

Received 23 November 2023, accepted 23 December 2023, date of publication 4 January 2024,
date of current version 16 January 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3349620

RESEARCH ARTICLE

A Three-Way Clustering Mechanism to Handle Overlapping Regions

ANWAR SHAH¹, BAHAR ALI², FAZAL WAHAB³, INAM ULLAH⁴, FAYEZ ALQAHTANI⁵,
AND AMR TOLBA⁶, (Senior Member, IEEE)

¹Department of Computer Science, FAST - National University of Computer and Emerging Sciences, Islamabad, 44000, Pakistan

²Department of Computer Science, Institute of Management Sciences, National University of Computer and Emerging Sciences, Peshawar Campus, Peshawar 25000, Pakistan

³College of Computer Science and Technology, Northeastern University, Shenyang 110819, China

⁴Department of Computer Engineering, Gachon University, Seongnam, Sujeong-gu 13120, Republic of Korea

⁵Software Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh 12372, Saudi Arabia

⁶Computer Science Department, Community College, King Saud University, Riyadh 11437, Saudi Arabia

Corresponding authors: Inam Ullah (inam.fragrance@gmail.com) and Anwar Shah (anwar.shah@nu.edu.pk)

This work was supported by King Saud University, Riyadh, Saudi Arabia, through the Researchers Supporting Project RSP2024R509.

ABSTRACT The conventional clustering methods assume a binary classification and establish a complete inclusive or exclusive type relation of an object with a cluster. In contrast, a three-way paradigm handles situations where an object may or may not belong to a cluster, i.e., uncertain. The objects belonging to the uncertainty region may lead to inclusion or exclusion after further processing and information. One of the use cases of the three-way paradigm is the overlapping region between different clusters. Effective computation of overlapping objects is crucial to the application's overall success. In this paper, we employ a three-way clustering approach inspired by image blurring and sharpening operations that consider the objects in the inside or outside regions of a cluster to be non-overlapping. The objects belonging to the partial region of more than one cluster are considered overlapping. The experiment conducted on Birds, Scenes, and 20 newsgroups datasets indicates that the proposed approach improves the F1 measure and hamming loss up to 18.6% and 4.9%, respectively. Furthermore, the system's robustness for overlapping regions is observed using typical clustering measures. The experimental results suggested that the proposed approach may improve the computation of overlapping regions effectively.

INDEX TERMS Clustering, three-way clustering, image processing, sharpening, blurring, overlapping.

I. INTRODUCTION

In many real-life applications, the clusters tend to have fuzzy boundaries and therefore have an overlapping region. A key challenge in such situations is to effectively identify objects belonging to the non-overlapping and overlapping regions [1], [2], [3]. The objects in the non-overlapping regions are clustered into a single cluster, while the objects belonging to the overlapping regions are clustered into multiple clusters, thereby depicting an overlapping area. The overlapping regions have many potential applications and may be considered an important research area. Xu proposed a method to detect the overlapped strawberries with the

help of a histogram of oriented gradients and support vector machine [4]. Similarly, the retrieval of the overlapping cloud top heights, segmentation of overlapping nuclei, and detection of overlapped gravitational wave signals and genes are some of the most rated applications of suitable detection of overlapping regions [5], [6], [7], [8].

Many researchers tried to find overlapping regions or objects using traditional machine learning algorithms or their variants, such as Overlapping K-Means (OKM) and Weighted OKM (WOKM), Multi-Cluster Overlapping K-means Extension (MCOKE), Overlapping Partitioning Cluster (OPC) and SVM-cone [9], [10], [11], [12]. Moreover, many existing approaches for overlapping clustering are based on methods derived from conventional hard clustering approaches and soft clustering strategies [13], [14], [15], [16], [17]. The

The associate editor coordinating the review of this manuscript and approving it for publication was Pasquale De Meo¹.

first category of these approaches includes the extensions of commonly used hard clustering algorithms such as k-means clustering, k-centroids clustering, and k-medoids clustering [9], [13], [18]. They incorporate rough sets or fuzzy set theories to formulate overlapping clustering algorithms such as fuzzy k-means, rough k-means, and rough-fuzzy k-means [15], [16], [19]. The essential idea is to make a single cluster assignment in case of non-overlapping or multiple cluster assignments in case of overlapping for an object by considering various association thresholds [9]. The key issue in these approaches is to find suitable thresholds [20]. There is another category for overlapping clustering, which uses graph theory. These approaches include star, estar, suffix tree-based, connected graph-based iterative scan, and overlapping clustering based on relevance (OClustR) [21], [22], [23], [24]. These approaches generally suffer from high overlapping, hinder useful information, and have a high computation that makes them unable to cope with real-life problems [24], [25].

Three-way clustering has recently been used as an effective and alternative approach for handling overlapping clusters [2], [26], [27]. The three regions of three-way clustering provide an easy interpretation for defining objects in the non-overlapping and overlapping regions. The objects in the inside regions are disjoint and mutually exclusive and therefore used to define the non-overlapping region. On the other hand, the objects in the partial regions may belong to multiple clusters and are therefore used to define the overlapping region. Yu et al. presented a density-based three-way decision approach using decision-theoretic rough sets for detecting overlapping regions [28]. Yu et al. formulate a three-way approach using interval sets to detect and shape the overlapping communities in complex networks [2]. Yu et al. also work on dynamic datasets and proposed a three-way approach that populates and increments a tree structure to detect overlapping regions [27]. In these approaches, the primary concern was formulating a three-way approach and exploring it in the different application areas. Furthermore, fixed and restricted thresholds are used to construct the three regions. Next, we have to automate the process of finding thresholds. Afridi et al. introduced variance variance-oriented three-way approach by incorporating game theoretic rough sets (GTRS) and genetic algorithms to automate the determination of suitable thresholds [26]. These methods still need refinements and improvements.

This paper introduces a three-way approach motivated by the blurring and sharpening spatial filtering operations in image processing and explores its application in identifying and detecting overlapping regions. In contrast to the previously proposed approaches, it does not need thresholds for constructing the three regions of a cluster. This approach converts the hard clusters into their respective images. Next, each cluster in its respective formatted image is realized as a typical object. These blurring and sharpening spatial filtering operations determine the core and support sets used to construct the three regions associated with a cluster. The data points in a single cluster's inside or partial region is

considered non-overlapping. On the other hand, the data points shared by multiple partial regions are identified as overlapping objects, and these regions are called overlapping regions. The contributions of this study can be precisely defined below:

- Introducing a three-way approach to identify and detect overlapping regions.
- The proposed approach does not need thresholds for detecting the three regions and the overlapping regions.

The performance of the proposed approach on Scenes and Birds datasets in comparison to some of the previous approaches, including 3WC-OR_{GTRS}, 3WC-OR_{OR}, (1, 0) model and (0.5, 0.5) model shows improvement in the typical evaluations and multi-label measures. More precisely, the proposed approach improved the F1 score and hamming loss up to by 21.6% and 7.2%, respectively.

The remaining article is structured from sections II to VI. Section II discusses the related literature. Sections III and IV introduce the proposed working methodology. Section V elaborates on the performance of the proposed method. Section VI concludes the research findings.

II. BACKGROUND

This section discusses the related background of this study. It includes 3WC and blurring and sharpening operations.

A. THREE-WAY CLUSTERING

Yao pioneered three-way clustering by extending the concept of three-way decisions [29], [30]. Let U be a universal set containing finite objects x_i , for $i = \{1, 2, 3, \dots, n\}$. A clustering algorithm results in a set of crisp partitions or clusters $C = \{c_1, c_2, \dots, c_n\}$. The main idea behind three-way clustering (3WC) is to represent a cluster c_k using core and support sets i.e., $c_k = \{Core(c_k), Support(c_k)\}$ such that $Core(c_k) \subseteq Support(c_k)$ and $Core(c_k), Support(c_k) \subset U$. The core set is the compact, condensed, and concise representation of a cluster containing objects strongly related to the cluster. In contrast, the support set is a cluster's expanded, diffused, and discursive representation, containing core objects and some additional objects weakly related to the cluster that relax the cluster representation. These two sets define the three regions associated with the cluster c_k ,

$$Inside(c_k) = Core(c_k), \quad (1)$$

$$Partial(c_k) = Support(c_k) - Core(c_k), \quad (2)$$

$$Outside(c_k) = U - Support(c_k). \quad (3)$$

The inside region includes the instances with a strong relationship to the cluster; therefore, they definitely belong to that cluster. The outside region contains objects not belonging to the cluster. The partial region contains uncertain objects that require further investigation to decide their relationship with the cluster. In literature, [20] and [31], some other notions are also used to represent a cluster, such as core $Co(c_k)$ and fringe $Fr(c_k)$. These sets are equivalent to the core set and support set in the sense of representation of a cluster

and exist in the following relation,

$$Co(c_k) = Core(c_k), \tag{4}$$

$$Fr(c_k) = Support(c_k) - Core(c_k). \tag{5}$$

The three regions, i.e., Inside, Outside, and Partial, are defined using an evaluation function and a threshold pair. The evaluation function and threshold pair determine the object's strength and placement, respectively. For a pair of thresholds (α, β) based on the value of the evaluation function $e(c_k, x_i)$, the three regions are defined as,

$$Inside(c_k) = \{x_i \in U | e(c_k, x_i) \geq \alpha\}, \tag{6}$$

$$Partial(c_k) = \{x_i \in U | \beta < e(c_k, x_i) < \alpha\}, \tag{7}$$

$$Outside(c_k) = \{x_i \in U | e(c_k, x_i) \leq \beta\}. \tag{8}$$

An object will belong to the inside region if it evaluates the evaluation function above α , mentioning its strong relationship with cluster c_k . An object will belong to the partial region if it evaluates the evaluation function between α and β , mentioning an uncertain relationship with the cluster c_k . Finally, an object will belong to outside if its evaluation function value evaluates less than β , mentioning no relationship with the cluster c_k . The determination of thresholds is the key to determining cluster boundaries for the inclusion and exclusion of an object x_i and clustering accuracy.

It is important to mention some notable works similar to the idea of 3WC with a different way to approach the solution using rough clustering, shadowed set clustering, fuzzy set clustering or interval set clustering [15], [32], [33], [34], [35].

