of *GCE* [44] does not restrict over-segmentation by setting the penalty for it. It generates high score if each pixel in an image is considered as a separate segment. Sometimes three complementary measures *VOI* [41], *GCE* [44], and *BDE* [42] produce high score for unrealistic bad segmentations [38]. All the techniques have to be contemplated simultaneously in order to ascertain the effectiveness of any segmentation algorithm.

The consecutive sections will discuss these experiments and their results in details.

4.1. Experimental setup

All the experiments were conducted using MATLAB R2013a environment on a computer with Intel Core i5 3.20 GHz CPU and 4 GB memory running Windows 8.1.

4.2. Comparison with state-of-the-art algorithms for determination of number of clusters

In this section, we compare the output of the EF-Index with the outputs of state-of-the-art algorithms for number of clusters determination [10,11,13,14,23]. The parameters used by these methods are mentioned in Table 1. The Traditional image set contains 25 natural images. The three image datasets BSDS300, BSDS500, SBD containing 300, 500, 715 images respectively are used in this experiment. Table 3 shows the overall percentage of similarity between results of the EF-Index algorithm and the existing state-of-the-art algorithms for determination number of clusters. Results obtained for some of the randomly selected individual images are depicted in Table 2. Table 3 reveals that output generated by our proposed algorithm matches significantly with the state-of-the-art indexes [10,11,13,14,23]. In terms of similarity, the best result is obtained for BSDS300 and BSDS500 datasets with *I-Index*. From Table 2, it can be said that our EF-Index algorithm determines the correct number of clusters for any natural image accurately, with no external intervention whatsoever.

4.3. Clustering-based image segmentation assisted by the EF-Index algorithm

In second part of experiment, we evaluated the clustering-based segmentation performed by the clustering algorithms [7,8], using the number of clusters, derived by the proposed EF-Index algorithm. The comparisons are done with state-of-the-art segmentation algorithms [31–34] using the segmentation evaluation measures [40–42]. Table 1 represents the parameters used by these methods.

Next, we shall briefly discuss about the datasets and compare the segmentation results obtained from them.

5. Berkeley Segmentation Dataset (BSDS300 and BSDS500)

The benchmarked image segmentation dataset, Berkeley Segmentation Dataset BSDS300 consists of 300 benchmarked Natural Images divided into 100 test and 200 train images. Whereas, the BSDS500 dataset extends the BSDS300 by adding another 200 images. One of the characteristics of the Berkeley dataset is that they provide multiple ground-truths for each of the image. They are generated by multiple human experts and are applied for appraising the performance of any segmentation algorithm.

In our experiment, we use four segmentation evaluation criteria, in which *PRI* (*Probabilistic Rand Index*) [40] evaluates the probabilistic performance of an algorithm by comparing with multiple ground truths. For other three measures [41,42] all the ground truths have been taken into account by considering the average performance corresponding to any image. Based on these four criteria, performances of different algorithms for BSDS300 and BSDS500 are presented in Tables 4 and 5 respectively.

From Tables 4 and 5 it can be said that, the segmentation results produced by [7,8] conducted on the basis of EF-Index outperforms the cutting edge segmentation algorithms [31–34]. The PRI values generated by our method are much higher as compared to other algorithms. This observation establishes superiority of our proposed method. However, in the cases of *VOI* and *GCE* the Mean shift [31], N-Cut [32], FH [33] algorithms generate better results than the present method. Fig. 5 depicts segmentation outcome of some sample images from two dataset mentioned above.

6. Stanford Background Dataset (SBD)

This benchmarked image segmentation dataset, named as Stanford Background Dataset comprises 715 natural images. Each image is provided with a text file, which contains labels corresponding to each pixel in them. The overall segmentation performance for all the segmentation algorithms [31–34] is shown in Table 6.

In this case, better results are observed form the combination of our EF-Index algorithm with clustering algorithms [7,8]. The PRI values are higher for our proposed technique, indicating better performance. However, regarding VOI [41], everyone else performed better except FH [32]. For GCE [44] also other algorithms viz. [31–34] outperformed our proposed one. In case of BDE [42] only FODPSO [34] produces more impressive results. Fig. 6 depicts segmentation outcome of some sample images from SBD.

Table 7
Comparison between number of segments (S) present in Ground-Truth and corresponding number of clusters (K) determined by EF-Index.

Dataset	$K<S$	$K=S$	$K>S$
BSDS300	72.34%	4.62%	23.04%
BSDS500	74.15%	4.41%	21.44%
SBD	76.36%	4.20%	19.44%

6.1. Comparison between the number of segments (S) of the Ground-Truth and the corresponding number of clusters (K) determined by proposed EF-Index

This section compares the number of segments (S) present in the expert generated Ground-Truths provided in the BSDS300, BSDS500, and SBD datasets, with the Number of Clusters (K) determined by EF-Index from corresponding Natural Images. Some of the examples are shown in Fig. 7. The overall comparison is provided in Table 7.

