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

Mrinal Kanti Bhowmik a, Tathagata Deb Nath a, Debotosh Bhattacharjee b, Paramartho 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

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 segments 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.

© 2019 Elsevier B.V. All rights reserved.
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 number 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 pre-processing 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, BSDSS500, 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 in night time at several atmospheric conditions. To characterize the texture, we have estimated entropy value where the visual frames shown in night time at several atmospheric conditions.

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 ≤ 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.
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:

\[
K = \text{COUNT}\left[\min(\Phi_{i,j})|\Phi_{i,j} = p\} \right]_{p=0,\ldots,255}
\]  

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

\[
\overrightarrow{F}_{AB} = \frac{C \times q_A \times q_B}{r_{AB}^2}
\]  

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).

\[
F_{PQ} = \frac{C \times I_p \times I_q}{r_{PQ}^2}
\]  

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 \( \Gamma \).

\[
\Gamma_{i,j} = \frac{I_{i,j}}{255}
\]  

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. Electrostatic Force \( \Phi_{i,j} \) on any \((i,j)\)th pixel of an image, \( \Gamma \), having \((M \times N)\) pixels will be:

\[
\Phi_{i,j} = \sum_{x=1}^{M} \sum_{y=1}^{N} \left[ \frac{C \times \Gamma_{i,j} \times \Gamma_{x,y}}{r_{(i,j)(x,y)}^2} \right]
\]  

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 \( \Gamma \). \( r_{(i,j)(x,y)} \) 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), \( \Phi \), is constructed, using Eq. (6), from the corresponding Force Matrix, \( F \), obtained from the given image, \( I \).

\[
\Phi_{i,j} = \left\lfloor \frac{255}{F_{\text{max}} - F_{\text{min}}} \times \left( F_{i,j} - F_{\text{min}} \right) \right\rfloor
\]  

where, \( F_{\text{max}} \) and \( F_{\text{min}} \) 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 \( \Phi \) will be invariant of any choice of \( C \). The influence of the Electrostatic Force Image, \( \Phi \), on the corresponding input image \( I \) is represented as the Force Influence Image, \( \Psi \), given by Eqs. (7) to (9):

\[
\Theta_p = \Theta_p \cup \{\Phi_{i,j}\}, \quad \forall \Phi_{i,j} = p
\]  

Fig. 2. (a) Two static point charges A and B, having distance \( r_{AB} \) exerting forces \( \left(F_{A,P}, F_{B,A}\right) \) 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 \( P \), in the normalized image is stored in the corresponding location of the Electrostatic Force image.
The set $\Theta_p$, which is initially NULL, stores the influences of Electrostatic Force on all the pixels of input image $I$ having the intensity value $p$.

$$\theta_p = \min\{\Theta_p\} \mid \theta_p \neq 0$$

(8)

For any intensity level, $p$, present in $I$, the minimum of all location-corresponding intensity values from $\Theta_p$ is stored in $\theta_p$.

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

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

(9)

To compute Electrostatic Force Index (EF-Index) all the existing values of $\theta_p$ are stored in set, $\Omega$, given in Eq. (10).

$$\Omega = \{x | x = \theta_p0 \leq p \leq 255\}$$

(10)

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

$$K = |\Omega|$$

(11)

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

To compile together, the final intensity $\Psi_{(i,j)}$ corresponding to any initial intensity $I_{(i,j)}$ is set preferably at the minimum and stored in $\Psi$, 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 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 $\Phi$ to the minimum and stored in $\Psi$, 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).

$$F_{(i,j)} = C \times \Gamma_{(i,j)} \times \sum_{x=1}^{M} \sum_{y=1}^{N} \frac{\Gamma_{(x,y)}}{\Gamma_{(i,j)(x,y)}}$$

(13)

We can assume, Eq. (14).

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

(14)

which is the cumulated intensity of the image.

Now considering Eq. (15).

$$d_{(i,j)} = \sum_{x=1}^{M} \sum_{y=1}^{N} \frac{1}{\Gamma_{(i,j)(x,y)}}$$

(15)

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

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

(16)

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

$$F_{(i,j)} = f(\phi_{(i,j)}, S, d_{(i,j)})$$

(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 $r_{(i,j)(x,y)}$. 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).

$$F_{(i,j)} = f_1(\Gamma_{(i,j)})$$

(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).

$$F_{(i,j)} = (\text{const} \times \Gamma_{(i,j)}) + \text{const}_2$$

(19)

Now, from Eq. (16) we can say that there is no constant in the sum form. So, in Eq. (19) the value of $\text{const}_2$ will become zero. By expressing $\text{const}_1$ as just $\text{const}$, we get Eq. (20).

