Clustering Using Shared Reference Points Algorithm Based On a Sound Data Model

Mohamed A. Abbas *mohamed.alyabbas@gmail.com*
Graduate student with the College of Computing and Information Technology
Arab Academy for Science and Technology
Alexandria, P.O. Box 1029, *Egypt.*

Amin A. Shoukry *amin.shoukry@ejust.edu.eg*
Department of Computer Science and Engineering
Egypt-Japan University of Science and Technology
New Borg El-Arab City, P.O. Box 179, Egypt

Abstract

A novel clustering algorithm CSHARP is presented for the purpose of finding clusters of arbitrary shapes and arbitrary densities in high dimensional feature spaces. It can be considered as a variation of the Shared Nearest Neighbor algorithm (SNN), in which each sample data point votes for the points in its k-nearest neighborhood. Sets of points sharing a common mutual nearest neighbor are considered as dense regions/ blocks. These blocks are the seeds from which clusters may grow up. Therefore, CSHARP is not a point-to-point clustering algorithm. Rather, it is a block-to-block clustering technique. Much of its advantages come from these facts: Noise points and outliers correspond to blocks of small sizes, and homogeneous blocks highly overlap. The proposed technique is less likely to merge clusters of different densities or different homogeneity. The algorithm has been applied to a variety of low and high dimensional data sets with superior results over existing techniques such as DBScan, K-means, Chameleon, Mitosis and Spectral Clustering. The quality of its results as well as its time complexity, rank it at the front of these techniques.

Keywords: Shared Nearest Neighbors, Mutual Neighbors, Spatial Data, High Dimensional Data, Time Series, Cluster Validation.

1. INTRODUCTION

The present paper is a modified version of [25]. The modification includes the incorporation of a new measure of cluster homogeneity (section 2.2) which has been used in defining a strict order for cluster's propagation in the proposed algorithm (section 3). Also, new validation indexes; as well as new data sets; have been adopted for the purpose of comparing the proposed algorithm with the previous techniques (section 4).

Clustering of data is an important step in data analysis. The main goal of clustering is to divide data objects into well separated groups so that objects lying in the same group are more similar to one another than to objects in other groups.

Given a set $\mathbf{P} = \{p_1, p_2, \ldots, p_N\}$ of data objects (sample points) to be clustered, where p_i is the i-th object. p_i is an n-dimensional column vector. $p_i = [p_{i1}, p_{i2}, \ldots, p_{in}]^T$ consisting of n measured attributes (n is the dimensionality of the feature space). The Jarvis-Patrick clustering technique [12], needs a measure of the distance between two objects and two integers: K and L. K is the size of the neighborhood list, and L is the number of common neighbors. This method works as follows:
Determine the K-nearest neighbors for each object in the set to be clustered. Two objects are placed in the same cluster if they are contained in each other's K nearest neighbors list: $p_i \in$ K-NB$_j$ and $p_j \in$ K-NB$_i$, where K-NB$_j$ and K-NB$_i$ denote the K-nearest neighbors of data points p_j and p_i, respectively. (1)
They have at least L nearest neighbors in common: $|NB_j \cap NB_i| > L$. (2)

As stated in [15], the principle of K-NB consistency of a cluster states that for any data object in a cluster its K-Nearest Neighbors should also be in the same cluster. The principle of K-Mutual Nearest-Neighbor consistency (K-MNB consistency) states that for any data object in a cluster its K-Mutual Nearest-Neighbors should also be in the same cluster. The principle of cluster K-MNB consistency is stronger than the cluster K-NB consistency concept, and it is also a more strict representation of the natural grouping in the definition of clustering. In the present work, the concept of K-MNB is used in developing a new clustering algorithm, CSHARP, (Clustering using SHAred Reference Points). CSHARP is a density based clustering algorithm in which dense regions are identified using mutual nearest neighborhood. Then, sets of points sharing a common mutual nearest neighbor are considered instead of points themselves in an agglomerative process.

1.1 Proposed Technique

Specifically, the technique proposed in this paper:

- Determines; for every data point; the largest possible set of points satisfying condition (1) only (the cardinality of this set lies in the range $[1 \dots K]$). For a point "p", this set is called its Reference-List and is denoted RL_{p_i}. Point "p" is considered as the "Representative Point" (or "Reference Point") of this Reference-List. Such sets of points are the seeds from which clusters may grow up.
- Avoids an early commitment to condition (2), and, instead, proceeds directly from point(s) to set(s) relation(s) (the Reference-Lists); instead of point(s) to point(s) relation(s) (as required by conditions (1) and (2), above). Therefore, data in CSHARP is processed as blocks of points not as individual points. Reference lists constitute a key concept in the proposed algorithm. To preserve K-MNB consistency, CSHARP processes data as blocks of tiny groups of Reference-Lists, on which clusters are built, agglomeratively.
- Allows clusters to be linked if their Reference- Lists share a number of points $\geq M$. Thus, parameter M controls the extent to which mutual neighborhood consistency is satisfied. Allowing clusters to grow in a chain is similar; though not identical; to the reachability relation in DBScan algorithm [5]. Reference-Lists of size $< T$ (a selected threshold) are either considered as noise or may merge with other clusters as will be explained in section 2.

Several experiments conducted on a variety of data sets show the efficiency of the proposed technique for the detection of clusters of different sizes, shapes and densities; whether in low or high dimensional feature spaces; in the presence of noise and outliers. Although the motivations behind the algorithm are quite heuristic, the experimental work showed the validity of the proposed approach.

1.2 Overview of Related Algorithms

Many clustering techniques have been developed based on different concepts. Several approaches utilize the concept of cluster center or centroid, other methods build clusters based on the density of the objects, and a lot of methods represent the data objects as vertices of graphs where edges represent the similarity between these objects.

Centroid based algorithms represent each cluster by using the centre of gravity of its instances. The most well-known centroid algorithm is the K-means [11]. The K-means method partitions a data set into k subsets such that all points in a given subset are close to the same centre. K-means then computes the new centers by taking the mean of all data points belonging to each cluster. The operation is iterated until there is no change in the centers locations. The result strongly depends on the initial guess of centroids, besides, it does not perform well on data with outliers or with clusters of different sizes or non globular shapes.

The key idea of density-based clustering is that for each instance of a cluster, a neighborhood of a given radius has to contain at least a minimum number of instances. One of the most well known density-based clustering algorithms is the DBScan [5]. In DBScan the density associated with an object is obtained by counting the number of objects in a region of a specified radius, ε, around the object. An object with density greater than or equal to a specified threshold, MinPts, is treated as a core (dense), otherwise it is considered as a non-core (sparse) object. Non-core objects that do not have a core object within the specified

radius are discarded as noise. Clusters are formed around core objects by finding sets of density connected objects that are maximal with respect to density-reachability. While DBScan can find clusters of arbitrary sizes and shapes, it cannot handle data containing clusters of different densities, since it uses global density parameters, MinPts and ε, which specify only the lowest possible density of any cluster.

Chameleon [13] and SNN [4] algorithms attempt to obtain clusters with variable sizes, shapes and densities based on K-nearest neighbor graphs. Chameleon finds the clusters in a data set by using a two-phase algorithm. In the first phase, it generates a K-nearest neighbor graph that contains links between a point and its K-nearest neighbors. Then it uses a graph partitioning algorithm to cluster the data items into a large number of relatively small sub-clusters. During the second phase, it uses an agglomerative hierarchical clustering algorithm to find the genuine clusters by repeatedly combining together these sub-clusters. The Shared Nearest Neighbors clustering algorithm, SNN [4] uses K-nearest neighbor approach for density estimation. It constructs a K-nearest neighbor graph in which each data object corresponds to a node which is connected to the nodes corresponding to its K-nearest neighbors. From the K-nearest neighbor graph a shared nearest neighbor graph is constructed, in which edges exist only between data objects that have each other in their nearest neighbor lists. A weight is assigned to each edge based on the number and ordering of shared neighbors. Clusters are obtained by removing all edges from the shared nearest neighbor graph that have a weight below a certain threshold t.

A recent clustering algorithm, Mitosis [22], is proposed for finding clusters of arbitrary shapes and arbitrary densities in high dimensional data. Unlike previous algorithms, it uses a dynamic model that combines both local and global distance measures. The model is depicted in the proposed dynamic range neighborhood, and the proposed clustering criteria which use distance relatedness to merge patterns together. Mitosis uses two main parameters f and k. Parameter f, controlling the neighborhood size, is the main parameter which decides the lower bound on the number of clusters that can be obtained. The value of f, should be varied in an incremental fashion so as not to deteriorate the speed of the algorithm. It can be selected just above the value of 1, and increased by small steps, to avoid unnecessary large neighborhood sizes. Parameter k controls the degree of merging patterns/clusters together, within the limits of the neighborhood decided by f. Increasing values of k, for the same f value, means decreasing the number of clusters obtained, and vice versa.