From Table 7 it is clear that usually the Number of Cluster (K) determined by the EF-Index algorithm is less than or equal to that of the number of segments (S) present in the Human Segmented Ground-Truths. It is consistent to the argument of Section 2.1, indicating justifiable number of clusters (K) determination by the EF-Index.

7. Conclusion

This paper addresses a parameter-free algorithm which can determine the number of clusters K and hence the number of segments S present in an image automatically. The EF-Index algorithm is effortless to understand and implement, and it generates promising outcomes for verities of images including the benchmark image datasets BSDS300 [30], BSDS500 [30], and SBD [31]. The number of clusters predicted by the proposed method closely correlates with those of the state-of-the-art algorithms [10,11,13,14,23].

The only limitation of the proposed algorithm is its high noise sensitivity. The number of cluster outcome may differ depending on the presence of noise in the image.

Designing a noise immune number of cluster determination algorithm would be an interesting task to be accomplished in future.

Conflict of interest

None.

Acknowledgements

The authors would like to express their obligation to 'Biometrics Laboratory' and Department of Computer Science & Engineering, Tripura University (A Central University), Suryamaninagar, Agartala, Tripura (West) for facilitating this research.

References

[1] L. Ladický, C. Russell, P. Kohli, P.H.S. Torr, Associative hierarchical CRFs for object class image segmentation, 2009 IEEE 12th International Conference on Computer Vision 2009, pp. 739–746 , (Kyoto).
[2] S. Gould, R. Fulton, D. Koller, Decomposing a scene into geometric and semantically consistent regions, 2009 IEEE 12th International Conference on Computer Vision 2009, pp. 1–8 , (Kyoto).
[3] M.P. Kumar, D. Koller, Efficiently selecting regions for scene understanding, 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2010, pp. 3217–3224 , (San Francisco, CA).
[4] D. Hoiem, A.A. Efros, M. Hebert, Recovering surface layout from an image, Int. J. Comput. Vis. 75 (2007) 151–172.
[5] B. Liu, S. Gould, D. Koller, Single image depth estimation from predicted semantic labels, 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2010, pp. 1253–1260 , (San Francisco, CA).
[6] S. Banks, Signal Processing, Image Processing and Pattern Recognition, Prentice-Hall, Englewood Cliffs, NJ, 1990.
[7] S.P. Lloyd, Least squares quantization in PCM, IEEE Trans. Inf. Theory IT-28 (2) (Mar. 1982) 129–136.
[8] J.C. Bezdek, R.J. Hathaway, M.J. Sabin, W.T. Tucker, Convergence theory for fuzzy c-means: counterexamples and repairs, IEEE Transactions on Systems, Man, and Cybernetics, vol. 17, no. 5, Sept.-Oct. 1987, pp. 873–877.
[9] T. Kanungo, D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman, A.Y. Wu, An efficient k-means clustering algorithm: analysis and implementation, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, Jul 2002, pp. 881–892.