1) ORTHOPAIRS BASED CLUSTERING APPROACHES

An orthopair 'O ∈ U' is a pair of two sets of positive (P) elements and negative (N) elements such that O = (P∪N), P∩N = ∅ and B = U - (P∪N) [36], [37]. Here, boundary (B) represents a set of objects that are not entirely positive or negative. In the three-way clustering perspective, the positive, negative, and boundary sets correspond to inside, outside, and partial regions, respectively. Three-way clustering utilizes the idea of orthopairs to represent a cluster. In literature, there are similar approaches to three-way clustering based on orthopairs, but they also have some differences. These approaches include, rough clustering [38], [39], [40], [41], [42], [43], [44], [45], [46], shadowed set [32], [47], [48], [49], [50], [51], [52] clustering, interval-set clustering and orthopartition clustering. These approaches are discussed below in comparison to 3WC.

2) 3WC AND ROUGH CLUSTERING

A rough set (R) was introduced using equivalence relations that may be defined based on a pair of lower and upper approximations as $[R, \bar{R}]$ till Palwak properties being satisfied [53], [54], [55]. Lingras and West introduced rough clustering based on the interval representation of rough sets [15]. In the case of rough clustering, an object, and a

cluster satisfy the following basic conditions of rough set theory,

- $\exists! Core_i \ni x_i$, it means that there exists exactly one lower approximation or equivalently a core to which an object can belong.
- $x_i \in Core_i \Rightarrow x_i \in Support_i$, It means that the belonging of an object to the lower approximation implies its belonging to the upper approximation of the same cluster.
- if $x_i \notin Core_i$ then $x_i \in \bigcup_{i=2}^n Partial_i$, such that $n = 2, 3, 4, \dots, n$, It means that the object not belonging to any cluster implies its presence in the upper approximation of at least two clusters.

Furthermore, the rough clustering may allow the lower approximation to be empty if objects belong to more than one upper approximation, i.e.,

- $\exists Core = \emptyset \iff x_i \in \bigcup_{i=2}^n Partial_i$ where $n = 2, 3, 4, \dots, n$.

In contrast, 3WC does not allow a core region to be empty. Moreover, there is no constraint on an object to belong with the boundary region or equivalently partial region of more than one cluster. Some of the notable works in the direction of rough set clustering are put forth in [38], [39], [40], [41], [42], [43], [44], [45], and [46].

3) 3WC AND SHADOW-SET CLUSTERING

Pedrycz introduced shadowed-set as a three-way approximation of fuzzy sets [49], [56]. The shadowed set uses a threshold pair (α, β) for a three-way approximation of a fuzzy set. The objects are approximated in these three regions, namely, core, shadow, and uncertain or excluded, as follows:

$$Core(S_i) = \{x \exists i \ x \in Core(S_i)\}, \tag{9}$$

$$Shadow(S_i) = \{x \exists i, \ x \in Shadow(S_i) \wedge \forall i \ x \notin Core(S_i)\}, \tag{10}$$

$$Uncertain(S_i) = \{x \forall i \ x \notin Shadow(S_i) \wedge x \forall i \ x \notin Core(S_i)\}. \tag{11}$$

The above Equation (9)-(11) differentiate 3WC with shadowed-set clustering. The shadowed set clustering allows the association of an object to more than one core region. Further, it allows empty core and empty shadowed region [49], [50]. In the shadowed set, it is possible that some of the objects became unclustered and declared as outliers [49].

4) 3WC AND ORTHOPARTITIONS CLUSTERING

An orthopartition \mathcal{O} is the collection of orthopairs O_i with the following basic properties [33],

- $\forall Core \in O_i \subset \mathcal{O}, Core_i \cap Core_j = \emptyset$, where $i \neq j$.
- $\bigcup_i (Core_i \cup Partial_i) = U$.
- $\forall x \in U, \exists i, j$ s.t. $x \in Partial_i \rightarrow x \in Partial_j$, where $i \neq j$.

The orthopartition framework has some differences with 3WC. The orthopairs are disjoint in orthopartitions, while

uncertain elements must belong to more than one partial region of two different orthopairs. In addition, orthopartitions allow a core region to be empty. Furthermore, it may also allow an empty partial region and even a complete orthopair [33], [36].

5) 3WC AND INTERVAL-SET CLUSTERING

Interval set clustering represents a cluster in the form of an interval set. It was basically an improvement to rough k-means clustering by proposing a two-level lower and upper approximation to unify the respective lower and upper approximation of a rough cluster under consideration [15], [57]. The following properties are obeyed by interval set clustering

- $\forall i \text{ Core}_i \neq \emptyset$
- $\bigcup_i \text{Support}_i = U$
- $\forall i \neq j, \text{Core}_i \cap \text{Core}_j = \text{Core}_i \cap \text{Partial}_j = \text{Core}_j \cap \text{Partial}_i = \emptyset$

The above equations show that the core region is not allowed to be empty, implying that the support or partial region will also not be empty. The second property suggests that each of the objects will be clustered. Finally, the core regions are mutually exclusive [15], [34], [57].

B. BLURRING AND SHARPENING OPERATIONS

Sharpening is a procedure used in the spatial domain to enhance the visual quality of an image by manipulating its edges. The enhancement is first reduced to separate the edges by their amplification and sum them back into the original image [58]. Mathematically, sharpening is defined as,

$$f_{\text{Sharpen}}(x, y) = f(x, y) + k \times f_{\text{mask}}(x, y) \quad (12)$$

where $f(x, y)$, $f_{\text{Sharpen}}(x, y)$ and $f_{\text{mask}}(x, y)$ represent the pixel values of the original image, sharpened image, and unsharp mask, respectively. Here, k is the scaling factor; its value may be 1 , ≥ 1 , or < 1 and determine the nature of sharpening, high boost filtering, and sharpening de-emphasizer, respectively. The sharpened image is obtained by deducting a blurred version of the processed image from the original image, which is given as,

$$f_{\text{mask}}(x, y) = f(x, y) - f_{lp}(x, y) \quad (13)$$

where f_{lp} is a low pass filtered image obtained using the most basic average filter and defined as,

$$f_{lp}(x, y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s, t) \quad (14)$$

where S_{xy} is an $m \times n$ processing window centered at pixel with coordinates (x, y) . The S_{xy} surrounds the neighborhood, i.e., immediate neighbors of pixel (x, y) . The blurring operation changes the intensity value of the pixel (x, y) to the average value of its neighborhood or immediate neighbors S_{xy} .

For a solid mathematical understanding of insight working mechanism of sharpening, we visually represent an input

image in Figure 1(b), and the resultant images after blurring and sharpening can be seen in Figures 1(a) and 1(c), respectively. Each cell or box approximates a pixel in the above images. The boxes are colored white, gray, or shaded, representing their intensities. The white color represents an intensity value of 1, and the gray color corresponds to 0, while the shaded boxes portray an intensity value between 0 and 1. Consider that the object of interest is represented by the white boxes positioned in the center of Figure 1(a). A low-pass filter attenuates higher frequencies and passes low frequencies, while a high-pass filter does not affect the higher frequencies. The average filter is a low-pass filter used to smoothen or blur the image. It only computes the average of the processing pixels and its immediate eight neighbors. The averaging result is replaced with the value of the processing pixel. The blurring operation expands the boundaries by contracting the object inside. The sharpening is carried out by high-pass filtering. It emphasizes finer details in the image while not affecting higher frequencies. It works exactly opposite to low-pass filters. The idea is to blur the original image using an average filter, subtract it from the original image to get the sharpening mask, and add it to the original image to sharpen it. The sharpening contracts the boundaries by expanding the object within boundaries or edges. Inspired by the idea of image blurring and image sharpening, a three-way clustering (3WC) approach is discussed in the coming section.

III. A THREE-WAY CLUSTERING APPROACH BASED ON IMAGE BLURRING AND SHARPENING (C3BS)

The C3BS has three consecutive steps:

- Converting hard clusters into their respective images
- Applying blurring sharpening operations on each cluster
- Extracting the evolved three-way soft clusters

This section elucidates each step in reasonable detail.

A. CONVERTING HARD CLUSTERS INTO THEIR RESPECTIVE IMAGES

The first step in the process is to represent the dataset in the form of a grid. The C3BS works on normalized attributes and division of the unit square (in case of two attributes) or unit hypercube (in case of more than two attributes) into an equally distributed grid [59]. The grid is realized as a pixelated image where each grid cell is equivalent to a pixel in an image. Each grid cell or pixel has the same size, containing objects or empty, representing their intensity. Further, we count the number of objects in each cell to determine their grayscale intensity.

This approach uses the Euclidean distance metric to measure the influence of each attribute. Therefore, first, we normalize the attributes A in the range $[0,1]$ to balance the effect of each attribute during distance analysis [60]. It scales the whole problem space to a unit hypercube. There are $O(n \cdot A)$ operations required to complete this step. Further, we divide the unit square (in case of two attributes) or unit hypercube (in case of more than two attributes) into p number

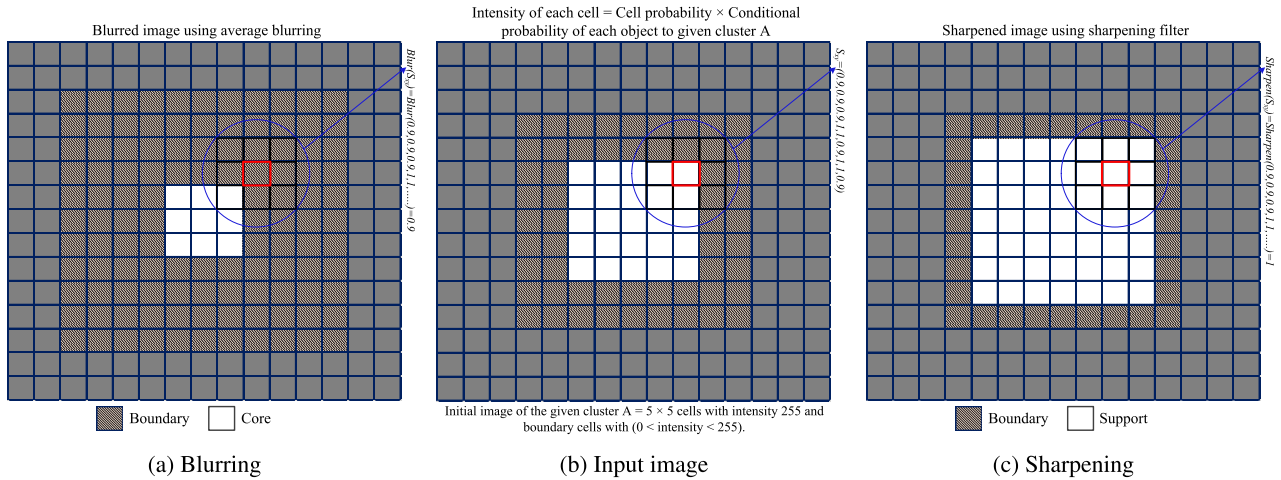


FIGURE 1. The blurring and sharpening operations.

of equidistant parts such that $p \in \mathbb{N}$. Hence, the total number of grid cells became p^A where p represents the number of grid cells or pixels in one dimension, and A denotes the number of attributes in the data space. Each grid cell can be located through A -tuple $(j_0, j_1, \dots, j_{A-1})$ where $j_k \in \{0, 1, \dots, p-1\}$. Furthermore, we can map this multidimensional grid to a single value index i.e., $I = \{0, 1, 2, \dots, p^A\}$. Algorithm 1 demonstrates the grid-based or image representation of data space.