In recent years, spectral clustering [20] has become one of the most popular modern clustering techniques. It starts from a similarity matrix between data objects, modifies it to a sparse matrix, then computes its Laplacian matrix "L". The first k eigenvectors of L constitute the first k columns of a matrix V. K-means algorithm is applied then on the row vectors of a normalized version of matrix V (where each original data object is assigned to the same cluster to which the corresponding row vector in V is assigned to). Spectral clustering is simple to implement and can be solved efficiently on standard linear algebra software. Additionally, it is more effective in finding clusters than some traditional algorithms such as K-means. However, it suffers from a scalability problem discussed in [17]. It cannot successfully cluster data sets that contain structures at different scales of size and density. To overcome these limitations, a novel spectral clustering algorithm [2] is proposed for computing spectral clustering using a sparse similarity matrix.

1.3 Outline of the Paper

The rest of this paper is organized as follows. Section 2 describes our approach for the definition of similarity and density (or reference points structure), which is the key concept to our clustering algorithm. Section 3 describes the algorithm itself followed by a logical model of the data entities it manipulates, a graph-based interpretation of its effect and its time complexity analysis. An anatomy of the proposed algorithm with respect to related algorithms is discussed next. Section 4 presents the data sets used in the experiments conducted to evaluate the performance of the algorithm using well known cluster validation indexes. Section 5 presents a short conclusion and possible future work.

2. BASIC DEFINITIONS

In this section we describe the basic concepts used in the proposed algorithm.

2.1 Reference-List and Reference-Block

As shown in Figure 1, although the Euclidean distance is a symmetric metric, from the SNN perspective (and considering K=4), point A is in 4-NB_B, however, point B is not in 4-NB_A. Given a set of points P, let $P = \{p_1, p_2, \ldots, p_k, p_{k+1}, p_{k+2}, \ldots, p_N\}$ be the ordered list of points according to their distances from a point p_i, and let K-NB$p_i = \{p_{i1}, p_{i2}, p_{i3}, \ldots, p_{ik}\}$ be the K-nearest neighbors of p_i. This represents the list of points that point p_i refers to. If $p_j \in$ K-NBp_i, then p_j is referred by p_i. Let RBp_i denote the set of points having p_i in their K-Nearest Neighborhood. Then, K-NB$p_i \cap$ RBp_i represents a set of dense points called a Reference-List, RLp_i, associated with point p_i. p_i is called the representative point of RLp_i. Each representative point p_i with its Reference-List, RLp_i constitute a Reference-Block. Points in a Reference-Block are shown in Figure 2.

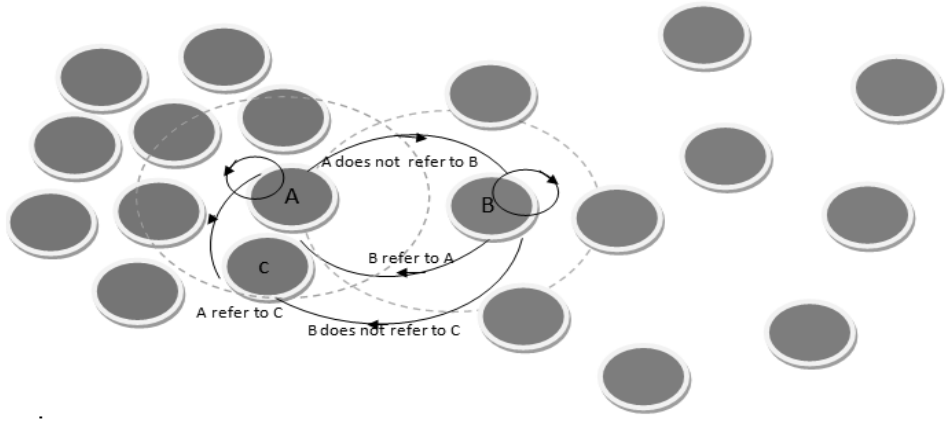


FIGURE 1: Concept of mutual neighboring: "A" is in 4-NBB, however, "B" is not in 4-NBA.

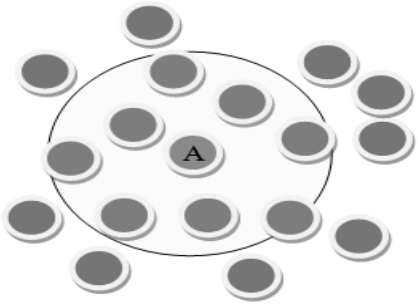


FIGURE 2: A Reference-List (points within circle), associated with a representative point "A"). All together they constitute a Reference-block.

There are three possible types of representative points:

- Strong Points (or Reference Points), representing blocks of size greater than a pre-defined threshold parameter T.
- Noise points, representing empty Reference-Lists. These points will be excluded initially from final clusters.
- Weak points which are neither strong points nor noise points. These points may be merged with another existing clusters if they are members of another strong points.

2.2 Homogeneity Factor

A homogeneity measure is proposed here for the ordering of the cluster's propagation process. While CSHARP[25] performs this process according to the cardinality of the reference lists (i.e their densities), the algorithm proposed here, considers both the density and homogeneity of the blocks that will be granted priority to propagate first.

Given $RL_{p_i} = \{q_1, q_2, \ldots q_c\}$, a Reference List associated with point p_i having cardinality $c = |RL_{p_i}| > T$ (i.e. corresponding to a strong reference point), its homogeneity factor α_i is computed as:

$$\alpha_i = \frac{Avg_{p_{ij}}}{Max_{p_{ij}}} \tag{3}$$

Where $Avg_{p_{ij}}$ is the average distance between p_i and its associated reference list points q_j, computed as:

$$Avg_{p_{ij}} = \frac{1}{c}\sum_{j=1}^{c} |p_i - q_j|,\ q_j \in RL_{p_i} \tag{4}$$

and $Max_{p_{ij}}$ is the maximum distance between p_i and its associated reference list points q_j, computed as:

$$\max |p_i - q_j|,\ q_j \in RL_{p_i} \tag{5}$$

Figure 3 shows three cases of strong reference lists with their corresponding homogeneity factors. Note that the odd case of a reference list with cardinality c=1 in which α_i=1 is excluded, since it does not correspond to a strong reference list. In the experiments described in section four, T is always greater than 2.

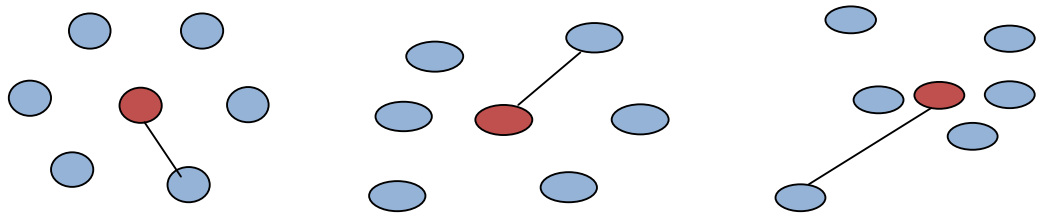


FIGURE 3: Three reference lists with different homogeneity factors (a) 0.98, (b) 0.80 and (c) 0.55

2.3 Cluster's Propagation

Given two clusters c_i, c_j such that $|c_i \cap c_j| \geq M$; where M is a chosen parameter, called merge-parameter, then the two clusters can be merged together. Hence, M measures the minimum link strength required between two clusters. In CSHARP data is processed as blocks of points (reference-blocks) not as individual points. Two clusters are merged if they share a number of points equal to or greater than some threshold M; as explained in Figure 4.