[10] D.L. Davies, D.W. Bouldin, A cluster separation measure, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-1, no. 2, April 1979, pp. 224–227.
[11] U. Maulik, S. Bandyopadhyay, Performance evaluation of some clustering algorithms and validity indices, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 12, Dec 2002, pp. 1650–1654.
[12] J.R. Claycomb, Applied electromagnetics using Quickfield™ & MATLAB (engineering), 81, Jones and Bartlett, 2010.
[13] Y. Liu, Z. Li, H. Xiong, X. Gao, J. Wu, S. Wu, Understanding and enhancement of internal clustering validation measures, IEEE Transactions on Cybernetics, Vol. 43, No. 3, June 2013, pp. 982–994.
[14] J. Chen, Y. Wu, X. Lin, Q. Xuan, DOE-AND-SCA: a novel SCA based on DNN with optimal eigenvectors and automatic cluster number determination, IEEE Access 6 (2018) 20764–20778.
[15] M. Lagrange, L.G. Martins, J. Murdoch, G. Tzanetakis, Normalized cuts for predominant melodic source separation, IEEE Transactions on Audio, Speech, and Language Processing, vol. 16, no. 2, Feb. 2008, pp. 278–290.
[16] R. Krishnapuram, A. Joshi, O. Nasraoui, L. Yi, Low-complexity fuzzy relational clustering algorithms for Web mining, IEEE Transactions on Fuzzy Systems, vol. 9, no. 4, Aug 2001, pp. 595–607.
[17] J. Ma, W. Xu, Y.H. Sun, E. Turban, S. Wang, O. Liu, An ontology-based text-mining method to cluster proposals for research project selection, IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, Vol. 42, No. 3, May 2012, pp. 784–790.
[18] R.F. Ling, A computer generated aid for cluster analysis, Comm. ACM 16 (1973) 355–361.
[19] G.D. Floodgate, P.R. Hayes, The Adansonian taxonomy of some yellow pigmented marine bacteria, J. Gen. Microbiol. 30 (1963) 237–244.
[20] R.B. Cattell, A note on correlation clusters and cluster search methods, Psychometrika 9 (3) (1944) 169–184.
[21] J.C. Bezdek, R. Hathaway, VAT: a tool for visual assessment of (cluster) tendency, Proc. Int'l Joint Conf. Neural Networks (IJCNN '02) 2002, pp. 2225–2230.
[22] S. Bandyopadhyay, U. Maulik, A. Mukhopadhyay, Multiobjective genetic clustering for pixel classification in remote sensing imagery, IEEE Transactions on Geoscience and Remote Sensing, vol. 45, no. 5, May 2007, pp. 1506–1511.
[23] S. Saha, S. Bandyopadhyay, Application of a new symmetry-based cluster validity index for satellite image segmentation, IEEE Geoscience and Remote Sensing Letters, vol. 5, no. 2, April 2008, pp. 166–170.
[24] L. Wang, C. Leckie, K. Ramamohanarao, J. Bezdek, Automatically determining the number of clusters in unlabeled data sets, IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 3, March 2009, pp. 335–350.
[25] Y. Li, M. Popescu, K.C. Ho, Improving automatic sound-based fall detection using iVAT clustering and GA-based feature selection, 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society 2012, pp. 5867–5870 , (San Diego, CA).
[26] J. Huband, J.C. Bezdek, R. Hathaway, bigVAT: visual assessment of cluster tendency for large data sets, Pattern Recogn. 38 (11) (2005) 1875–1886.
[27] R. Hathaway, J.C. Bezdek, J. Huband, Scalable visual assessment of cluster tendency, Pattern Recogn. 39 (2006) 1315–1324.
[28] J.C. Bezdek, R.J. Hathaway, J. Huband, Visual assessment of clustering tendency for rectangular dissimilarity matrices, IEEE Trans. Fuzzy Systems 15 (5) (2007) 890–903.
[29] P. Berkhin, Survey of clustering data mining techniques, Accrue Software, San Jose, CA, 2002.
[30] D. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001, vol.2, 2001, pp. 416–423 , (Vancouver, BC).
[31] D. Comaniciu, P. Meer, Mean shift: a robust approach toward feature space analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 5, May 2002, pp. 603–619.
[32] Jianbo Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 22, no. 8, Aug 2000, pp. 888–905.
[33] P. Felzenszwalb, D. Huttenlocher, Efficient graph-based image segmentation, Int. J. Comput. Vis. 59 (2004) 167–181.
[34] P. Ghamisi, M.S. Couceiro, F.M.L. Martins, J.A. Benediktsson, Multilevel image segmentation based on fractional-order Darwinian particle swarm optimization, IEEE Transactions on Geoscience and Remote Sensing, vol. 52, no. 5, May 2014, pp. 2382–2394.
[35] A.Y. Yang, J. Wright, S. Sastry, Y. Ma, Unsupervised segmentation of natural images via lossy data compression, Comput. Vis. Image Understand. (2007) 212–225.
[36] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addison Wesley, 1993 458.
[37] A. Hoover, et al., An experimental comparison of range image segmentation algorithms, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 7, July 1996, pp. 673–689.
[38] M. Mignotte, Segmentation by fusion of histogram-based *K*-means clusters in different color spaces, IEEE Transactions on Image Processing, vol. 17, no. 5, May 2008, pp. 780–787.
[39] R.C. Gonzalez, R.E. Woods, Digital Image Processing, Addison Wesley, 1993 31–38.
[40] R. Unnikrishnan, C. Pantofaru, M. Hebert, Toward objective evaluation of image segmentation algorithms, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, June 2007, pp. 929–944.
[41] M. Meila, Comparing clusterings — an axiomatic view, Proc. 22nd Int. Conf. Machine Learning 2005, pp. 577–584.
[42] J. Freixenet, X. Munoz, D. Raba, J. Marti, X. Cufi, Yet another survey on image segmentation: region and boundary information integration, Proc. 7th Eur. Conf. Computer Vision Part III, Copenhagen, Denmark May 2002, pp. 408–422 , (LNCS).
[43] D.C. Munson, A note on Lena, IEEE Trans. Image Process. 5 (1996) 3.
[44] A. Jain, Fundamentals of Digital Image Processing, vol. 83, Prentice-Hall, 1989.
[45] L.A.F. Park, J.C. Bezdek, C. Leckie, R. Kotagiri, J. Bailey, M. Palaniswami, Visual assessment of clustering tendency for incomplete data, IEEE Transactions on Knowledge and Data Engineering, vol. 28, No. 12, Dec. 1 2016, pp. 3409–3422.
[46] I.E. Evangelou, D.G. Hadjimitsis, A.A. Lazakidou, C. Clayton, Data mining and knowledge discovery in complex image data using artificial neural networks, Proc. Workshop Complex Reason. Geogr. Data, 2001 , (Paphos, Cyprus).
[47] D. Kumar, J.C. Bezdek, M. Palaniswami, S. Rajasegarar, C. Leckie, T.C. Havens, A hybrid approach to clustering in big data, IEEE Transactions on Cybernetics, vol. 46, no. 10, Oct. 2016, pp. 2372–2385.