Volume 6, Issue 4, April 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Enhancement in DBSCAN using SIFT Technique

Harshit Tandon*, Manoj Kumar Gupta
Department of Computer Science Engineering, Sharda University, Greater Noida,
Uttar Pradesh, India

Abstract- DBSCAN (Density Based Spatial Clustering of Applications with Noise) is a pioneer density based algorithm. It can discover clusters of any arbitrary shape and size in databases containing even noise and outliers. DBSCAN however are known to have a number of problems such as: (a) it requires user's input to specify parameter values for executing the algorithm; (b) it is prone to dilemma in deciding meaningful clusters from datasets with varying densities; (c) and it incurs certain computational complexity. Many researchers attempted to enhance the basic DBSCAN algorithm, in order to overcome these drawbacks, such as VDBSCAN, FDBSCAN, DD_DBSCAN, and IDBSCAN. In this study, we survey over different variations of DBSCAN algorithms that were proposed so far. These variations are critically evaluated and their limitations are also listed. In this paper we propose an extended version of classic DBSCAN which integrates properties of SIFT technology to specify the boundary state data values by using HS-DBSCAN (Hierarchical SIFT DBSCAN)algorithm defined within the framework.

Kewords- DBSCAN, SIFT technique, clustering, fuzzy rules.

I. INTRODUCTION

Today we are living in an era of information so it becomes very essential for us to extract useful information from huge chunk of collected data. Data mining is one of the tasks in the Knowledge discovery in databases (KDD) which is a non-trivial process of identifying authentic, novel, practically applicable, and understandable patterns in large data sets to produce a specific enumeration of patterns over the data [1]. In general terms, data mining is an analytic process used to extract information from huge chunk of data.Clustering is a process of classifying data into natural groups without any prior knowledge of data distribution in underlying set, maintaining homogeneity within a cluster and heterogeneity between clusters. As it has a very wide range of application, the task of generating clusters has been studied under various disciplines, including data mining.HIERARCHICAL ALGORITHM: Hierarchical clustering forms clusters based on the concept of "Dendrogram" (clusters formed based on tree data structure). All the objects in the data sets are present at the root node of the dendrogram tree as a single cluster. To create dendrogram tree we can either use Agglomerative (bottom up or merging) approach where we merge the similar cluster as we move upwards in the hierarchy[2(18)] or we can use Divisive (top down or dividing) approach where we split the clusters repeatedly while moving downwards in the hierarchy. Some examples of hierarchical clustering techniques are BIRCH, CURE [10] and ROCK [11].

In partitional clustering, rather than creating clusters in several steps, partitioning clustering also called as nonhierarchical clustering creates them in single step. We get only one partition of the dataset as the output of the algorithms for partitioning clustering instead of getting a clustering structure as an output. Nearest neighbor and squared error are some of the algorithms for partition based clustering just like the K-mean and PAM algorithms. The use of an iterative process for creating clusters gives an advantage to partition based algorithms although it has a limitation that only the user can decide about the total number of clusters that can be made.

Density based clustering techniques give the ability of producing arbitrary shaped clusters unlike the partitioning and hierarchical methods where clusters can be of spherical shape only. DBSCAN and OPTICS are examples of density based clustering techniques. These algorithms are very useful in discovering consistent clusters of different shapes and sizes from large chunk of data and also facilitate detection of noisy outliers. DBSCAN algorithm has the time complexity of $O(n^2)$ where n represents to the size of datasets which makes it unsuitable for data mining applications with very large datasets. It requires two global inputs, minimum objects (μ) and radius (Ɛ) which are specified by user. DBSCAN [2] also seize to detect the density varied clusters properly due to global input minimum objects.

Grid based clustering is an another type of clustering technique which arranges object space to form a grid like structure for which it quantizes object space into cells of finite number. The processing time of this approach is very high, giving it an advantage over others. Its speed of processing is dependent on the total number of cells present in every dimension in the quantized space.