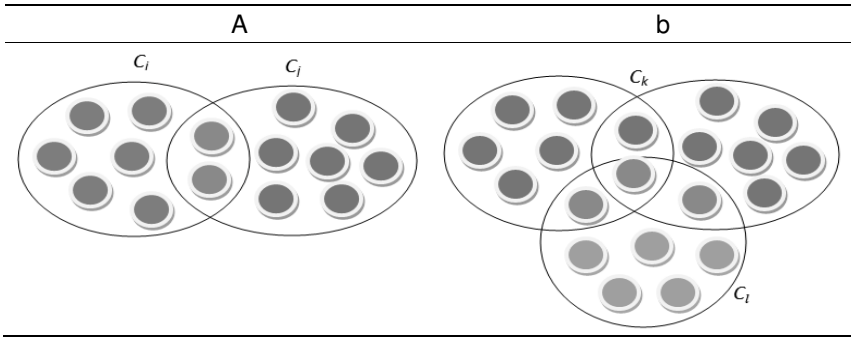


FIGURE 4: Two steps of Cluster propagation with M = 2 (the merging parameter) for (a) Ck = $|c_i \cup c_j|$ such that $|c_i \cap c_j| \geq$ M and (b) Cu = Ck ∪ Cl such that $|c_k \cap c_l| \geq$ M.

To illustrate the process of clusters propagation, Chameleon's data set DS5 is used. DS5 consists of 8000 spatial data points. All genuine clusters were detected at the parameters setting(K=24, T=18, M=6). The number of strong points obtained were 5122.

Several snapshots of the clustering process are shown in Figure 5. Figure 5, also, illustrates how clusters propagate agglomeratively, and simultaneously, in CSHARP.

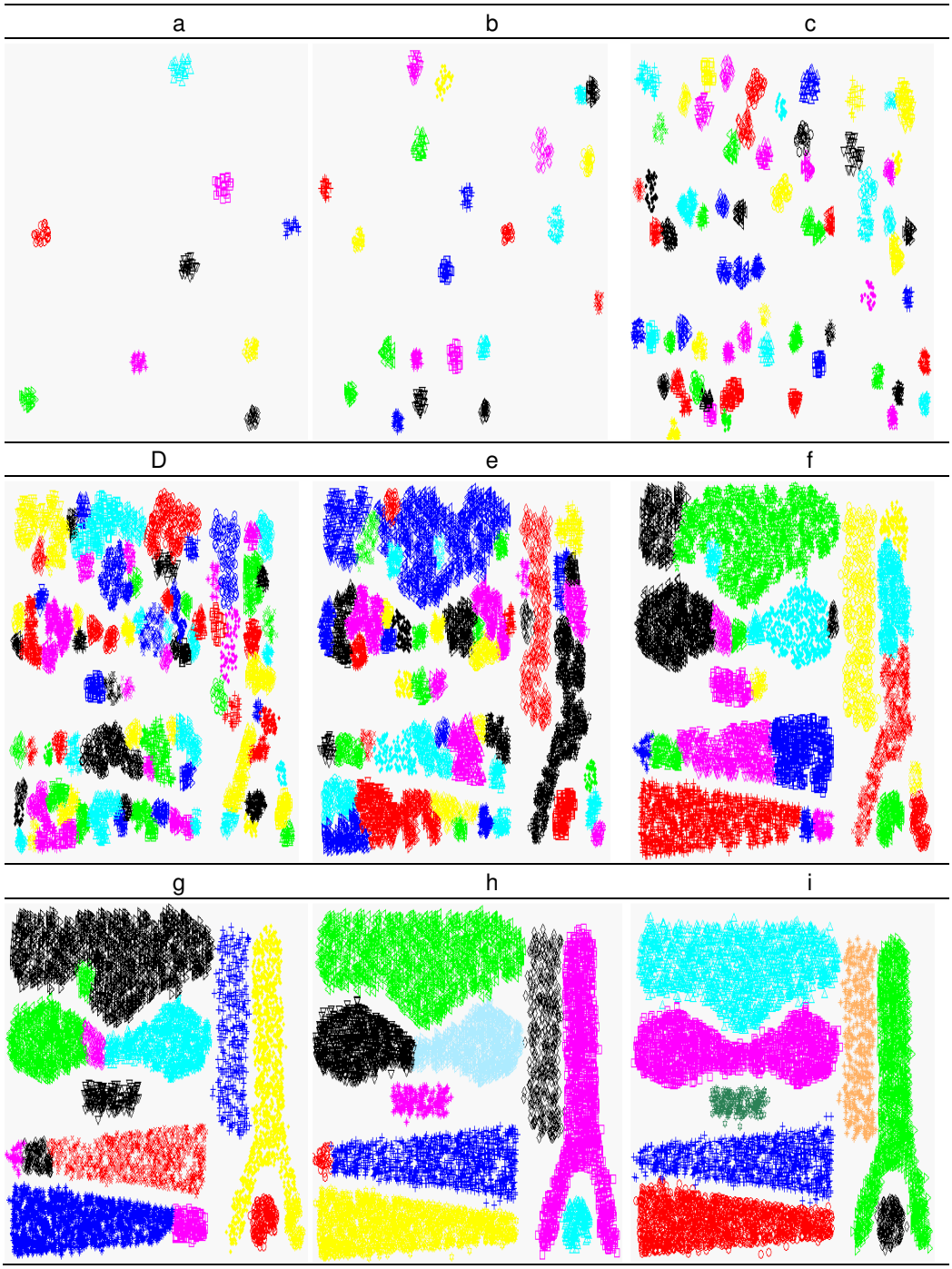


FIGURE 5: Nine snapshots of Modified CSHARP cluster propagation for the data set Chameleon DS5 at different iterations (a) 10, (b) 25, (c) 100, (d) 500, (e) 1000, (f) 2000, (g) 3000, (h) 4000 and (i) 5122.

3. MODIFIED CSHARP ALGORITHM

Figure 6 describes the modified CSHARP algorithm. Next, a logical data model and a graph-based interpretation of the algorithm are given. Finally, the time complexity of this algorithm is analyzed in section 3.2.

Input: Data points $\mathbf{P} = \{p_1, p_2 \ldots, p_n\}$;
$\mathbf{K}$ {size of the neighborhood of a point};
$\mathbf{T}$ {threshold parameter for the size of a reference list} and
$\mathbf{M}$ {merge parameter for the size of the intersection between two reference blocks}.

Output: $\mathbf{C}$ set of generated clusters.

Construct similarity matrix $\mathbf{S}$.

{Construct the Refer-To-List, K-NBp_i for each point $p_i \in P$}

K-NB$p_i = \{p_j \mid d(p_i, p_j) \leq d(p_i, p_k)\}$

{construct the Referred-By-List, RBp_i for each point $p_i \in P$.

for all $p_i \in P$ **do**
    $V \leftarrow$ K-NBp_i
    for all $v_j \in V$ **do**
        If $p_i \in$ K-NBv_j **then**
            $v_j \in$ RBp_i
        end if
    end for
end for

{From K-NBp_i and RBp_i, Construct the reference Lists RLp_i

for all $p_i \in P$ **do**
    RLp_i= K-NB$p_i \cap$ RBp_i.
end for

Form a sorted (in a descending order) list L. L = $\{p_i \mid p_i$ is a representative point$\}$, based on the densities (i.e. $|RL_{p_i}|$). Exclude the weak points from list L (those points for which $|RL_{p_i}| < T$).

Sort the new list L' according to the homogeneity factors α_i's of the Reference-Lists RLp_i.

{Building Clusters}

$i \leftarrow 0$ {Initialize i}
$C_0 = \cup \{p_0, RL_{p_0}\}$
label point p_0 as belonging to cluster C_0.
label each point in RLp_0 as belonging to cluster C_0.
While $i \leq |L'|$ **do**
    $i \leftarrow i + 1$ {increment i}
    $C_i = \cup \{p_i, RL_{p_i}\}$
    label point p_i as belonging to cluster C_i.
    label each point in RLp_i as belonging to cluster C_i.
    if $|\{p_i \cup RL_{p_i}\} \cap C_u| \geq M$, {where $u = 1,2,\cdots,i\text{-}1$} **then**
        $C_i = \cup \{p_i, RL_{p_i}, C_u\}$
        Update C_u labels, by marking each point in C_u as belonging to cluster C_i
    end if
end while

FIGURE 6: CSHARP Clustering Algorithm

Note that although a reference list associated with a weak point does not participate in

Cluster's growing, a weak point may itself belong to a Reference list of another strong point. This implies that this weak point is not considered as noise.

3.1 Logical Data Model and Graph-based Interpretation

Figure 7 depicts the conceptual data model underlying the proposed algorithm. Two entities are shown; namely, the data sample and Reference-List associated with a data sample entities; as well as a one one-to-many" relation between them and one "many-to-many" relation associated with each of them.