The number of objects in each cell is a key determinant of its relevant intensity. In order to determine the intensity of each cell or pixel, the number of objects in each cell is counted, and the gridded data is clustered using a clustering algorithm or labeled data to obtain the initial partitions. The intensity level used is grayscale between 0 to 255. The number of objects $n(o)$ in each cell is divided by the maximum number $max(o)$ of objects contained in a participating cell to scale the values between 0 and 1. The intensity of each cell in a 2D grid is defined as,

$$I(x, y) = \frac{n(o)}{max(o)} \times P(o_i|C_i) \quad (15)$$

In the above equation, $I(x, y)$ shows the intensity of the corresponding cell, $P(o_i|C_i)$ determines the probability of x_i given C_i , where x_i belongs to the grid cell under processing. In the case of n -dimensions, the intensity of a pixel is given by,

$$I(x_1, x_2, \dots, x_n) = \frac{n(o)}{max(o)} \times P(o_i|C_i) \quad (16)$$

where x_1, x_2, \dots, x_n and $I(x_1, x_2, \dots, x_n)$ represents the n -attributes and intensity of the cell corresponding to these dimensions, respectively.

B. BLURRING AND SHARPENING OPERATIONS

The pixelated image obtained in the previous step is feasible for blurring and sharpening operations. In order to blur the

Algorithm 1 I-Map Algorithm

Input A universal set U , $p > 1$
Output A pixelated image I , Number of objects in each cell $n[i]$, Mapping of data into image I

```

1: function I-Map( $U$ )
2:   for each  $o_i \in U$  do
3:     Normalize  $A$  attributes between 0 and 1
4:   end for
5:   for each  $a$  in  $A$  do
6:     Divide  $A$  unit intervals  $\in [0,1]$  in  $p$  parts (or
7:     equivalently)
8:     unit hypercube  $[0, 1]^A$  into  $p^A$  cells (gridcell) to
9:     obtain  $I$ 
10:   end for
11:   for each  $i$  in  $p^A$  do
12:      $n[i] =$  Count number of objects
13:   end for
14:   for each  $o_i$  in  $U$  do
15:     Map co-ordinates of  $x_i$  to  $gridcell \in I$ 
16:     Check for duplicate entry
17:     if  $o_i \in$  more than one cell then
18:       mark it with an identifier
19:     end if
20:     Get Mapping
21:   end for
22:   return  $I, n[p]$ , Mapping
23: end function

```

image, the intensity of each pixel is updated using the average filter. Let $Neig_q(cell_i)$ be the set of q immediate neighboring pixels of a particular pixel $cell_i$. The blurring operation is formally defined as,

$$e_{blur}(cell_{c_k}, cell_i) = \frac{1}{mn} \sum_{cell_j \in Neig_q(cell_i)} I(cell_{c_k}) \quad (17)$$

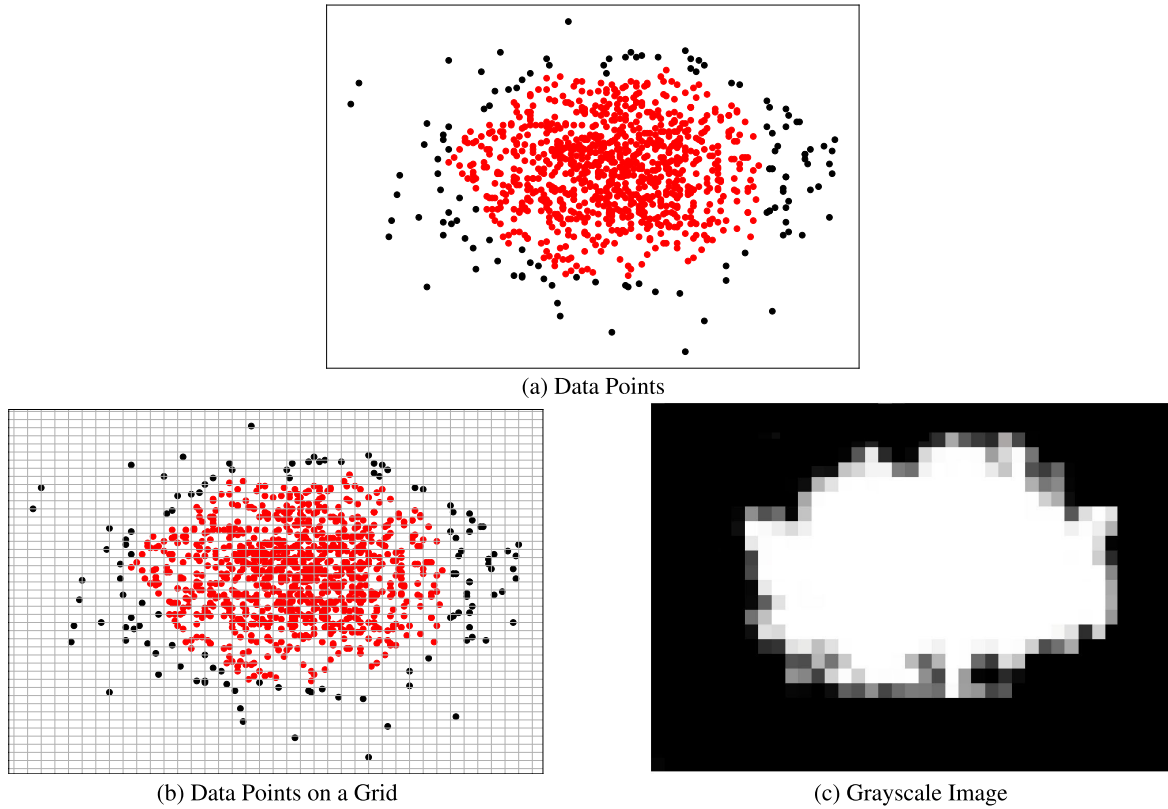


FIGURE 2. Two-dimensional image representation of two-dimensional data [61].

The above equation causes a reduction in the intensity of processing pixel $cell_{c_k}$ with respect to its immediate neighboring pixels $cell_i$ to an average value. Thus, the blurring operation results in cluster contraction by expanding boundaries.

The blurring operation will only produce a value of 1 if all the neighboring pixels have a value of 1. This property of blurring operation can be given as,

$$e_{blur}(cell_{c_k}, cell_i) = \begin{cases} 1 & \forall (cell_j) \in Neig_q(cell_i) \quad I(cell_{c_k}) = 1 \\ < 1 & otherwise \end{cases} \quad (18)$$

The above condition will only be satisfied by the pixels strongly associated with the cluster under consideration. Hence, the objects belonging to the processing pixel are strongly related to the cluster c_k .

The sharpening operation updates the intensity of a pixel $cell_{c_k}$ concerning the immediate neighboring pixels $cell_i$. A sharpened image is derived from the addition of the original image to the substitutional result of the original image with the blurred image. Mathematically, it is given as,

$$e_{sharp}(cell_{c_k}, cell_i) = I(cell_{c_k}) + [I(cell_{c_k}) - e_{blur}(cell_{c_k}, cell_i)] \quad (19)$$

The above equation enhances the intensity of the processing pixel. Thus, the sharpening operation results in an inside cluster expansion while contracting boundaries.

The sharpening operation results in a value of 1 for a pixel in each of the following cases.

- Case 1: If the blurring operation for the pixel evaluates to the value equals 1, i.e., $e_{blur}(cell_{c_k}, cell_i) = 1$.
- Case 2: If the pixel has the intensity value of 1. The sharpening operation will be evaluated to a value greater than 1, which will be truncated to 1.
- Case 3: In some scenarios where the intensity of the pixel is greater than the computed mean intensity of its q immediate neighboring pixels.

The operations of blurring and sharpening are equivalent to the core set and support set, respectively. The core set contains elements in strong relation to the cluster, while the support set includes elements with relation ranges from strong to weak. The cells are finally mapped to objects for a visual demonstration of the clustered data of the given dataset.

1) THREE-WAY CLUSTERS EXTRACTION

We noted in the above section that the blurring operation incorporates a strict condition indicating a strong bonding between an object and its associated cluster. Since the core set has similar features, therefore we can find a tie-in between

both of them and define the core set as,

$$\begin{aligned} Core(c_k) &= \{o_i \in U \mid o_i \in O_{(x_1, \dots, x_n)} \wedge \\ &e_{blur}(cell_{c_k}, cell_i) = 1\} \end{aligned} \quad (20)$$

The above equation aims to produce the core set. It emphasized that the core set contains objects or pixels where the blurring operation is evaluated to a value of one. According to Equation 17, it can only be accomplished when all the q immediate neighboring cells intensify with a value equal to one. This strict condition will be obeyed by a few members who are strongly associated with the cluster, resulting in a shrinkage of the hard cluster.

Next, we observe a sharpening operation incorporating a relatively relaxed condition indicating a weak bonding between an object and the associated cluster. Since support set has similar characteristics, therefore a tie-in can be found among both of them and define support set as,

$$\begin{aligned} Support(c_k) &= \{o_i \in U \mid o_i \in O_{(x_1, \dots, x_n)} \wedge \\ &e_{sharp}(cell_{c_k}, cell_i) = 1\} \end{aligned} \quad (21)$$

The above equation aims to produce the support set. It means that the support set contains those objects or cells where the sharpening operation results in a value of one. The definition of sharpening in Equation 19 indicates that a cell or objects will be included in the support region when at least one corresponding cell has an intensity of 1. This results in a relaxation of the given hard cluster.