Model Based Clustering Algorithms make assumptions of a model for every cluster and obtains the best suitable data for the given model.FUZZY ALGORITHMS.Fuzzy algorithms recommend soft clustering schema as they assume that the objects can never be grouped in clusters of fixed size. FCM (Fuzzy C-Means) is an example of fuzzy clustering algorithms [12].

This paper is structured in four sections. The first section gives a brief introduction about the topic. In the next section we have done the background survey. Implementation has been discussed in the third section and finally the fourth section contains the obtained results.

II. LITERATURE SURVEY

B. Borah [4] proposes IDBSCAN, an improved version of DBSCAN which introduces a sampling technique to handle two limitations of original DBSCAN. Experiments show that sampling based IDBSCAN is more profitable than DBSCAN when it comes to minimizing input-output (I/O) cost and memory needed for clustering without compromising with the quality level of the cluster. IDBSCAN improved the I/O cost, however it needs users to manually define the value of the threshold parameters.

El Sonbaty [6] proposes an enhancement in DBSCAN which uses following procedure for generating potential clustering result from large volume of datasets. During the pre-processing stage before analyzing the dataset, it is partitioned using CLARANS. This partitioning minimizes the effort required for searching the core object as this approach limits the search of core object to the single partitioned region instead of searching it from the whole dataset.

X. P. Yu [24] proposed a density based clustering algorithm known as KNNDBSCAN (K-nearest neighbors DBSCAN). The quality of the clusters obtained either by applying DBSCAN algorithm or any of its extensions such as VDBSCAN [14], EDBSCAN [16] etc basically depends on the accuracy of the values of the input parameters i.e MinPts and Eps. The main problem accompanying majority of the density based algorithm is determining the global values of these input parameters. Unlike the DBSCAN which requires two input parameters values, KNNDBSCAN needs value for a single input parameter called "K". This parameter is capable in determining values of these input parameters unsupervisedly. The value of "K" does not affect the resulting clusters. To determine the arbitrary shaped clusters, KNNDBSCAN merges two approaches which are K-nearest neighbors and DBSCAN. This collective approach functions as follow. First, we determine width and related neighbors for every data point window. Then we partition the entire dataset into Fuzzy clusters (FCs). This partitioning is done by means of KNN based on KDE-based rules. This will improve the performance as the number of scans gets reduced. Now, for each FCs we calculate density threshold ε and *MinPts* which is determined on the basis of the Entropy theory. At last, every local threshold values are mapped onto the global value of ε and simultaneously every FCs is clustered in parallel independently. This increases the speed of the clustering process and also reduces the memory overhead by keeping the only FC that is to be clustered instead of the entire dataset. The main success points comprise of: (1) automating the calculation of density threshold; (2) Unlike the DBSCAN, this approach clusters the datasets in parallel which makes it capable of carrying out the clustering process at higher speed; (3) since only single partition of dataset is taken into account for making clusters so it reduces the memory overhead, as whole dataset is not needed to be stored in memory while carrying out the clustering process.

GRIDBSCAN [19] is one more important variation of DBSCAN. They tackled the issues associated with majority of the density-based clustering algorithms. The main problem is that they are not efficient in carrying out clustering process accurately in the presence of clusters having different densities. [15]proposes a three level clustering mechanism in this literature so as to provide a solution to this problem. In the first level, it provides suitable grids such that density in each grid is similar. In the next level, it merges those cells that have similar densities. At this level, the suitable value of ε and *MinPts* is also recognized in each grid. In the final step, these identified parameters values are used while applying the DBSCAN algorithm so as to find the required final number of clusters. Although GRIDBSCAN is better than DBSCAN in terms of accuracy but GRIDBSCAN may be hectic in terms of computational complexity when employed for large spatial data [19].To overcome this problem in this literature an extension of DBSCAN is proposeduring fuzzy logic to apply a soft constraint and decrease the rigidness of these parameters. In this literature fuzzy logic is applied only on input parameter MinPts and experimental results show that it increased the efficiency of this algorithm. We can try to apply this soft constraint either on second input ε or on both parameters simultaneously to identify that if it increases the efficiency of this enhanced extension any further any further [8]. Li Ma proposes grid based extension of DBSCAN algorithm based on Map Reduce to overcome the problem of memory and I/O overhead. It replaces all the values in the grid with its center value so that the computational overhead gets reduced with the decrease in amount of data processed by algorithm. Grid based DBSCAN is combined with Map reduce to make it suitable even for large data processing.