- The relation "Has as member in its K-NB list" is not symmetric.
- The relation "Has as member" between a Reference List and its members is symmetric.
- The relation "Overlaps" is symmetric but not transitive.
- The entity "Reference List Associated with a Data sample"; say "$\mathbf{p}$", is obtained as the intersection of two sets: the set of points having μ in their K-Nearest Neighborhood and the K-Nearest Neighborhood of "$\mathbf{p}$" itself.

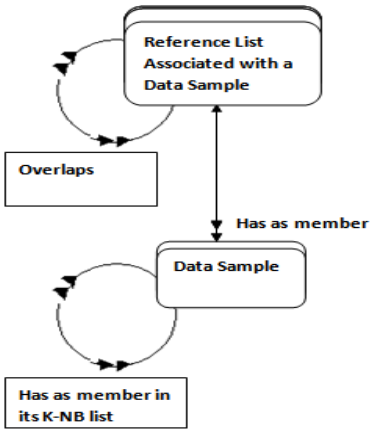


FIGURE 7: Conceptual data model.

Consider the following weighted graph $(\mathbf{V},\mathbf{E})$, where a vertex in this graph corresponds to a Reference-List and an edge corresponds to an Overlaps relation between two Reference Lists. The weight of an edge corresponds to the cardinality of the intersection set between two Reference lists. Now, the effect of the CSHARP algorithm can be viewed as follows: Edges with weights less than the threshold $\mathbf{M}$ are removed. This decomposes the graph into independent components. The remaining connected vertices (i.e. vertices connected by edges having weights greater than $\mathbf{M}$) are combined, the union of the data samples belonging to their reference lists correspond to the obtained clusters.

3.2 Time Complexity

The time complexity for computing the similarity matrix is $O(N^2)$, where N is the number of data points. This can be reduced to O(N log N), by using a data structure such as a k-d tree [1] or an R-tree[6]. The space complexity for t computing distances between samples is O(NF) where N is the number of data points and F is the number of features (or dimensions).

The time complexity of the algorithm can be analyzed as follows:

- line 2, finding k-nearest neighbors(refer-to-list): k iterations through all N data points are needed, hence it has a complexity of O(NK)
- lines 3-10, finding referred-by-list: it has, also, a complexity of O(NK)
- lines 11-13, finding reference-points: for each data point, its k-nearest neighbors are searched for mutual neighborhood. As a binary search is adopted; its complexity is of O(K log K), thus, the overall complexity of this step is O(NK log k).
- Line 14, computing homogeneity factor procedure: has a complexity of $\mathbf{O(NK)}$, as k iterations are needed for each of the N data points .

- line 15, sorting data sets procedure: sorting has a complexity of O(N log N), as binary sort is used.
- lines 25-28, clusters overlapping procedure: detecting overlapping among clusters has a linear complexity of O(K),since we iterate through reference-blocks of processed points and detect previously labeled points.
- lines 20-29, clusters propagation procedure: this is the main procedure. It has a complexity of O(NK) , since we iterate through all N data points, running clusters overlapping procedure for each data point. Accordingly, the overall complexity is O(NK log N) .

The overall time complexity for the modified CSHARP algorithm is O(NK log N) where N is the number of data points and K is the number of nearest neighbors. Modified CSHARP has a space complexity of O(NK), where N is the number of data points and K is the number of nearest neighbors used

3.3 Anatomy of CSHARP vs. Jarvis-Patrick and SNN Algorithms

- In SNN, similarity between two points p and q is computed as the number of nearest neighbors they share. In contrast, CSHARP's similarity is computed as the number of reference points two blocks share.
- In contrast to Jarvis-Patrick and SNN, CSHARP is a block-to-block rather than point-to-point clustering.
- Jarvis-Patrick and SNN work with k-nearest neighborhood which corresponds to a non-symmetric relationship between data points. On the other hand, CSHARP relies on a symmetric relation between any point and its reference list points.
- Jarvis-Patrick and SNN use static k-nearest neighbor lists, while CSHARP starts with static k-nearest neighbor lists then turns them into dynamic Reference Lists.
- In Jarvis-Patrick and SNN, the association between points is one-to-one, while it is one-to many in CSHARP.
- Both SNN and CSHARP propagate clusters simultaneously and agglomeratively, while Jarvis-Patrick builds a similarity graph then decomposes it by removing the weakest links.

4. EXPERIMENTAL RESULTS

4.1 Datasets Used

4.1.1. 2-d Chameleon's DS5 data set

DS5 consists of 8,000 points. It is a mixture of clusters with different densities used by Chameleon algorithm to illustrate its efficiency in obtaining arbitrary density clusters. The aim is to classify the data into 8 clusters of different densities.

4.1.2. Eight Low Dimensional Datasets

- Iris data set: This is a well known database in the pattern recognition literature. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the others 2; the latter are NOT linearly separable from each other [16].
- The Synthetic Control Charts (SCC) data set; obtained from UCR repository [14]; it includes 600 patterns, each of 60 dimensions (time points).
- The pen digit character recognition data from the UCI repository [16], consists of 10992 patterns, each of 16 dimensions. Features (dimensions) are used to describe the bitmaps of the digit characters. The aim is to properly classify the digit characters to 10 classes from 0 to 9.
- Libras movement data set: The dataset contains 15 classes of 24 instances each, where each class references to a hand movement type. It consists of 360 patterns, each of 91 dimensions.
- The Breast Cancer Wisconsin Diagnostic data from the UCI repository, consists of 569 patterns, each of 30 dimensions. The aim is to classify the data into two diagnosis (malignant or benign).
- SPECT heart data set: The dataset describes diagnosing of cardiac Single Proton

Emission Computed Tomography (SPECT) images. It consists of 267 patterns, each of 22 dimensions. Each of the patients is classified into two categories: normal and abnormal

- The Protein localization data; obtained from the UCI repository; is the Ecoli data set with 336 proteins of seven dimensions. The aim is to properly classify it to eight classes.
- The Protein Localization Sites, obtained from the UCI repository, consists of 1484 patterns, each of 8 dimensions. This database contains information about a set of Yeast cells. The task is to determine the localization site of each cell by partitioning it into 10 classes of varying distribution.

4.1.3. Two High Dimensional Datasets

- Corel image features data set: This dataset contains image features extracted from a Corel image collection. 2074 images of 144 dimensions were selected in this experiment according to criteria discussed in [2].
- Arcene data set which consists of 200 patterns each of 10,000 features. The task is to distinguish cancer versus normal patterns from mass-spectrometric data. This is a two-class classification problem with continuous input variables. Arcene has been part of the NIPS 2003 feature selection challenge.

4.2 Cluster Validation

As described in [21], for a given data set, a clustering algorithm can always produce a partitioning whether or not a particular structure in the data really exists. Different clustering approaches usually yield different results. Even for the same algorithm, the selection of a parameter or the presentation order of the input patterns may affect the final results. Therefore, effective evaluation criteria are critically important to provide users with a degree of confidence in the obtained clustering results. These assessments should be objective and have no preferences to any algorithm. V-measure [24], Purity and Entropy [23] are used for the purpose for clustering validation. Noise is taken into consideration in the validation process. This reduces the indexes values than if a noise-free validation process is adopted. However, this presents a more accurate assessment.

4.3 Results and Performance Evaluation

To compare the results obtained by the Modified CSHARP with those obtained by other algorithms, the nine data sets presented above have been used.

4.3.1. Chameleon's DS5 Data Set.

Five algorithms are compared: K-means, DBScan, SNN, Chameleon, and mitosis, in addition to Modified CSHARP.

Due to the presence of different densities in the DS5 data set, DBScan either identifies the lower density cluster but merges the two neighboring higher density ones, or do not identify the lower density cluster, but identifies the higher density ones. DBScan at the parameters setting (ε = 10 and MinPts = 3) can only identify six rather than eight clusters. Similarly, SNN merges two clusters together due to its static nature derived from DBScan. Both DBScan and SNN , (Figure 8b, and d) were unable to obtain a good clustering solution. Modified CSHARP obtained the genuine clusters at the parameters setting given in Table 1b for K in the range [23...25], Figure 8f.

Mitosis has been able to obtain the genuine clusters at parameter settings (f = 2.15 and k = 2.5) (Figure 8e). Mitosis results were obtained after discarding outliers. Clusters of sizes less than 1% of the data size are identified as outliers. While Mitosis uses a static model for discarding noise, CSHARP [25] focuses on the detection of chains of dense connected regions (defined by strong points) possibly including non-dense regions ; represented by weak points; as explained in section 2.1. Hence, in CSHARP any point not taken into consideration during cluster's propagation is considered as noise. Table 2 gives the numbers and percentages of strong, weak and noise points found in all the investigated data sets.