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

(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):

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

(21)

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


33
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 \( F(i,j) + \Delta F(i,j) \) on the \((i,j)\)th pixel due to change in \( \Gamma(i,j) \) of Eq. (21) becomes Eq. (22).

\[
F(i,j) + \Delta F(i,j) = (\Gamma(i,j) + \Delta \Gamma(i,j)) \times \text{const} \\
= (\Gamma(i,j) \times \text{const}) + (\Delta \Gamma(i,j) \times \text{const})
\]

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

\[
\Delta F(i,j) = \pm (\Delta \Gamma(i,j) \times \text{const})
\]

Now by dividing both sides of Eq. (23) with \( F(i,j) \) we arrive at Eq. (24).

\[
\frac{\Delta F(i,j)}{F(i,j)} = \frac{\Delta \Gamma(i,j) \times \text{const}}{\Gamma(i,j) \times \text{const}}
\]

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

\[
= \frac{\Delta \Gamma(i,j)}{\Gamma(i,j)} 
\]

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

\[
F(i,j) \geq \Delta F(i,j)
\]

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:

\[
F(i,j) = \left[ \sum_{x=1}^{M} \sum_{y=1}^{N} C \times \Gamma(i,j) \times \Gamma(x,y) \right] 
\]
<table>
<thead>
<tr>
<th>Name</th>
<th>Index</th>
<th>EFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lena</td>
<td>KM</td>
<td>5</td>
</tr>
<tr>
<td>Camera man</td>
<td>FCM</td>
<td>5</td>
</tr>
<tr>
<td>Barbara</td>
<td>KM</td>
<td>10</td>
</tr>
<tr>
<td>Boat</td>
<td>KM</td>
<td>11</td>
</tr>
<tr>
<td>Peppers</td>
<td>KM</td>
<td>7</td>
</tr>
<tr>
<td>House</td>
<td>KM</td>
<td>7</td>
</tr>
<tr>
<td>Airplane</td>
<td>KM</td>
<td>7</td>
</tr>
<tr>
<td>Arctichare</td>
<td>KM</td>
<td>6</td>
</tr>
<tr>
<td>Baboon</td>
<td>KM</td>
<td>7</td>
</tr>
<tr>
<td>Boy</td>
<td>KM</td>
<td>5</td>
</tr>
<tr>
<td>Flinstones</td>
<td>KM</td>
<td>5</td>
</tr>
<tr>
<td>Boy</td>
<td>KM</td>
<td>5</td>
</tr>
<tr>
<td>Frymire</td>
<td>KM</td>
<td>7</td>
</tr>
<tr>
<td>Girl</td>
<td>KM</td>
<td>8</td>
</tr>
<tr>
<td>Goldhill</td>
<td>KM</td>
<td>9</td>
</tr>
<tr>
<td>Monarch</td>
<td>KM</td>
<td>7</td>
</tr>
<tr>
<td>Mountain</td>
<td>KM</td>
<td>9</td>
</tr>
<tr>
<td>Pool</td>
<td>KM</td>
<td>9</td>
</tr>
<tr>
<td>Sails</td>
<td>KM</td>
<td>5</td>
</tr>
<tr>
<td>Zelda</td>
<td>KM</td>
<td>5</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Traditional images</th>
<th>BSDS300</th>
<th>BSDS500</th>
<th>SBD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Index</td>
<td>EFS No.</td>
<td>Name</td>
</tr>
<tr>
<td>Lena</td>
<td>KM</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Camera man</td>
<td>FCM</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Barbara</td>
<td>KM</td>
<td>10</td>
<td>7</td>
</tr>
<tr>
<td>Boat</td>
<td>KM</td>
<td>11</td>
<td>8</td>
</tr>
<tr>
<td>Peppers</td>
<td>KM</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>House</td>
<td>KM</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Airplane</td>
<td>KM</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>Arctichare</td>
<td>KM</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>Baboon</td>
<td>KM</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>Boy</td>
<td>KM</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>Flinstones</td>
<td>KM</td>
<td>5</td>
<td>7</td>
</tr>
<tr>
<td>Goldhill</td>
<td>KM</td>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>Monarch</td>
<td>KM</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Mountain</td>
<td>KM</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>Pool</td>
<td>KM</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Sails</td>
<td>KM</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Zelda</td>
<td>KM</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>
The force on the \((i,j)\)th pixel will be:
\[
F_{(i,j)} = \frac{\Gamma_{(i,j)}}{C^2} \sum_{i=1}^{M} \sum_{j=1}^{N} C \times \Gamma_{(x,y)}
\]
(29)