There exists a relationship between the $Core(c_k)$ and $Support(c_k)$ i.e., $Core(c_k) \subseteq Support(c_k)$ based on their definitions in Equations (20)-(21). This relationship indicates that a cell in the core set must be present inside the support set. Furthermore, a cell with an intensity value of one for all its immediate neighboring cells will be included in the core set, while at least one of its immediate neighboring cells with an intensity value of one resides in the support set. Three regions of a three-way cluster based on the twain sets $Core(c_k)$ and $Support(c_k)$ are defined as,

$$\begin{aligned} Inside(c_k) &= Core(c_k) \\ &= \{o_i \in U \mid o_i \in O_{(x_1, \dots, x_n)} \wedge e_{blur}(cell_{c_k}, cell_i) = 1\} \quad (22) \\ Outside(c_k) &= U - Support(c_k) \\ &= \{o_i \in U \mid o_i \in O_{(x_1, \dots, x_n)} \wedge e_{sharp}(cell_{c_k}, cell_i) \neq 1\} \quad (23) \\ Partial(c_k) &= Support(c_k) - Core(c_k) \\ &= \{o_i \in U \mid o_i \in O_{(x_1, \dots, x_n)} \wedge e_{sharp}(cell_{c_k}, cell_i) = 1 \wedge \\ &e_{blur}(cell_{c_k}, cell_i) \neq 1\} \end{aligned} \quad (24)$$

The above Equations (20)-(21) are used to obtain three regions of a cluster. The inside region has a compact and consolidated representation containing cells or objects strongly bonded to the cluster. The partial region has a relaxed representation of the same cluster containing cells or objects weakly related to the cluster. The outside region contains those objects or cells that have no relationship with the cluster.

Algorithm 2 C3BS for Overlapping Clusters

Input A universal set $U = \{o_1, o_2, o_3, \dots, o_n\}$, An initial partition $C = \{c_1, c_2, \dots, c_K\}$.

Output The set OR and NOR depicting overlapping and non-overlapping regions.

```

1: for each  $c_k \in C$  do
2:   Obtain  $Image_{c_k}$  corresponding to cluster  $c_k$ 
3:   Apply cluster blur and cluster sharp operations on  $Image_{c_k}$ 
4:    $Core(c_k) = \{o_i \in U \mid o_i \in O_{(x_1, \dots, x_n)} \wedge$ 
       $e_{blur}(cell_{c_k}, cell_i) = 1\}$ 
5:    $Support(c_k) = \{o_i \in U \mid o_i \in O_{(x_1, x_2, \dots, x_n)} \wedge$ 
       $e_{sharp}(cell_{c_k}, cell_i) = 1\}$ 
6:    $Inside(c_k) = \{o_i \in U \mid o_i \in O_{(x_1, \dots, x_n)} \wedge$ 
       $e_{blur}(cell_{c_k}, cell_i) = 1\}$ 
7:    $Outside(c_k) = \{o_i \in U \mid o_i \in O_{(x_1, \dots, x_n)} \wedge$ 
       $e_{sharp}(cell_{c_k}, cell_i) \neq 1\}$ 
8:    $Partial(c_k) = \{o_i \in U \mid o_i \in O_{(x_1, \dots, x_n)} \wedge$ 
       $e_{sharp}(cell_{c_k}, cell_i) \wedge e_{blur}(cell_{c_k}, cell_i)$ 
9: end for
10:  $C' = \left\{ \left( Inside(c_1), Partial(c_1), Outside(c_1) \right), \dots, \right.$ 
       $\left. \left( Inside(c_K), Partial(c_K), Outside(c_K) \right) \right\}$ 
11:  $OR = U - \left[ \bigcup_{i=1}^k Inside(c_k) \cup \{o_i \in U \mid \nexists_{k,j} o_i \in Partial(c_k) \wedge$ 
       $o_i \in Partial(c_j)\} \right]$ 
12:  $NOR = U - OR$ 
13: for each  $o_i \in OR$  do
14:   Decide  $o_i$  in all multiples clusters for which  $o_i \in$ 
       $Inside(c_k)$ 
15: end for
16: for each  $o_i \in NOR$  do
17:   Decide  $o_i$  in a cluster for which  $o_i \in Inside(c_k) \vee$ 
       $o_i \in Partial(c_k)$ 
18: end for

```

IV. A C3BS BASED ALGORITHM FOR OVERLAPPING REGIONS

This section presents a use case for the C3BS, i.e., overlapping regions. The algorithm 2 is the modified version of C3BS to handle the overlapping clusters. This algorithm has two inputs: a universal set U and its initial partition. The algorithm's output is the sets of OR and NOR representing the sets containing the overlapping and non-overlapping objects. The algorithm begins by executing an algorithm corresponding to C3BS, which creates three-way clusters corresponding to each initial cluster given in the set C . In line 2, the algorithm defines the set of overlapping objects given by the set OR based on the results of three-way clustering. In particular, all the objects that are not inside any cluster and do not belong to the partial region of a single cluster are considered to be overlapping objects. The second set defined in line 2 reflects only those objects in a single cluster's partial region. Line 3 defines the non-overlapping

objects given by the set NOR . In lines 4 to 6, the overlapping objects are assigned to multiple clusters. More specifically, an object in the overlapping region is assigned to all those clusters for which it belongs to its respective partial regions. Finally, in lines 7 to 9, the non-overlapping objects are assigned to a single cluster. Each object, in this case, is assigned to a single cluster for which it belongs to its respective inside or partial region.

V. PERFORMANCE OF C3BS FOR OVERLAPPING REGIONS

In this section, we analyze the performance of C3BS for overlapping clusters. Multi-labeled datasets may be used to depict overlapping clusters by realizing a one-to-one correspondence between clusters and classes. In particular, all the multi-labeled instances may be treated as overlapping objects. We use multi-labeled datasets of scenes and birds. The scenes dataset consists of 2407 instances, 294 features, and six clusters. A total of 177 instances have multi-labels and are, therefore, in the overlapping region, while the remaining 2230 instances lie in the non-overlapping region. The bird's dataset consists of 645 instances, 277 features, and 19 clusters. There are 162 instances with multi-labels which suggests these instances overlap, while the remaining 189 instances lie in the non-overlapping region. We considered two types of benchmarks for evaluating performance. The first type of benchmark provides a general insight into the quality of the clusters. The measures of DB index, silhouette, and classification accuracy are used for this purpose. The second type of benchmark provides insight into the performance of correctly predicting the overlapping objects. The hamming loss and F1 measures defined for multi-label problems are used for this purpose [62]. Let TL_i and PL_i represent the true label and predicted labels for an object o_i , respectively. The F1 measure for a dataset with N objects is given as,

$$F1 = \frac{1}{N} \sum_{i=1}^N \frac{2 \times |TL_i \cap PL_i|}{|TL_i| + |PL_i|} \quad (25)$$

Hamming loss is the fraction of wrongly labeled objects used in multi-class classification scenarios and given as,

$$Hamming\ loss = \frac{1}{KN} |TL_i \cap PL_i^c| + |TL_i^c \cap PL_i| \quad (26)$$

In the above equation, K represents the number of classes or clusters in the dataset while PL_i^c and TL_i^c are the complements sets of PL_i and TL_i , respectively. Hamming loss is used to detect both prediction errors and classification errors.

Table 1 reports the experimental results for the considered datasets. It may be noted that overlapping clustering studies generally considered datasets that are artificially created and are not available publicly. Moreover, each study uses different datasets to verify their respective approaches. For these reasons, we provide comparisons with our previous work on overlapping clustering in the first place [26]. The comparative algorithms are based on threshold settings that are fixed or automatically determined using game-theoretic rough sets. The (0.5, 0.5) and (1, 0) models correspond to three-way clusters obtained with thresholds of $(\alpha, \beta) = (0.5, 0.5)$ and $(1, 0)$, respectively, using equations (6)-(8). The 3WC-OR_{GA} results are bounded by strict conditions where (0.5, 0.5) is more generally experiencing poor Hamming loss and Silhouette score for the Scenes dataset. The 3WC-OR_{GA} has a compromising DB index for the Scenes and Silhouette for Birds datasets. It determines its strength of better interpretation of the consistency of the objects within the cluster. On the other hand, 3WC-OR_{GTRS} uses game-theoretic rough sets to automatically determine thresholds. The multi-label measures are very compromising for the 3WC-OR_{GTRS} model. The (1, 0) model has the same F1 score as 3WC-OR_{GTRS}; however, it has a high Hamming loss comparatively. Moreover, the SVM-cone and MCOKE methods evaluate good accuracies. The typical measurement metrics for OKM and OPC are very close. The WOKM performs poorly due to its weighted nature. The proposed model significantly improves the results for both datasets. More specifically, it tackles overlapping clustering in a well-organized way and reduces miss-hits. The typical evaluation and multi-label measure are improved for both Scenes and Birds datasets. The F1 measure indicates the overall performance, which is improved by 18.6% for the Scene dataset and 13% for the Birds dataset. Similarly, miss-hits are determined by Hamming loss, and it has been improved by 2.1% for the Scene dataset, where 2.4% for the Birds dataset.

A. ANALYZING OVERLAPPING REGIONS

In this section, we take a closer look at the performance of the C algorithm in the area of overlapping regions. More precisely, we observe the robustness of C in computing the overlapping regions. For such analysis, we use three metrics i.e., $Precision_{OL}$, $Recall_{OL}$ and $Accuracy_{OL}$ [26]. Mathematically, these metrics can be defined as (27)–(29), shown at the bottom of the page.

$$Precision_{OL} = \frac{\text{Correctly identified objects from overlapping region}}{\text{Total Objects}} \quad (27)$$

$$Recall_{OL} = \frac{\text{Overlapping objects correctly assigned to overlapping region}}{\text{Total Objects}} \quad (28)$$

$$Accuracy_{OL} = \frac{\text{Correctly identified overlapping and non overlapping objects}}{\text{Total Objects}} \quad (29)$$

TABLE 1. Typical and multi-label evaluation measures.