TABLE 1: Modified CSHARP's range of parameters setting for DS5 data set.

K	T	M
23	[3...5]	7
23	19	[6,7]
23	22	[6…9]
24	[15…17]	7
24	18	6
24	22	[6...9]
24	23	[4...9]
25	[16…18]	[7,8]
25	22	[6...9]
25	23	[5...9]

4.3.2. Eight Dimensional Data

Five data sets have been used to compare the results obtained by Modified CSHARP to the ground truth as well as to the results obtained by DBScan, K-means, Mitosis and Spectral clustering (as a state-of- the art technique [2]) algorithms. Chameleon and SNN have not been included in these experiments, due to the difficulty of adjusting their parameters. The Euclidean distance has been adopted as a metric for all data sets. Numerous experiments (100 experiments at least per algorithm per dataset) have been done on each algorithm to obtain its best indexes' values. V-Measure, Purity, and Entropy have been used to evaluate all the above clustering algorithms.

TABLE 2: Number of strong, weak and noise points found in the tested data sets and their percentages ;at given parameters settings; relative to the size of the corresponding data set.

Dataset	size	Strong Points	Weak Points	Noise Points
Chameleon DS5 (K=24, T=18 and M=6)	8000	6399 (79.98%)	1601 (20.02%)	225 (2.81%)
Synthetic Control Charts (K=14, T=7 and M=4)	600	479 (79.83%)	121 (20.17%)	1 (0.17%)
Pen Digit Data (K=31, T=14 and M=9)	10992	8732 (79.44%)	2260 (20.56%)	253 (2.30%)
Breast Cancer Diagnostic (K=41, T=22 and M=14)	569	492 (86.47%)	77 (13.53%)	15 (2.64%)
Ecoli (K= 22, T=11 and M=8)	336	240 (71.43%)	96 (28.57%)	25 (7.44%)
Yeast (K= 44, T=25 and M=15)	1484	848 (57.14%)	636 (42.86%)	134 (9.03%)
Arcene (K= 11, T=3 and M=1)	200	180 (90.0%)	20 (10.0%)	4 (2.0%)
SPECT Heart (K= 30, T=7 and M=4)	267	222 (83.15%)	45 (16.85%)	18 (6.74%)
Libras Movement (K= 10, T=7 and M=3)	360	173 (48.06%)	187 (51.94%)	46 (12.78%)
Iris (K= 24, T=8 and M=9)	360	138 (92.00%)	12 (8.00%)	0 (0.00%)
Corel (K= 43, T=21 and M=13)	2074	1230 (59.31%)	844 (40.69%)	120 (5.78%)

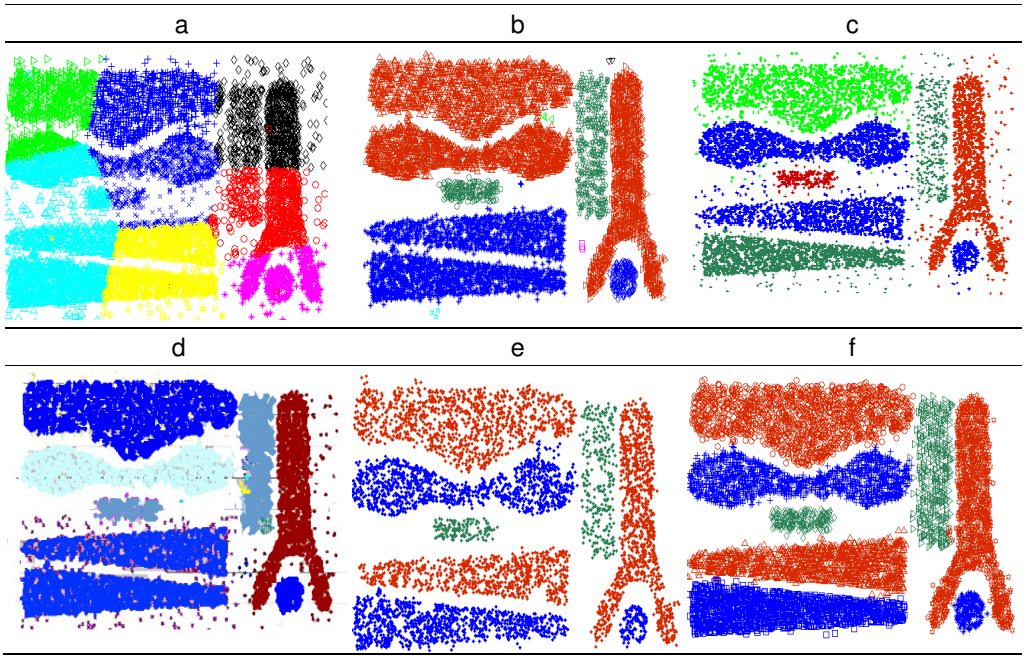


FIGURE 8: Results for the DS5 data set. Clusters are identified by a specific color, and a specific marker for: (a) K-means clusters; (b) DBScan clusters; (c) Chameleon clusters; (d) SNN clusters; (e) Mitosis clusters; and (f) Modified CSHARP clusters.

- Iris data set: Modified CSHARP and spectral clustering performed better for this data set, reached the highest indexes for F-measure and Purity and lowest index for entropy as shown in Figure 10a. The concept of relatedness fails with iris data set, since it consists of three classes, two of them are not linearly separable; thus, Due to the merging of these two clusters, Mitosis obtained the lowest indexes for this data set. Mitosis failed to detect the original three clusters and detected only two clusters. On the contrary, Modified CSHARP and Spectral clustering detected the three original clusters, Figure 11.
- Time series data: For SCC, The original six clusters are shown in Figure 9. All algorithms (with the exception of CSHARP) failed to discover the cluster labeled normal, due to its relatively low density. DBScan using the parameters setting (ε = 50 and MinPts = 3) detected the cyclic cluster. However, it merged the Up-Shift, and the Up-Trend clusters together, as well as the Down-Shift and the Down-trend clusters. K-means, at $k = 6$ merged patterns from different clusters together. The values of the V-Measure, Purity and Entropy are shown in Figure 10b. Due to the discovery of the Normal low dense cluster; at the parameters setting (K=11, T=5 and M=3); CSHARP reached the maximum values for both V-measure and Purity indexes and the lowest for Entropy index, Figure 12.

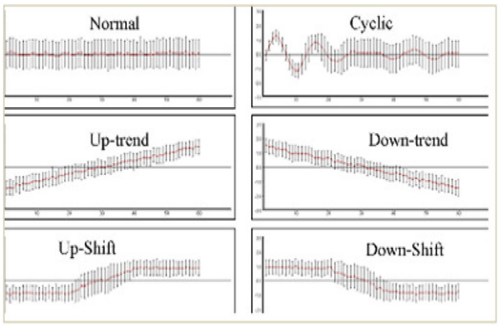
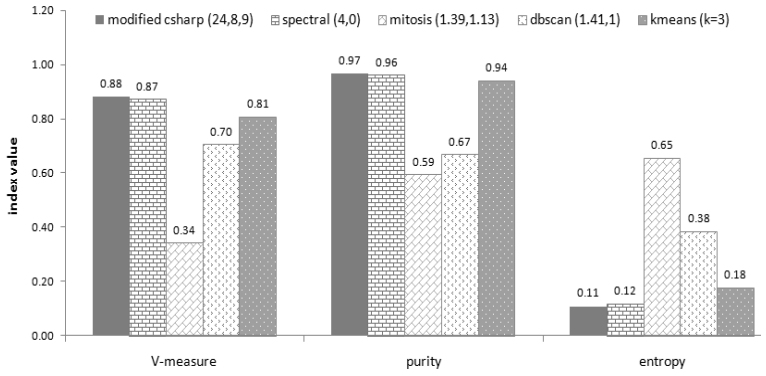


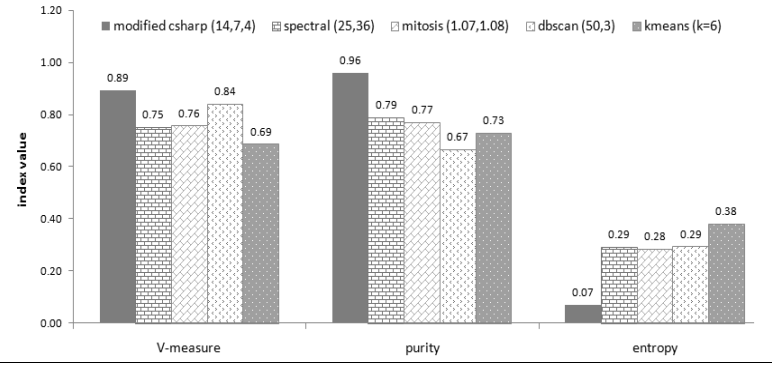
FIGURE 9: Time series clusters (mean ± standard deviation) of SCC data original classes.