For any \((M \times N)\) image the quantity \(\sum_{i=1}^{M} \sum_{j=1}^{N} C \times \Gamma_{(x,y)}\) will be similar for all the pixels in the image. Therefore, considering \(\sum_{i=1}^{M} \sum_{j=1}^{N} C \times \Gamma_{(x,y)} = \text{const}_1\), we get:
\[
F_{(i,j)} = [\Gamma_{(i,j)} \times \text{const}_1]
\]
(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 \(\mu\), representing mean of the Gaussian curve. After force transformation all original intensity the values in the \(6\sigma\) range of the Gaussian curve will be represented by the final value \(\mu\).

Similarly for an image having multi Gaussian histogram with \(K\) modes, which are separated by a distance \(d > 6\sigma\), will generate \(K\) different \(\mu\)s 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.

### Table 3
Comparison of traditional BSDS500, BSDS550, and SBD datasets.

<table>
<thead>
<tr>
<th>Data set</th>
<th>Similar results with EF-Index</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DIff Index</td>
</tr>
<tr>
<td>Traditional images</td>
<td>KM 80.77% 65.28% 76.92% 84.62% 80.77%</td>
</tr>
</tbody>
</table>

### Table 4
Quantitative segmentation evaluation for BSDS300.

<table>
<thead>
<tr>
<th>PRI</th>
<th>VOI</th>
<th>GCE</th>
<th>BDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean shift</td>
<td>0.685</td>
<td>2.720</td>
<td>0.208</td>
</tr>
<tr>
<td>N-Cut</td>
<td>0.554</td>
<td>2.570</td>
<td>0.181</td>
</tr>
<tr>
<td>FH</td>
<td>0.686</td>
<td>3.405</td>
<td>0.307</td>
</tr>
<tr>
<td>FODPSO</td>
<td>0.631</td>
<td>3.746</td>
<td>0.491</td>
</tr>
<tr>
<td>EF-Index</td>
<td><strong>0.763</strong></td>
<td>3.484</td>
<td>0.348</td>
</tr>
<tr>
<td>(Fuzzy C-means)</td>
<td><strong>0.762</strong></td>
<td>3.616</td>
<td>0.359</td>
</tr>
</tbody>
</table>

### Table 5
Quantitative segmentation evaluation for BSDS550.

<table>
<thead>
<tr>
<th>PRI</th>
<th>VOI</th>
<th>GCE</th>
<th>BDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean shift</td>
<td>0.693</td>
<td>3.248</td>
<td>0.311</td>
</tr>
<tr>
<td>N-Cut</td>
<td>0.553</td>
<td><strong>2.577</strong></td>
<td>0.181</td>
</tr>
<tr>
<td>FH</td>
<td>0.692</td>
<td>3.441</td>
<td>0.310</td>
</tr>
<tr>
<td>FODPSO</td>
<td>0.638</td>
<td>3.787</td>
<td>0.500</td>
</tr>
<tr>
<td>EF-Index</td>
<td><strong>0.770</strong></td>
<td>3.518</td>
<td>0.362</td>
</tr>
<tr>
<td>(Fuzzy C-means)</td>
<td><strong>0.770</strong></td>
<td>3.639</td>
<td>0.371</td>
</tr>
</tbody>
</table>

![Fig. 5. Image segmentation results from Berkeley Segmentation Data Set (BSDS300 and BSDS500).](image-url)
3.3. Determination of appropriate number of quantization levels (EF-Index) of input image

Force Influence Image, \(\Psi\), discussed in Section 3.2, will contain number of quantization levels. The process to construct \(\Psi\) 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, \(\Phi\) of \(I\). The generation of EF-Image from input image is given in Algorithm 1.

The Force Influence Image, \(\Psi\), is depicted in Fig. 4. (d). The graph of intensities of \(\Phi\) 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 \(\Phi\) 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 \(\Phi\). 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, \(\Psi\). 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 \(\Psi\) vs. \(I\) graph of Fig. 4. (f). As a result, there is a possibility that the number of levels present in \(\Phi\) vs. \(I\) graph may decrease in the \(\Psi\) 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, \(\Psi\), will eventually turn-out to be the number of clusters, \(K\), as per Eq. (11) inherent in the original image \(I\).