Dataset	Approach	Typical evaluation measures			Multi-label measures	
		Accuracy	DB index	Silhouette	F1-Measure	Hamm.loss
Scenes	C3BS	0.919 ± 0.022	1.650 ± 0.021	0.661 ± 0.024	0.910 ± 0.073	0.097 ± 0.012
	3WC-OR _{GTRS}	0.883 ± 0.019	2.105 ± 0.005	0.179 ± 0.018	0.701 ± 0.014	0.118 ± 0.019
	3WC-OR _{GA}	0.894 ± 0.022	2.493 ± 0.008	0.239 ± 0.029	0.724 ± 0.022	0.098 ± 0.022
	(1,0) model	0.673 ± 0.021	2.213 ± 0.004	0.201 ± 0.051	0.702 ± 0.014	0.292 ± 0.032
	(0.5,0.5) model	0.894 ± 0.026	2.910 ± 0.003	0.115 ± 0.039	0.694 ± 0.001	0.169 ± 0.015
	SVM-cone	0.775 ± 0.015	2.910 ± 0.001	0.115 ± 0.008	0.591 ± 0.003	0.170 ± 0.010
	MCOKE	0.684 ± 0.016	2.970 ± 0.005	0.107 ± 0.029	0.584 ± 0.003	0.174 ± 0.052
	OKM	0.652 ± 0.014	2.990 ± 0.007	0.097 ± 0.019	0.544 ± 0.006	0.199 ± 0.012
	OPC	0.601 ± 0.032	3.001 ± 0.013	0.091 ± 0.022	0.494 ± 0.002	0.213 ± 0.024
	WOKM	0.537 ± 0.006	3.140 ± 0.013	0.081 ± 0.017	0.514 ± 0.001	0.247 ± 0.035
Birds	C3BS	0.954 ± 0.003	0.366 ± 0.207	0.965 ± 0.013	0.912 ± 0.041	0.052 ± 0.020
	3WC-OR _{GTRS}	0.940 ± 0.023	0.308 ± 0.006	0.901 ± 0.022	0.740 ± 0.014	0.076 ± 0.023
	3WC-OR _{GA}	0.943 ± 0.026	0.311 ± 0.010	0.755 ± 0.035	0.782 ± 0.023	0.077 ± 0.026
	(1,0) model	0.890 ± 0.034	0.174 ± 0.002	0.841 ± 0.015	0.740 ± 0.025	0.103 ± 0.023
	(0.5,0.5) model	0.901 ± 0.029	0.535 ± 0.002	0.759 ± 0.025	0.714 ± 0.019	0.094 ± 0.025
	SVM-cone	0.824 ± 0.021	0.582 ± 0.013	0.765 ± 0.009	0.694 ± 0.007	0.104 ± 0.005
	MCOKE	0.804 ± 0.020	0.325 ± 0.014	0.688 ± 0.009	0.688 ± 0.017	0.114 ± 0.001
	OKM	0.769 ± 0.012	0.301 ± 0.001	0.686 ± 0.018	0.674 ± 0.021	0.129 ± 0.011
	OPC	0.768 ± 0.013	0.211 ± 0.004	0.596 ± 0.024	0.664 ± 0.013	0.130 ± 0.017
	WOKM	0.664 ± 0.006	0.281 ± 0.003	0.599 ± 0.017	0.610 ± 0.011	0.151 ± 0.055

TABLE 2. Quality attributes of overlapping regions.

Dataset	Approach	Precision _{OL}	Recall _{OL}	Accuracy _{OL}	Dataset	Approach	Precision _{OL}	Recall _{OL}	Accuracy _{OL}
Scene	C3BS	0.967	0.989	0.987	Birds	C3BS	0.867	0.963	0.936
	3WC-OR _{GTRS}	0.401	0.576	0.905		3WC-OR _{GTRS}	0.538	0.555	0.730
	3WC-OR _{GA}	0.187	0.033	0.918		3WC-OR _{GA}	0.327	0.095	0.677
	(1,0) model	0.082	1.000	0.182		(1,0) model	0.297	0.936	0.372
	(0.5,0.5) model	0.000	0.000	0.926		(0.5,0.5) model	0.000	0.000	0.707
	SVM-cone	0.501	0.562	0.552		SVM-cone	0.441	0.476	0.451
	MCOKE	0.115	0.128	0.120		MCOKE	0.276	0.284	0.281
	OKM	0.072	0.085	0.080		OKM	0.221	0.227	0.224
	OPC	0.015	0.028	0.019		OPC	0.200	0.207	0.205
	WOKM	0.010	0.019	0.013		WOKM	0.017	0.021	0.019

TABLE 3. Overlapping measurement for 25% classes with overlapping objects.

Method	Precision	Recall	Accuracy	F1 Score	Hamming loss
C3BS	0.899 ± 0.012	0.875 ± 0.004	0.885 ± 0.021	0.886 ± 0.014	0.108 ± 0.001
3WC-OR _{GTRS}	0.811 ± 0.011	0.811 ± 0.007	0.809 ± 0.022	0.811 ± 0.012	0.111 ± 0.002
3WC-OR _{GA}	0.735 ± 0.012	0.764 ± 0.024	0.746 ± 0.022	0.750 ± 0.017	0.117 ± 0.014
(1, 0) model	0.684 ± 0.017	0.714 ± 0.041	0.694 ± 0.031	0.698 ± 0.001	0.157 ± 0.002
(0.5, 0.5) model	0.632 ± 0.002	0.662 ± 0.021	0.641 ± 0.011	0.647 ± 0.002	0.169 ± 0.003
SVM-cone	0.653 ± 0.004	0.688 ± 0.001	0.665 ± 0.004	0.670 ± 0.024	0.177 ± 0.002
MCKOKE	0.581 ± 0.003	0.555 ± 0.003	0.561 ± 0.007	0.568 ± 0.035	0.241 ± 0.007
OKM	0.558 ± 0.004	0.539 ± 0.005	0.541 ± 0.019	0.548 ± 0.014	0.242 ± 0.002
OPC	0.548 ± 0.006	0.505 ± 0.018	0.518 ± 0.018	0.526 ± 0.011	0.245 ± 0.002
WOKM	0.518 ± 0.007	0.471 ± 0.019	0.485 ± 0.021	0.494 ± 0.001	0.301 ± 0.003

Table 2 reports the quality-based performance of C3BS in correctly identifying the overlapping regions by considering typical classification measures. The measure of precision in the table shows how many predicted overlapping objects are truly overlapping objects. On the other hand, recall shows what proportion of the actual overlapping objects are correctly classified as overlapping objects. Finally, the measure of accuracy depicts the percentage of correct assignments of objects to the overlapping and non-overlapping regions. The next set of experiments is conducted on text data from 20 newsgroups. This data set contained news from 20 groups

broadly categorized as political, religious, sports, sales, electronics, and graphics-related news. The experiments are performed in two phases. The first phase of experiments is reported in Table 3 - Table 6, and the results for the second phase are tabulated as Table 7 and Table 8. The first phase is to evaluate the performance of the proposed approach in case of an increase in the overlapping classes. Table 3 reports the results when only 25% of the classes, i.e., five, participated in the experiments.

Table 4 shows the results when the classes became 50%, i.e., ten participated. The performance of participant methods

TABLE 4. Overlapping measurement for 50% classes with overlapping objects.

Method	Precision	Recall	Accuracy	F1 Score	Hamming loss
C3BS	0.846 ± 0.001	0.925 ± 0.011	0.882 ± 0.025	0.883 ± 0.014	0.102 ± 0.002
3WC-OR _{GTRS}	0.786 ± 0.005	0.888 ± 0.011	0.832 ± 0.027	0.834 ± 0.015	0.108 ± 0.004
3WC-OR _{GA}	0.719 ± 0.014	0.881 ± 0.014	0.788 ± 0.027	0.792 ± 0.016	0.114 ± 0.005
(1, 0) model	0.653 ± 0.012	0.853 ± 0.017	0.734 ± 0.022	0.739 ± 0.017	0.152 ± 0.003
(0.5, 0.5) model	0.617 ± 0.023	0.878 ± 0.018	0.718 ± 0.014	0.725 ± 0.019	0.159 ± 0.002
SVM-cone	0.638 ± 0.022	0.821 ± 0.017	0.713 ± 0.014	0.718 ± 0.018	0.168 ± 0.002
MCKOKE	0.585 ± 0.014	0.852 ± 0.015	0.688 ± 0.025	0.694 ± 0.021	0.233 ± 0.005
OKM	0.595 ± 0.022	0.808 ± 0.024	0.679 ± 0.031	0.686 ± 0.003	0.221 ± 0.007
OPC	0.568 ± 0.041	0.816 ± 0.015	0.663 ± 0.001	0.670 ± 0.006	0.239 ± 0.001
WOKM	0.533 ± 0.023	0.743 ± 0.017	0.613 ± 0.017	0.621 ± 0.001	0.299 ± 0.010

TABLE 5. Overlapping measurement for 75% classes with overlapping objects.

Method	Precision	Recall	Accuracy	F1 Score	Hamming loss
C3BS	0.838 ± 0.017	0.933 ± 0.014	0.882 ± 0.041	0.883 ± 0.014	0.098 ± 0.011
3WC-OR _{GTRS}	0.780 ± 0.014	0.880 ± 0.017	0.824 ± 0.041	0.827 ± 0.021	0.102 ± 0.004
3WC-OR _{GA}	0.713 ± 0.015	0.891 ± 0.011	0.788 ± 0.021	0.792 ± 0.017	0.107 ± 0.002
(1, 0) model	0.646 ± 0.001	0.853 ± 0.001	0.730 ± 0.032	0.735 ± 0.022	0.149 ± 0.014
(0.5, 0.5) model	0.610 ± 0.022	0.878 ± 0.002	0.714 ± 0.034	0.721 ± 0.032	0.157 ± 0.017
SVM-cone	0.633 ± 0.021	0.831 ± 0.003	0.713 ± 0.025	0.718 ± 0.021	0.155 ± 0.021
MCKOKE	0.580 ± 0.014	0.840 ± 0.002	0.679 ± 0.027	0.686 ± 0.021	0.229 ± 0.022
OKM	0.590 ± 0.010	0.819 ± 0.004	0.679 ± 0.011	0.686 ± 0.025	0.228 ± 0.011
OPC	0.563 ± 0.017	0.828 ± 0.005	0.663 ± 0.012	0.670 ± 0.024	0.239 ± 0.007
WOKM	0.528 ± 0.005	0.753 ± 0.011	0.613 ± 0.017	0.621 ± 0.023	0.299 ± 0.005

TABLE 6. Overlapping measurement for 100% classes with overlapping objects.