- Pen Digits data set: CSHARP gave good solution and gave a very high Purity and low Entropy relative to the other solutions as shown in Figure 10c. DBScan and K-means combined several characters in one cluster. For instance, K-means combined 5 with (8, 9) and 1 with (2, 7), DBScan combined (1,2) and (5,9) and Mitosis combined 1 with (2, 7) and (5,9).
- Disease diagnosis data set: For breast cancer diagnosis, CSHARP obtained the maximum values for all the indexes at the parameters setting (K=50, T=8 and M=5). Figure 13a gives the corresponding validity indexes values.
- SPECT heart and Libras Movement : Modified CSHARP performed better for all indexes as shown in Figures 13b and c respectively.
- Ecoli and Yeast: The values of the indexes are given in Figures 13e and f, respectively showing the superiority of CSHARP over all other algorithms.

a



b



c

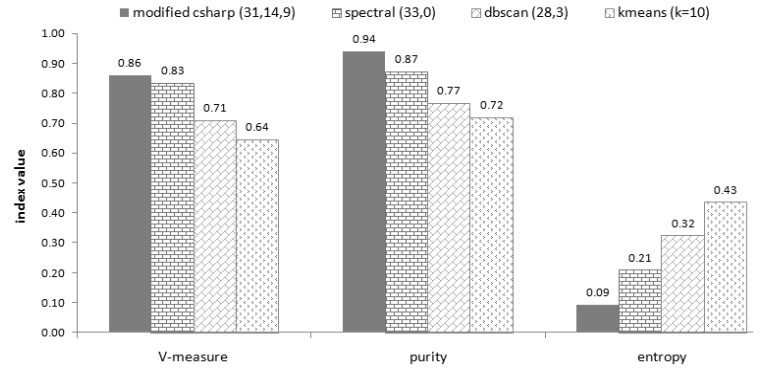


FIGURE 10: V-Measure, Purity and Entropy for: (a) Iris data set, and (b) SCC time series, and (c) pen digit recognition data set.

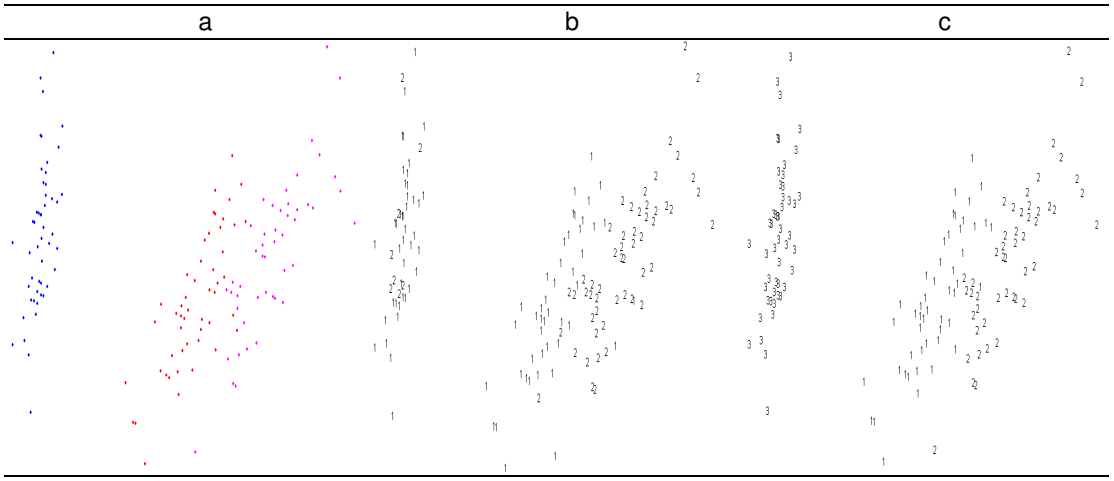


FIGURE 11: Plotting Iris data set using PCA for (a) reference solution, (b) Mitosis, and (c) Modified CSHARP and Spectral clustering.

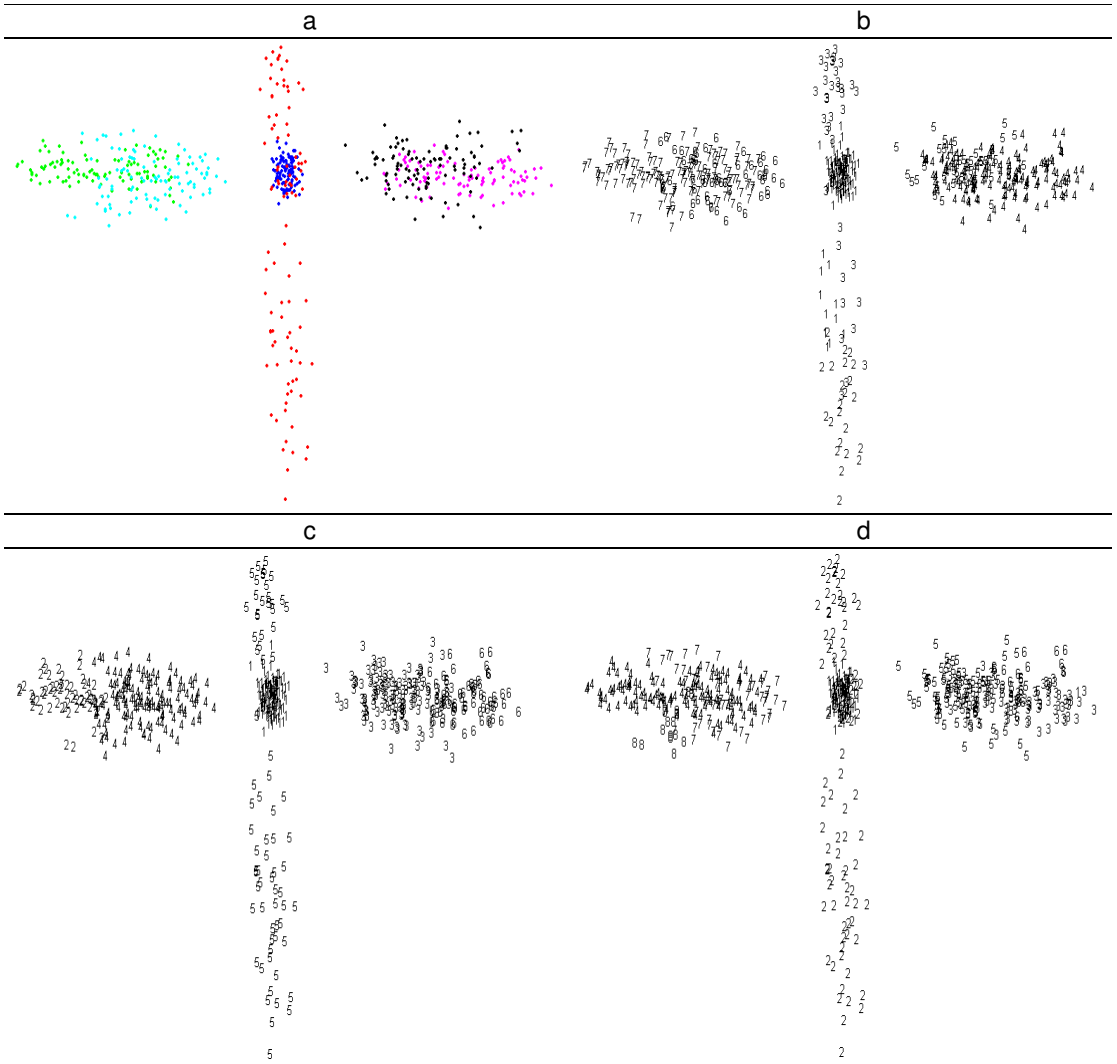
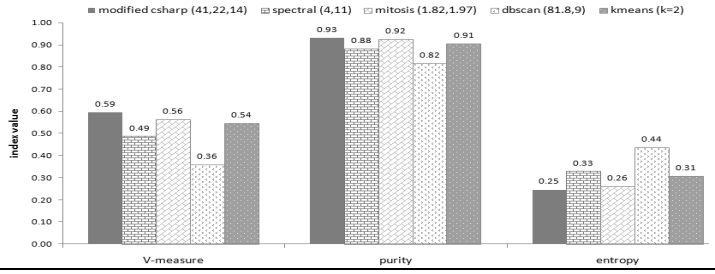
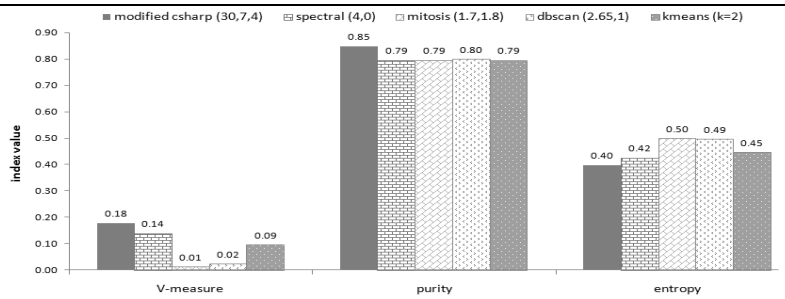