![Table 6](image)

<table>
<thead>
<tr>
<th></th>
<th>PRI</th>
<th>V0I</th>
<th>GCE</th>
<th>BDE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean shift</td>
<td>0.642</td>
<td>2.318</td>
<td>0.169</td>
<td>16.430</td>
</tr>
<tr>
<td>N-Cut</td>
<td>0.746</td>
<td>2.149</td>
<td>0.197</td>
<td>11.706</td>
</tr>
<tr>
<td>FH</td>
<td>0.755</td>
<td>6.575</td>
<td>0.431</td>
<td>12.184</td>
</tr>
<tr>
<td>FODPSO</td>
<td>0.743</td>
<td>3.292</td>
<td>0.422</td>
<td>9.416</td>
</tr>
<tr>
<td>EF-Index (Fuzzy C-means)</td>
<td>0.764</td>
<td>3.885</td>
<td>0.481</td>
<td>10.031</td>
</tr>
<tr>
<td>EF-Index (K-means)</td>
<td>0.762</td>
<td>4.012</td>
<td>0.491</td>
<td>10.075</td>
</tr>
</tbody>
</table>

---

![Fig. 6](image)

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

![Fig. 7](image)

**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, $\Phi$. This EF-Image is used to generate the Force Influence Image, $\Psi$. The number of gray levels present in $\Psi$ 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 online at http://www.eecs.berkeley.edu/~yang/software/lossy_segmentation/.

Algorithm 1. EF-Image generation algorithm.

<table>
<thead>
<tr>
<th>Input: $I_{MN}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initialize: $F_{MN}$, $\Phi_{MN}$</td>
</tr>
<tr>
<td>1. for $x = 1, \ldots, M$</td>
</tr>
<tr>
<td>2. for $y = 1, \ldots, N$</td>
</tr>
<tr>
<td>3. $\Gamma_{(x,y)} \leftarrow I_{(x,y)} / 255$</td>
</tr>
<tr>
<td>4. end for</td>
</tr>
<tr>
<td>5. end for</td>
</tr>
<tr>
<td>6. for $x = 1, \ldots, M$</td>
</tr>
<tr>
<td>7. for $y = 1, \ldots, N$</td>
</tr>
<tr>
<td>8. $F_{(x,y)} \leftarrow \text{Electrostatic Force on } \Gamma_{(x,y)}$ using equation (10)</td>
</tr>
<tr>
<td>9. end for</td>
</tr>
<tr>
<td>10. end for</td>
</tr>
<tr>
<td>11. $\Phi_{MN} \leftarrow \text{Adjusted values of } F_{MN}$ applying equation (11)</td>
</tr>
</tbody>
</table>

Output: $\Phi_{MN}, \Gamma_{MN}$

Algorithm 2. Electrostatic Force Index determination algorithm.

<table>
<thead>
<tr>
<th>Input: $\Gamma_{MN}, \Phi_{MN}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initialize: $\eta \leftarrow \text{NULL set}$, $\sigma \leftarrow \text{array}$, $\theta \leftarrow \text{NULL}$, $\Omega_{MN} \leftarrow \text{2D array}$, $\Psi_{MN} \leftarrow \text{2D array}$</td>
</tr>
<tr>
<td>// $\eta$ stores the coordinate values from $\Gamma$</td>
</tr>
<tr>
<td>// $\sigma$ stores the intensity values from $\Phi$</td>
</tr>
<tr>
<td>// $\theta$ preserves the Electrostatic Force – Index (EF-Index) as the final output.</td>
</tr>
<tr>
<td>1. for $i = 0, \ldots, 255$</td>
</tr>
<tr>
<td>2. $\theta \leftarrow \text{NULL}$</td>
</tr>
<tr>
<td>3. $\eta \leftarrow \text{Coordinate of all the pixels in } \Gamma \text{ having intensity } i$</td>
</tr>
<tr>
<td>4. $\sigma \leftarrow \Phi_{(x,y)}, \forall (x,y) \in \eta$</td>
</tr>
<tr>
<td>5. $\theta \leftarrow \text{minimum of } \sigma$</td>
</tr>
<tr>
<td>6. if $\theta \neq \text{NULL}$</td>
</tr>
<tr>
<td>7. $\Psi_{(x,y)} \leftarrow \theta, \forall (x,y) \in \eta$</td>
</tr>
<tr>
<td>8. $\Omega \leftarrow \Omega \cup {\theta}$</td>
</tr>
<tr>
<td>9. end if</td>
</tr>
<tr>
<td>10. end for</td>
</tr>
<tr>
<td>11. $K \leftarrow \text{Number of gray-levels present in } \Omega$</td>
</tr>
</tbody>
</table>

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 datasets 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 from 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.

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