Method	Precision	Recall	Accuracy	F1 Score	Hamming loss
C3BS	0.831 ± 0.003	0.942 ± 0.004	0.882 ± 0.007	0.883 ± 0.021	0.095 ± 0.005
3WC-OR _{GTRS}	0.774 ± 0.005	0.905 ± 0.005	0.832 ± 0.006	0.834 ± 0.022	0.099 ± 0.006
3WC-OR _{GA}	0.706 ± 0.007	0.911 ± 0.008	0.792 ± 0.007	0.796 ± 0.014	0.104 ± 0.002
(1, 0) model	0.640 ± 0.001	0.853 ± 0.014	0.725 ± 0.009	0.731 ± 0.017	0.134 ± 0.016
(0.5, 0.5) model	0.597 ± 0.001	0.878 ± 0.006	0.705 ± 0.012	0.711 ± 0.018	0.149 ± 0.022
SVM-cone	0.621 ± 0.004	0.862 ± 0.009	0.717 ± 0.021	0.722 ± 0.015	0.149 ± 0.014
MCKOKE	0.568 ± 0.028	0.865 ± 0.004	0.679 ± 0.060	0.686 ± 0.011	0.224 ± 0.019
OKM	0.578 ± 0.031	0.842 ± 0.018	0.679 ± 0.033	0.686 ± 0.001	0.225 ± 0.001
OPC	0.552 ± 0.017	0.865 ± 0.041	0.667 ± 0.025	0.674 ± 0.017	0.231 ± 0.001
WOKM	0.514 ± 0.009	0.785 ± 0.005	0.613 ± 0.002	0.621 ± 0.044	0.295 ± 0.002

TABLE 7. First iteration (low-level overlapping probability).

Method	Precision	Recall	Accuracy	F1 Score	Hamming loss
C3BS	0.886 ± 0.005	0.857 ± 0.004	0.882 ± 0.014	0.871 ± 0.014	0.103 ± 0.005
3WC-OR _{GTRS}	0.780 ± 0.003	0.771 ± 0.007	0.799 ± 0.012	0.775 ± 0.012	0.113 ± 0.008
3WC-OR _{GA}	0.714 ± 0.004	0.764 ± 0.024	0.753 ± 0.017	0.738 ± 0.017	0.138 ± 0.006
(1, 0) model	0.662 ± 0.009	0.696 ± 0.017	0.692 ± 0.014	0.679 ± 0.012	0.147 ± 0.008
(0.5, 0.5) model	0.600 ± 0.006	0.681 ± 0.001	0.656 ± 0.012	0.638 ± 0.017	0.161 ± 0.007
SVM-cone	0.629 ± 0.004	0.653 ± 0.002	0.663 ± 0.017	0.641 ± 0.021	0.161 ± 0.006
MCKOKE	0.555 ± 0.005	0.512 ± 0.018	0.554 ± 0.007	0.533 ± 0.025	0.251 ± 0.005
OKM	0.542 ± 0.004	0.520 ± 0.005	0.554 ± 0.019	0.531 ± 0.027	0.252 ± 0.006
OPC	0.530 ± 0.002	0.530 ± 0.018	0.554 ± 0.018	0.530 ± 0.027	0.257 ± 0.010
WOKM	0.500 ± 0.001	0.500 ± 0.019	0.535 ± 0.021	0.500 ± 0.022	0.275 ± 0.011

improved with an increase in the number of classes. The precision for C3BS drops by 5.3%, and recall has an increase of 5%, resulting in a drop of F1 score by 3%. Overall the positive difference of C3BS drops in the F1 score from 7.5% to 4.9%, but still high in comparison.

Further, the classes increased from 50% to 75%, i.e., 15 participants. The performance became improved for all the methods. Overall, the precision value for C3BS drops by 0.8% while recall increases by the same percentage, resulting in a maintained 88.3% F1 score. In comparison,

the C3BS is still dominant, improving hamming loss from 0.102 to 0.098. The final addition of 5 classes made the whole dataset completely included. These experiments are tabulated as Table 6. The C3BS has a drop in precision by 0.8% with an increase in the recall by 0.9% and hamming loss from 0.098 to 0.095. These experiments show that the performance of most of the overlapping methods improves with an increase in overlapping classes. The C3BS performed well in this set of experiments by improving Hamming loss, i.e., decreasing miss hits.

TABLE 8. Second iteration (high level overlapping probability).

Method	Precision	Recall	Accuracy	F1 Score	Hamming loss
C3BS	0.783 ± 0.001	0.890 ± 0.014	0.86 ± 0.025	0.833 ± 0.001	0.111 ± 0.103
3WC-OR _{GTRS}	0.704 ± 0.005	0.837 ± 0.012	0.802 ± 0.027	0.765 ± 0.005	0.135 ± 0.113
3WC-OR _{GA}	0.619 ± 0.014	0.825 ± 0.023	0.752 ± 0.027	0.707 ± 0.005	0.142 ± 0.138
(1, 0) model	0.552 ± 0.012	0.792 ± 0.025	0.694 ± 0.022	0.651 ± 0.014	0.149 ± 0.147
(0.5, 0.5) model	0.492 ± 0.023	0.813 ± 0.027	0.670 ± 0.014	0.614 ± 0.014	0.153 ± 0.161
SVM-cone	0.434 ± 0.022	0.666 ± 0.027	0.638 ± 0.014	0.526 ± 0.025	0.222 ± 0.161
MCKOKE	0.405 ± 0.014	0.736 ± 0.025	0.617 ± 0.025	0.523 ± 0.027	0.235 ± 0.251
OKM	0.375 ± 0.022	0.631 ± 0.031	0.595 ± 0.031	0.470 ± 0.027	0.314 ± 0.252
OPC	0.333 ± 0.041	0.628 ± 0.001	0.573 ± 0.001	0.435 ± 0.025	0.353 ± 0.257
WOKM	0.294 ± 0.023	0.512 ± 0.017	0.516 ± 0.017	0.373 ± 0.021	0.381 ± 0.275

The next phase is to evaluate the performance of the proposed approach in case of an increase in the overlapping objects or overlapping regions. The experiments are performed initially on the 20 newsgroups dataset, and the results have been tabulated as the first iteration of the process in Table 1. In the next iteration, we include words that can increase overlapping objects or regions. For instance, words related to hockey are included in the electronics news, making it more favorable for both hockey and electronics classes. Similarly, a list of words has been added to increase the chances of overlapping and overlapping regions. The results are reported in Table 8. The recall for C3BS increased by 3.3% while the precision, F1 score, and Hamming loss dropped by 10.3%, 3.8%, and 0.8%. Similarly, the participants have a drop in F1 score and hamming loss. In comparison, with rival methods, the Hamming loss improved from 1% to 2.4%. These experiments show that the performance drop may be due to an increased overlapping region. The C3BS performed well in this set of experiments by maintaining its supremacy over the state-of-the-art methods.

VI. TIME COMPLEXITY

There are two algorithms, 1 and 2. Algorithm 1 can be summarized in four parts, i.e., normalization (Lines 2-4), grid generation (Lines 5-8), counting objects in cells (Lines 9-11), and mapping coordinates to cells (Lines 12-19).

Normalization involves iterating through each object in the universal set U and normalizing its attributes. Let there be N objects in U , and each object has A attributes, then the time complexity for normalization is $O(N \times A)$. Grid generation is possible by dividing each attribute into p parts. If there are A attributes and each is divided into p parts, the total number of grid cells will be p^A . So, the time complexity for grid generation is $O(p^A)$. Next, to count objects in each grid cell, the algorithm counts the number of objects in that cell. Since there are p^A cells and N objects, the time complexity for counting is $O(N \times p^A)$. Finally, mapping coordinates to cells takes place. The algorithm maps each object's coordinates to the corresponding grid cell in the pixelated image. It involves determining the cell for each of the A attributes, and therefore, the time complexity for mapping is $O(N \times A)$. The overall time complexity T_{I-Map} is the sum of these steps:

$$T_{I-Map} = O(N \times A) + O(p^A) + O(N \times p^A) + O(N \times A)$$

Since we are concerned with asymptotic behavior, it can be simplified to:

$$T_{I-Map} = O(N \times p^A) + O(p^A) \quad (30)$$

We consider the dominant term $O(N \times p^A)$ to express this in big O notation. Therefore, the time complexity of the I-Map algorithm is $O(N \times p^A)$. Regarding omega notation, since $O(N \times p^A)$ is also a lower bound, we can say the time complexity is also $\Omega(N \times p^A)$.

Algorithm 2 can also be summarized in four parts, i.e., the first part is the iteration over clusters (Lines 1-9), cluster operations (Lines 2-7), and Core, Support, Inside, Outside, Partial regions creation (Lines 4-8). The algorithm iterates over each cluster in the initial partition phase; for each cluster, the algorithm computes sets $Core(c_k)$, $Support(c_k)$, $Inside(c_k)$, $Outside(c_k)$, and $Partial(c_k)$. The computation involves iterating over objects in U and checking conditions based on the blur and sharp operations. If there are N objects in U and the image size is M , the time complexity for these sets is $O(N \times M)$ for each cluster, resulting in $O(K \times N \times M)$ overall.

The next part is the Construction of C' (Line 10). The algorithm constructs a new set C' by combining each cluster's Inside, Partial, and Outside sets. If there are K clusters, the time complexity for this step is $O(K \times N \times M)$. In the next step, the Construction of OR and NOR occurred in (Lines 11-12). The algorithm constructs overlapping (OR) and non-overlapping (NOR) sets based on the constructed C' . The time complexity for this step is $O(K \times N)$. Finally, the iteration over OR and NOR occurs in (Lines 13-18). The algorithm in this step iterates over objects in OR and NOR , making decisions based on cluster membership. Let there are L objects in OR and M objects in NOR ; the time complexity for this step is $O(L + M)$.

The overall time complexity is the sum of these steps:

$$T_{C3BS} = O(K \times N \times M) + O(K \times N \times M) + O(K \times N) + O(L + M)$$

Since we are concerned with asymptotic behavior, we can simplify this to:

$$T_{C3BS} = 2 \times O(K \times N \times M) + O(K \times N) + O(L + M) \quad (31)$$

We consider the dominant term $O(K \times N \times M)$ to express this in big O notation. Therefore, the time complexity of the

C3BS algorithm is $O(K \times N \times M)$. Regarding omega notation, since $O(K \times N \times M)$ is also a lower bound, we can say the time complexity is also $\Omega(K \times N \times M)$.