FIGURE 12: Plotting SCC time series data set using PCA for (a) reference solution, (b) Mitosis, (c) Spectral clustering, and (d) Modified CSHARP.

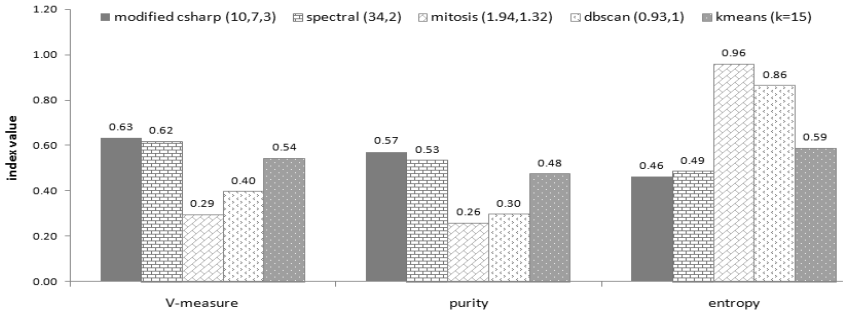
a



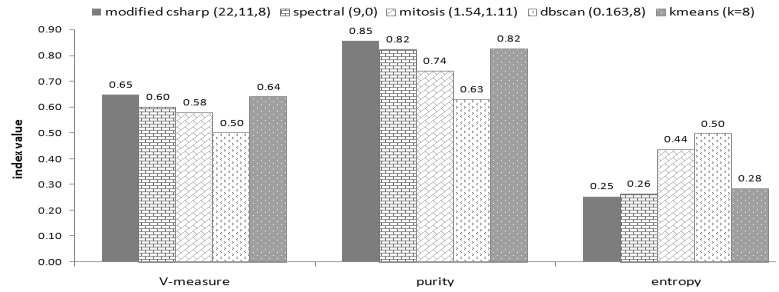
b



c



d



e

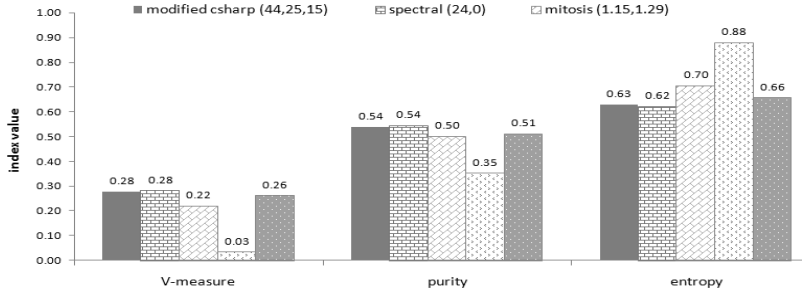


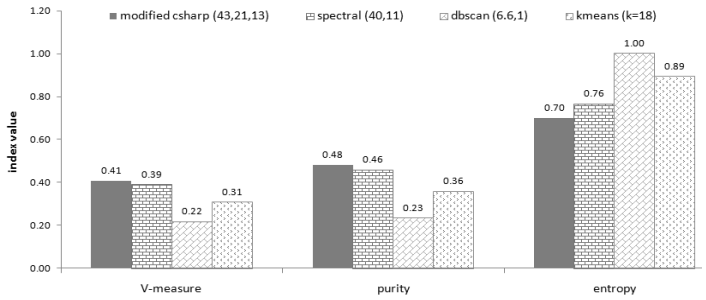
FIGURE 13: V-Measure, Purity and Entropy for: (a) Breast cancer Wisconsin diagnostic data set, (b) SPECT heart data set, and (c) Libras movement data set.(d) Ecoli data set, and (e) Yeast data set.

4.3.3. High Dimensional Data

To investigate the performance of the Modified CSHARP when applied to high dimensional data sets, two data sets were used from the 2003 Nips Feature extraction challenge [7]. Both CSHARP and spectral clustering obtained competitive results as shown in Figures 14a and b for Corel and Arcene data sets, respectively, with CSHARP performing slightly better for all indexes. Spectral clustering algorithm implemented by [2] uses two parameters, "K" for the number of K-nearest neighbors; used to generate a sparse symmetric distance matrix and σ value used in similarity function S,

$$\text{where } S_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{2*\sigma_i\sigma_j}\right).$$

a



b

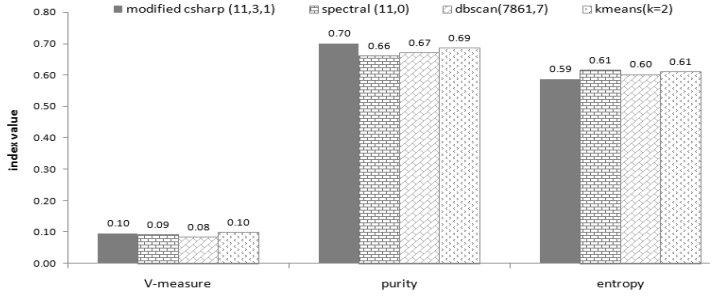


FIGURE 14: V-Measure, Purity and Entropy for: (a) Corel data set and (b) Arcene data set.

In this section, the efficiency of Modified CHARP is recorded when compared with well-known clustering algorithms such as K-means, DBScan, Chameleon, Mitosis, and Spectral Clustering. V-measure, Purity, and Entropy are used showing the superiority of CSHARP over the other tested algorithms for almost all used data sets .

Modified CSHARP succeeded to overcome the limitations that the other algorithms suffer from. It can deal with classes of different densities whereas DBScan cannot, deal with arbitrary shapes whereas K-means cannot, deal with arbitrary cluster's sizes where spectral clustering cannot, and deal with interlaced (overlapped) clusters where Mitosis cannot. Moreover, it can scale easily whereas Chameleon cannot. Therefore, it can be said that the proposed technique is less likely to merge clusters of different densities or different homogeneities as indicated by the obtained results. Next, the performance of Modified CSHARP relative to the other algorithms is investigated.

4.4 Speed Performance

The speed of Modified CSHARP has been compared to the speed of CSHARP, Chameleon and DBScan as shown in Figure. 15, using the DS5 data set, after dividing it into data subsets, each of size 1000 patterns. The subsets are added incrementally, and the speed of the algorithm is recorded for each increment. The time considered is the time required for running the core clustering algorithms, excluding the time for obtaining the similarity matrix

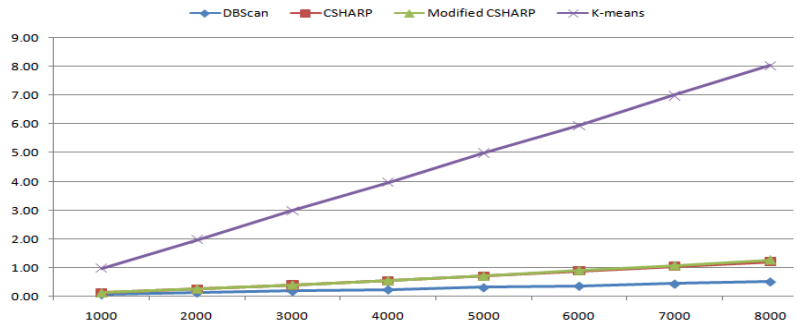
between the sample points. The time is measured in seconds. The average running times are computed for DBScan, K-means, and Modified CSHARP for 10 runs, with standard deviation listed in Table 3. Chameleon was tested only once for each increment. K-means was iterated 100 times for each experiment to provide better convergence.