III. METHODOLOGY

The classic DBSCAN uses two input parameters which are calculated very rigidly. We use fuzzy logic to generate the fuzzy rules in order to relax the dependency upon the accurate calculation of the input parameter values. This reduce the rigidity of dependence on the calculation of input parameters. This dependence on input parameters is further improved by integrating fundamental properties of SIFT technique. This allows us to use some other attributes to reduce the redundancy and calculation overhead while putting data values in different clusters.

A. Pseudocode

Algorithm. ***Hierarchical SHIFT DBSCAN***($MinPts_{Min}$, $MinPts_{Max}$)
Step 1: Start
Step 2: Require Dataset density threshold for Min_{Pts} and Max_{pts}
Step 3: Require Initial cluster (c)
Step 4: Assign dist = 0,
Step 5" farthest = 0.

Step 6: For each data point x perform the following
Step 7: Initialize sum =0
Step 8: for Input: D = {t1, t2, t3 … tn} // Set of elements
Step 9: MinPts //Number of points in cluster
Ɛ // Maximum distance for density measure
//Apply the ***Hierarchical SHIFT DBSCAN*** to specify the boundary state data values in DBSCAN
Step 10: K = {K1, K2, K3 …, Kk} // Set of clusters
Step 11: Method: k=0; // initially there are no cluster
Step 12: For each of the cluster centers c assigned so far perform the
Step 13: Find the distance between c and x
Step 14: sum += d
Steps 15: If sum > dist then assign dist = sum and farthest = x
Steps 16: for i = 1 to n do
Steps 17: if ti is not in a cluster, then
Steps 18: X = {tj | tj is density-reachable from ti};
Steps 19: if X is a valid cluster, then
Steps 20: k = k+1;
Steps 21: Kk = X;
Steps 22: variable farthest represents the farthest data point from all the centers assigned so far

Proposed Model: The flowchart describes the process of integrating fundamental properties of SIFT with the DBSCAN. Firstly, we cluster the data set by using classic DBSCAN clustering technique and then calculate centroid values for every cluster. Now, we group data points based on classes generated based on SIFT properties and then we calculate their centroid values as well. Now, we reassign every data point to a cluster after comparing centroid values from both steps. The new generated clusters are more efficient in terms of accuracy and time taken.