Now, to determine the total time complexity of the process involving Algorithm 1 (I-Map) and Algorithm 2 (C3BS), we need to consider the time complexity of each algorithm separately and then combine them. Therefore, the total time complexity T_{Total} of the whole process, considering the worst-case scenario and taking the dominant term, is given by:

$$\begin{aligned} T_{Total} &= T_{I-Map} + T_{C3BS} \\ T_{Total} &= O(N \times p^A) + O(K \times N \times M) \\ T_{Total} &= O(K \times N \times M) \end{aligned} \quad (32)$$

Since we are considering the worst-case scenario, we take the dominant term $O(K \times N \times M)$. Therefore, the total time complexity of the combined process is $O(K \times N \times M)$. Regarding omega notation, the time complexity is also $\Omega(K \times N \times M)$.

VII. CONCLUSION

In real life, in many applications, such as wireless sensor networks and topic modeling, an instance can belong to more than one cluster. Three-way clustering is capable of organizing a cluster into three regions which can be used to identify the objects belonging to multiple clusters. We explored the role of blurring and sharpening in identifying the overlapping objects among the clusters and proposed a blur-sharp-based algorithm. The algorithm first obtains the three regions associated with each cluster. It next finds the objects that belong to the partial regions of more than one cluster. Compared with previous state-of-the-art methods, the experiments were conducted on two widely used multi-label datasets, Scenes and Birds, showing that the proposed approach effectively solves overlapping clustering. Furthermore, the performance of C3BS is evaluated on a textual dataset, namely, 20 newsgroups. More precisely, C3BS improves the results on Scenes, Birds, and 20 newsgroups datasets by up to 18.6%, 13%, 4.9%, respectively, for F1 score and 2.1%, 2.4% and 0.4%, respectively for Hamming loss. In our future work, we will refine the 3WC paradigm to handle more complex scenarios. Furthermore, methods will be investigated to optimize the computational efficiency of 3WC, especially for larger datasets.

REFERENCES

- [1] W. Liu, T. Suzumura, H. Ji, and G. Hu, "Finding overlapping communities in multilayer networks," *PLoS ONE*, vol. 13, no. 4, Apr. 2018, Art. no. e0188747.
- [2] H. Yu, P. Jiao, Y. Yao, and G. Wang, "Detecting and refining overlapping regions in complex networks with three-way decisions," *Inf. Sci.*, vol. 373, pp. 21–41, Dec. 2016.
- [3] H. Yu and Y. Wang, "Three-way decisions method for overlapping clustering," in *Proc. Int. Conf. Rough Sets Current Trends Comput.* Berlin, Germany: Springer, 2012, pp. 277–286.
- [4] Y. Xu, K. Imou, Y. Kaizu, and K. Saga, "Two-stage approach for detecting slightly overlapping strawberries using HOG descriptor," *Biosyst. Eng.*, vol. 115, no. 2, pp. 144–153, Jun. 2013.
- [5] S. Teng, C. Liu, Z. Tan, J. Li, N. Xu, X. Hu, P. Zhang, W. Yan, and B.-J. Sohn, "A multispectral method for retrieving overlapping cloud top heights from passive radiometers," *Remote Sens. Environ.*, vol. 286, Mar. 2023, Art. no. 113425.
- [6] I. Kiran, B. Raza, A. Ijaz, and M. A. Khan, "DenseRes-UNet: Segmentation of overlapped/clustered nuclei from multi organ histopathology images," *Comput. Biol. Med.*, vol. 143, Apr. 2022, Art. no. 105267.
- [7] B. W. Wright, M. P. Molloy, and P. R. Jaschke, "Overlapping genes in natural and engineered genomes," *Nature Rev. Genet.*, vol. 23, no. 3, pp. 154–168, Mar. 2022.
- [8] E. Pizzati, S. Sachdev, A. Gupta, and B. S. Sathyaprakash, "Toward inference of overlapping gravitational-wave signals," *Phys. Rev. D, Part. Fields*, vol. 105, no. 10, May 2022, Art. no. 104016.
- [9] G. Cleuziou, "An extended version of the k -means method for overlapping clustering," in *Proc. 19th Int. Conf. Pattern Recognit.*, Dec. 2008, pp. 1–4.
- [10] J. Z. Huang, M. K. Ng, H. Rong, and Z. Li, "Automated variable weighting in k -means type clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 5, pp. 657–668, May 2005.
- [11] Y.-L. Chen and H.-L. Hu, "An overlapping cluster algorithm to provide non-exhaustive clustering," *Eur. J. Oper. Res.*, vol. 173, no. 3, pp. 762–780, Sep. 2006.
- [12] S. Baadel, F. Thabtah, and J. Lu, "MCOKE: Multi-cluster overlapping k -means extension algorithm," *Int. J. Comput., Control, Quantum Inf. Eng.*, vol. 9, no. 2, pp. 427–430, 2015.
- [13] A. Chaturvedi, J. D. Carroll, P. E. Green, and J. A. Rotondo, "A feature-based approach to market segmentation via overlapping k -centroids clustering," *J. Marketing Res.*, vol. 34, no. 3, p. 370, Aug. 1997.
- [14] L. Gabrielli, S. Giuffrida, and M. R. Trovato, "Gaps and overlaps of urban housing sub-market: Hard clustering and fuzzy clustering approaches," in *Appraisal: From Theory to Practice*. Cham, Switzerland: Springer, 2017, pp. 203–219.
- [15] P. Lingras and C. West, "Interval set clustering of web users with rough K -means," *J. Intell. Inf. Syst.*, vol. 23, no. 1, pp. 5–16, Jul. 2004.
- [16] S. Mitra, H. Banka, and W. Pedrycz, "Rough-fuzzy collaborative clustering," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 4, pp. 795–805, Sep. 2006.
- [17] P. G. Sun, L. Gao, and S. Shan Han, "Identification of overlapping and non-overlapping community structure by fuzzy clustering in complex networks," *Inf. Sci.*, vol. 181, no. 6, pp. 1060–1071, Mar. 2011.
- [18] R. Krishnapuram, A. Joshi, and L. Yi, "A fuzzy relative of the k -medoids algorithm with application to web document and snippet clustering," in *Proc. Int. Conf. Fuzzy Syst.*, vol. 3, Aug. 1999, pp. 1281–1286.
- [19] J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*. York, NY, USA: Springer, 2013.
- [20] P. Wang and Y. Yao, "CE3: A three-way clustering method based on mathematical morphology," *Knowl.-Based Syst.*, vol. 155, pp. 54–65, Sep. 2018.
- [21] J. A. Aslam, E. Pelekhev, and D. Rus, "The star clustering algorithm for static and dynamic information organization," *J. Graph Algorithms Appl.*, vol. 8, no. 1, pp. 95–129, 2004.
- [22] M. Goldberg, S. Kelley, M. Magdon-Ismael, K. Mertsalov, and A. Wallace, "Finding overlapping communities in social networks," in *Proc. IEEE 2nd Int. Conf. Social Comput.*, Aug. 2010, pp. 104–113.
- [23] O. Zamir and O. Etzioni, "Web document clustering: A feasibility demonstration," in *Proc. 21st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 1998, pp. 46–54.
- [24] A. Pérez-Suárez, J. F. Martínez-Trinidad, J. A. Carrasco-Ochoa, and J. E. Medina-Pagola, "OClustR: A new graph-based algorithm for overlapping clustering," *Neurocomputing*, vol. 121, pp. 234–247, Dec. 2013.
- [25] S. Baadel, F. Thabtah, and J. Lu, "Overlapping clustering: A review," in *Proc. SAI Comput. Conf. (SAI)*, Jul. 2016, pp. 233–237.
- [26] M. K. Afridi, N. Azam, and J. Yao, "Variance based three-way clustering approaches for handling overlapping clustering," *Int. J. Approx. Reasoning*, vol. 118, pp. 47–63, Mar. 2020.
- [27] H. Yu, C. Zhang, and G. Wang, "A tree-based incremental overlapping clustering method using the three-way decision theory," *Knowl.-Based Syst.*, vol. 91, pp. 189–203, Jan. 2016.
- [28] H. Yu, Y. Wang, and P. Jiao, "A three-way decisions approach to "density-based overlapping clustering," in *Transactions on Rough Sets XVIII*. Berlin, Germany: Springer, 2014, pp. 92–109.
- [29] Y. Yao, "The superiority of three-way decisions in probabilistic rough set models," *Inf. Sci.*, vol. 181, no. 6, pp. 1080–1096, Mar. 2011.