The adopted algorithms as well as the proposed Modified CSHARP algorithm have been executed on a machine with the following configuration: 3.00 GHz processor, 1.0 GB RAM, and running Linux operating system (Ubuntu 10.04 LTS).

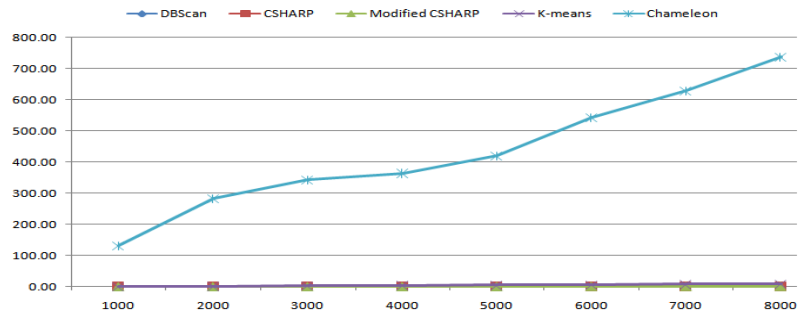
TABLE 3: Standard Deviation for 10 Runs on Modified CSHARP, DBScan, and K-means on Chameleon's DS5.

Data Size	1000	2000	3000	4000	5000	6000	7000	8000
K-means	0.020	0.008	0.017	0.015	0.021	0.018	0.023	0.043
DBScan	0.008	0.012	0.017	0.025	0.028	0.021	0.017	0.034
Modified CSHARP	0.013	0.013	0.014	0.016	0.016	0.019	0.025	0.037

a



b



c

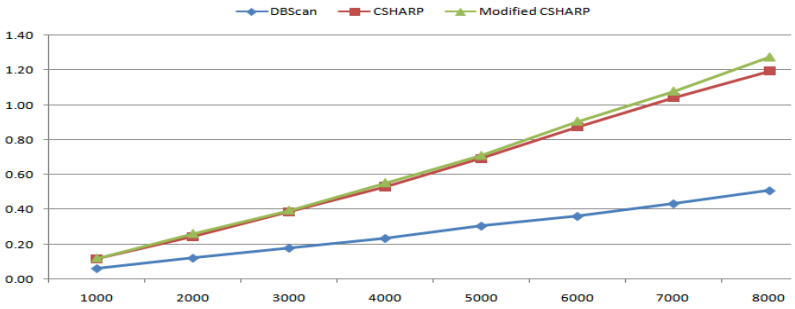


FIGURE. 15: Speed of CSHARP and Modified CSHARP using Chameleon's DS5 data set, compared to (a) DBScan and K-means (b) DBScan, Kmeans, and Chameleon (c) DBScan.

5. CONCLUSION

In this paper, a modified version of the novel shared nearest neighbors clustering algorithm, CSHARP[25] has been presented. Density and homogeneity are combined in a new homogeneity factor used to order the merging of blocks of points . It can find clusters of varying shapes, sizes and densities; even in the presence of noise and outliers and in high dimensional spaces as well. It is based on the idea of letting each data point vote for its K-nearest neighbors and adopting the points with the highest votes as clusters' seeds. Two clusters can be merged if their link strength is sufficient. Any data point not belonging to a cluster is considered as noise. The results of our experimental study on several data sets are encouraging. A wide range of possible parameters settings yield satisfactory solutions using the validation indexes adopted in [8], [9] and [10]. Modified CSHARP solutions have been found, in general, superior to those obtained by DBScan, K-means and Mitosis and competitive with spectral clustering algorithm adopted in[2].

More work is needed to introduce a procedure for parameters selection. Also, we intend to investigate the behavior of our clustering approach on other types of data. Moreover, we intend to parallelize our algorithm as its clustering propagation is inherently parallel, as has been shown in section 2.2. Finally, we have made the algorithm publicly available to other researchers at http://www.csharpclustering.com.

ACKNOWLEDGMENT

Authors would like to thank Dr. Noha A. Yousri for several useful discussions. Noha A. Yousri is with the Department of Computer and System Engineering, Faculty of Engineering, Alexandria University, Alexandria, Egypt.

REFERENCES

[1] Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. Communications of the ACM, 18(9):509–517.

[2] Chen, W.-Y., Song, Y., Bai, H., Lin, C.-J., and Chang, E. Y. (2011). Parallel spectral clustering in distributed systems. IEEE Trans. Pattern Anal. Mach. Intell, 33(3):568–586.

[3] Ding, C. H. Q. and He, X. (2004). K-nearest-neighbor consistency in data clustering: incorporating local information into global optimization. In Haddad, H., Omicini, A., Wainwright, R. L., and Liebrock, L. M., editors, SAC, pages 584–589. ACM.

[4] Ertoz, Steinbach, and Kumar (2003). Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. In SIAM International Conference on Data Mining, volume 3.

[5] Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In KDD, pages 226–231.

[6] Guttman, A. (1984). R-trees: A dynamic index structure for spatial searching. In ACM SIGMOD. Also published in/as: UCB, Elec.Res.Lab, Res.R. No.M83-64, 1983, with Stonebraker,M.

[7] Guyon, I., Gunn, S., Nikravesh, M., and Zadeh, L. (2006). Feature Extraction: Foundations and Applications. Springer Verlag.

[8] Halkidi, M., Batistakis, Y., and Vazirgiannis, M. (2002). Cluster validity methods: part I. SIGMOD Record (ACM Special Interest Group on Management of Data), 31(2):40–45.

[9] Hammouda, K. M. and Kamel, M. S. (2002). Phrase-based document similarity based on an index graph model. In ICDM, pages 203–210. IEEE Computer Society.

[10] Hubert, L. and Arabie, P. (1985). Comparing partitions. Journal of Classification, 2:193–218.

[11] Jain, A. K. and Dubes, R. C. (1988). Algorithms for Clustering Data. Prentice Hall, Englewood Cliffs.

[12] Jarvis, R. and Patrick, E. (1973). Clustering using a similarity measure based on shared near neighbors. IEEE Transactions on Computers, 22(11):1025–1034.

[13] Karypis, G., Han, E.-H. S., and NEWS, V. K. (1999). Chameleon: Hierarchical clustering using dynamic modeling. Computer, 32(8):68–75.

[14] Keogh, E. J., Xi, X., Wei, L., and Ratanamahatana, C. A. The ucr time series classification/clustering, homepage: (www.cs.ucr.edu/ eamonn/time_series_data/), 2006.

[15] Lee, J.-S. and Ólafsson, S. (2011). Data clustering by minimizing disconnectivity. Inf. Sci, 181(4):732–746.

[16] Murphy, P. M. and Aha, D. W. (1992). UCI repository of machine learning databases. Machine-readable data repository, University of California, Department of Information and Computer Science, Irvine, CA.

[17] Nadler, B. and Galun, M. (2006). Fundamental limitations of spectral clustering. In Schölkopf, B., Platt, J. C., and Hoffman, T., editors, NIPS, pages 1017–1024. MIT Press.

[18] Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods.

American Statistical Association Journal, 66(336):846–850.

[19] Rijsbergen, C. J. V. (1979). Information Retrieval, 2nd edition. Butterworths, London.

[20] von Luxburg, U. (2007). A tutorial on spectral clustering. Statistics and Computing, 17(4):395–416.

[21] Xu, R. and II, D. C. W. (2005). Survey of clustering algorithms. IEEE Transactions on Neural Networks, 16(3):645–678.

[22] Yousri, N. A., Kamel, M. S., and Ismail, M. A. (2009). A distance-relatedness dynamic model for clustering high dimensional data of arbitrary shapes and densities. Pattern Recognition, 42(7):1193–1209.

[23] Ying Zhao and George Karypis. 2001. Criterion functions for document clustering: Experiments and analysis. Technical Report TR 01.40, Department of Computer Science, University of Minnesota

[24] Andrew Rosenberg and Julia Hirschberg, 2007. V--Measure: a conditional entropy--based external cluster evaluation measure. EMNLP '07

[25] Mohamed A. Abbas Amin A. Shoukry, and Rasha F. Kashef, ICDM 2012. CSHARP: A Clustering Using Shared Reference Points Algorithm, International Conference on Data Mining, Penang, Malaysia, World Academy of Science, Engineering and Technology.