- [30] Y. Yao, "An outline of a theory of three-way decisions," in *Proc. Int. Conf. Rough Sets Current Trends Comput.* Berlin, Germany: Springer, 2012, pp. 1–17.
- [31] H. Yu, "A framework of three-way cluster analysis," in *Proc. Int. Joint Conf. Rough Sets*, in Lecture Notes in Computer Science, 2017, pp. 300–312.
- [32] W. Pedrycz, "Interpretation of clusters in the framework of shadowed sets," *Pattern Recognit. Lett.*, vol. 26, no. 15, pp. 2439–2449, Nov. 2005.
- [33] A. Campagner and D. Ciucci, "Orthopartitions and soft clustering: Soft mutual information measures for clustering validation," *Knowl.-Based Syst.*, vol. 180, pp. 51–61, Sep. 2019.
- [34] Y. Yao, P. Lingras, R. Wang, and D. Miao, "Interval set cluster analysis: A re-formulation," in *Proc. Int. Workshop Rough Sets, Fuzzy Sets, Data Mining, Granular-Soft Comput.* Berlin, Germany: Springer, 2009, pp. 398–405.
- [35] J. Zhou, Z. Lai, D. Miao, C. Gao, and X. Yue, "Multigranulation rough-fuzzy clustering based on shadowed sets," *Inf. Sci.*, vol. 507, pp. 553–573, Jan. 2020.
- [36] A. Campagner and D. Ciucci, "A formal learning theory for three-way clustering," in *Proc. Int. Conf. Scalable Uncertainty Manage.* Cham, Switzerland: Springer, 2020, pp. 128–140.
- [37] D. Ciucci, "Orthopairs and granular computing," *Granular Comput.*, vol. 1, no. 3, pp. 159–170, Sep. 2016.
- [38] X. Cheng and R. Wu, "Clustering path profiles on a website using rough k -means method," *J. Comput. Inf. Syst.*, vol. 8, no. 14, pp. 6009–6016, 2012.
- [39] S. Gupta and P. Kumar, "An overlapping community detection algorithm based on rough clustering of links," *Data Knowl. Eng.*, vol. 125, Jan. 2020, Art. no. 101777.
- [40] W. A. Hassanein and A. A. Elmelegy, "An algorithm for selecting clustering attribute using significance of attributes," *Int. J. Database Theory Appl.*, vol. 6, no. 5, pp. 53–66, Oct. 2013.
- [41] M. Li, S. Deng, L. Wang, S. Feng, and J. Fan, "Hierarchical clustering algorithm for categorical data using a probabilistic rough set model," *Knowl.-Based Syst.*, vol. 65, pp. 60–71, Jul. 2014.
- [42] P. Lingras, M. Chen, and D. Miao, "Qualitative and quantitative combinations of crisp and rough clustering schemes using dominance relations," *Int. J. Approx. Reasoning*, vol. 55, no. 1, pp. 238–258, Jan. 2014.
- [43] P. Lingras and G. Peters, "Applying rough set concepts to clustering," in *Rough Sets: Selected Methods and Applications in Management and Engineering*. London, U.K.: Springer, 2012, pp. 23–37.
- [44] V. P. Murugesan and P. Murugesan, "A new initialization and performance measure for the rough k -means clustering," *Soft Comput.*, vol. 24, no. 15, pp. 11605–11619, Aug. 2020.
- [45] I.-K. Park and G.-S. Choi, "Rough set approach for clustering categorical data using information-theoretic dependency measure," *Inf. Syst.*, vol. 48, pp. 289–295, Mar. 2015.
- [46] H. Yu, Z. Liu, and G. Wang, "An automatic method to determine the number of clusters using decision-theoretic rough set," *Int. J. Approx. Reasoning*, vol. 55, no. 1, pp. 101–115, Jan. 2014.
- [47] M. A. Ibrahim, T. O. William-West, A. F. D. Kana, and D. Singh, "Shadowed sets with higher approximation regions," *Soft Comput.*, vol. 24, no. 22, pp. 17009–17033, Nov. 2020.
- [48] X. Li, P. Geng, and B. Qiu, "A cluster boundary detection algorithm based on shadowed set," *Intell. Data Anal.*, vol. 20, no. 1, pp. 29–45, Jan. 2016.
- [49] W. Pedrycz, "Shadowed sets: Representing and processing fuzzy sets," *IEEE Trans. Syst., Man Cybern. B, Cybern.*, vol. 28, no. 1, pp. 103–109, Feb. 1998.
- [50] Y. Yao, S. Wang, and X. Deng, "Constructing shadowed sets and three-way approximations of fuzzy sets," *Inf. Sci.*, vols. 412–413, pp. 132–153, Oct. 2017.
- [51] X. R. Zhao and Y. Yao, "Three-way fuzzy partitions defined by shadowed sets," *Inf. Sci.*, vol. 497, pp. 23–37, Sep. 2019.
- [52] J. Zhou, W. Pedrycz, and D. Miao, "Shadowed sets in the characterization of rough-fuzzy clustering," *Pattern Recognit.*, vol. 44, no. 8, pp. 1738–1749, Aug. 2011.
- [53] P. Lingras and G. Peters, "Rough clustering," *Wiley Interdiscipl. Rev., Data Mining Knowl. Discovery*, vol. 1, no. 1, pp. 64–72, 2011.
- [54] Z. Pawlak, J. Grzymala-Busse, R. Slowinski, and W. Ziarko, "Rough sets," *Commun. ACM*, vol. 38, no. 11, pp. 88–95, 1995.
- [55] Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data*, vol. 9. Dordrecht, The Netherlands: Springer, 2012.
- [56] W. Pedrycz, "Shadowed sets: Bridging fuzzy and rough sets," *Rough Fuzzy Hybridization: A New Trend in Decision-Making*. Singapore: Springer, 1999, pp. 179–199.
- [57] M. Chen and D. Miao, "Interval set clustering," *Expert Syst. Appl.*, vol. 38, no. 4, pp. 2923–2932, Apr. 2011.
- [58] A. Polesel, G. Ramponi, and V. J. Mathews, "Image enhancement via adaptive unsharp masking," *IEEE Trans. Image Process.*, vol. 9, no. 3, pp. 505–510, Mar. 2000.
- [59] Y. Gu, R. K. Ganesan, B. Bischke, A. Bernardi, A. Maier, H. Warkentin, T. Steckel, and A. Dengel, "Grid-based outlier detection in large data sets for combine harvesters," in *Proc. IEEE 15th Int. Conf. Ind. Informat. (INDIN)*, Jul. 2017, pp. 811–818.
- [60] M. B. Goldstein, *Anomaly Detection in Large Datasets*. Munich, Germany: Verlag Dr. Hut, 2014.
- [61] A. Shah, N. Azam, E. Alanazi, and J. Yao, "Image blurring and sharpening inspired three-way clustering approach," *Int. J. Speech Technol.*, vol. 52, no. 15, pp. 18131–18155, Dec. 2022.
- [62] J. Yin, T. Tao, and J. Xu, "A multi-label feature selection algorithm based on multi-objective optimization," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2015, pp. 1–7.



ANWAR SHAH received the Ph.D. degree from the National University of Computer and Emerging Sciences, Pakistan, in 2022. He is currently an Assistant Professor with the Department of Computer Science, National University of Computer and Emerging Sciences. His scholarly contributions have found their place in prominent academic journals, underscoring his dedication to producing impactful research. His current research interests include three-way clustering, deep learning, rough sets, blockchain, metaverse, cybersecurity, and image processing.



BAHAR ALI received the Ph.D. degree in computer science. He is currently a Distinguished Computer Scientist specializing in machine learning, three-way clustering, information security, data science, and blockchain and cryptocurrency. He is also an Assistant Professor and an AI Coordinator with the National University of Computer and Emerging Sciences—FAST and actively involved in teaching students with the graduate, master's, and Ph.D. level. Highly skilled in various programming languages and possesses a comprehensive understanding of android OS security. His research contributions are published in leading journals. He actively participates in funded projects.



FAZAL WAHAB received the master's degree in software engineering from Northeastern University, Shenyang, China, in 2020. He is currently a Ph.D. Scholar with the College of Computer Science and Technology, Northeastern University. His research interests include cyber security (MANET and the IoT), computer networking (routing protocols), and deep learning.



INAM ULLAH received the B.Sc. degree in electrical engineering (telecommunication) from the Department of Electrical Engineering, University of Science and Technology Bannu (USTB), Khyber Pakhtunkhwa, Pakistan, in 2016, and the master's and Ph.D. degrees in information and communication engineering from the College of Internet of Things (IoT) Engineering, Hohai University (HHU), Changzhou Campus, China, in 2018 and 2022, respectively. He has completed

his postdoctoral research with Brain Korea 2021 (BK21), Chungbuk Information Technology Education and Research Center, Chungbuk National University, Cheongju-si, South Korea, in March 2023. He is currently an Assistant Professor with the Department of Computer Engineering, Gachon University, South Korea. His research interests include robotics, the Internet of Things (IoT), wireless sensor networks (WSNs), underwater communication and localization, underwater sensor networks (USNs), artificial intelligence (AI), big data, and deep learning. He has authored more than 70 peer-reviewed articles on various research topics. He is a TPC Member of ACM RACS 2023, Poland, in August 2023. He served as a Guest Editor for various journals, such as *Computers in Human Behavior*, *Sensors*, *Electronics*, *Journal of Marine Science and Engineering*, *Frontiers in Sensors*, and *Artificial Intelligence and Applications*. He is a Reviewer of many prominent journals, including IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, *KSII Transactions on Internet and Information Systems*, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING, IEEE ACCESS, *Sustainable Energy Technologies and Assessments*, *Future Generation Computer Systems* (FGCS), *Computers and Electrical Engineering* (Elsevier), IEEE INTERNET OF THINGS JOURNAL, *Digital Communications and Networks* (Elsevier), *Nature* (Springer), *Wireless Communication and Mobile Computing* (WCMC), *Alexandria Engineering Journal*, *Sensors*, *Electronics*, *Remote Sensing*, *Applied Sciences*, and *Computational Intelligence and Neurosciences*. His awards and honors include the Best Student Award from USTB, in 2015, where he was also received the Prime Minister Laptop Scheme Award, in April 2015. Top-10 Students Award of the College of Internet of Things (IoT) Engineering and the Top-100 Students Award, HHU, in June 2019; the Jiangsu Province Distinguish International Students Award (30,000 RMB) in 2019 and 2020; the Certificate of Recognition from HHU, in 2021 and 2022, where he was also received the Top-100 Students Award, in May 2022; the Top-10 Outstanding Students Award, HHU, in June 2022; and the Distinguished Alumni Award from USTB, in October 2022.



FAYEZ ALQAHTANI is currently a Full Professor with the Software Engineering Department, College of Computer and Information Sciences, King Saud University (KSU). He was appointed as the Director of the Computer Division, Deanship of Student Affairs. He is also a member of several academic and professional associations, such as the Association for Computing Machinery (ACM), the Australian Computer Society, and the Association for Information Systems. He has conducted

research projects in several areas of information and communication technology, such as web 2.0, information security, enterprise architecture, software process improvement, the Internet of Things, and fog computing. He has participated in several academic events.



AMR TOLBA (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees from the Mathematics and Computer Science Department, Faculty of Science, Menoufia University, Egypt, in 2002 and 2006, respectively. He is currently a Full Professor in computer science with King Saud University (KSU), Saudi Arabia. He has authored/coauthored over 160 scientific articles in top-ranked (ISI) international journals, such as IEEE INTERNET OF THINGS JOURNAL, *ACM TOIT*, *IEEE CEMAG*, IEEE

ACCESS, IEEE SYSTEMS JOURNAL, *FGCS*, *JNCA NC&A*, *JAIHC*, *COMNET*, *COMCOM*, *P2PNET*, *VCOM*, and *WWWJ*. He has translated four books into the Arabic language. His main research interests include artificial intelligence (AI), the Internet of Things (IoT), data science, and cloud computing. He served as a Technical Program Committee (TPC) Member at several conferences, such as DSIT 2022, CICA2022, EAI MobiHealth 2021, DSS 2021, AEMCSE 2021, ICBDM 2021, ICISE 2021, DSS 2020, NCO 2020, ICISE2019, ICCSEA 2019, DSS 2019, FCES 19, ICISE 2018, ESG'18, Smart Data'17, NECO 2017, NC'17, WEMNET'17, NET'17, and Smart Data'16. He has been included in the list of the top 2% of influential researchers globally (prepared by scientists from Stanford University, USA), in 2020, 2021, and 2022, respectively. He served as an associate editor/guest editor for several ISI journals.

...