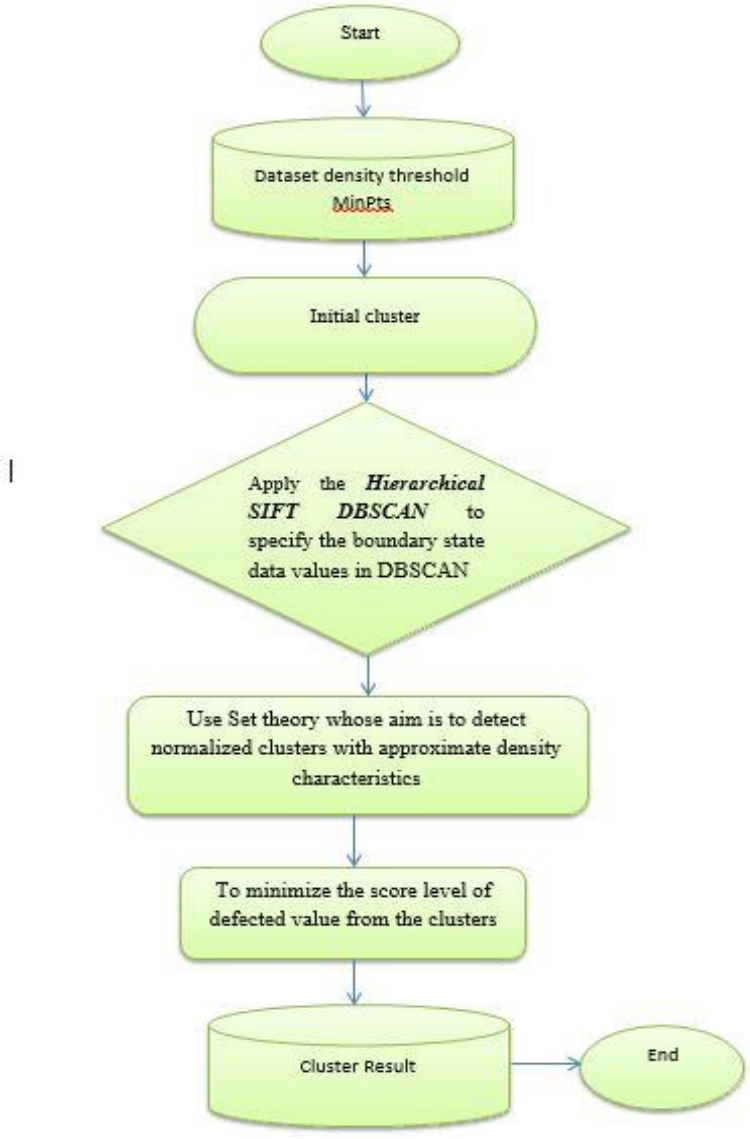

Figure 1: Flowchart of the proposed model

B. Implementation

The steps for….

- A hierarchical clustering method is a procedure that transforms a dissimilarity matrix into a sequence of nested partitions. A dissimilarity matrix D is a square and symmetric matrix that contains all the pairwise dissimilarities between the samples that should be clustered. We have to show the estimated lowest possible upper bound in of recovery rates treated by applying it for the recovery of old clusters of a fairly complex set of simulated data. The obtained recovery rate is close to the estimated best possible upper bound of recovery rates for the dataset.
- The most frequently used inter-cluster merging techniques are single linkage (clusters are merged based on the shortest distance between objects in the two clusters), complete linkage (merging is based on the largest distance between objects) and average linkage (based on the average distance between objects).
- By using HS-DBSCAN technique we can go through minimal number of cluster set for pattern base mining with same density value that can be match by using data hierarchy.
- Our algorithm utilizes the average linkage merging approach, because it takes into account information from all objects in a cluster.
- Density based hierarchy would classify with different object value for dataset behind this scalability for our dataset get consider easily.

IV. RESULT AND ANALYSIS

A. Dataset Used

We have used the records of 650 patients, were randomly assigned to one of the following 10 treatment groups (65 subjects per group)
Drug X (10 mg)
Statin (10, 20, 40 or 80 mg)
Drug X (10 mg) + Statin (10, 20, 40 or 80 mg)
Lipid profile (LDL cholesterol, HDL CHolesterol and Triglycerides) was measured at baseline (BL) and at 12 weeks (after the start of treatment).In addition to the lipid profile, patient age, gender and Cardiac Heart Disease (CHD) risk category was also logged at baseline.
The columns in the data are as follows:
ID - Patient ID
Group - Treatment group, Dose_A - Dosage of Statin (mg), Dose_X - Dosage of Drug X (mg), Age - Patient Age, Gender - Patient Gender, Risk - Patient CHD risk category (1 is high risk, and 3 is low risk), LDL_BL - HDL_BL & TC_BL - Lipid levels at baseline, LDL_12 wks , HDL_12wks & TC_12wks - Lipid levels after treatment
We will import the data into a dataset array that affords better data management and organization.

B. Result

In figure 2 primary efficancy endpoint is the level of LDL cholesterol at start of treatment, which is compared with LDL C levels at baseline to LDL C levels after treatment. The mean LDL C level at baseline is around 4.2 and mean level after treatment is 2.5. So, at least for the data pooled across all the treatment groups, it seems that the treatment causes lowering of the LDL cholesterol levels.This is used to find a suitable value which can be used as a threshold value for carrying out the computation part.

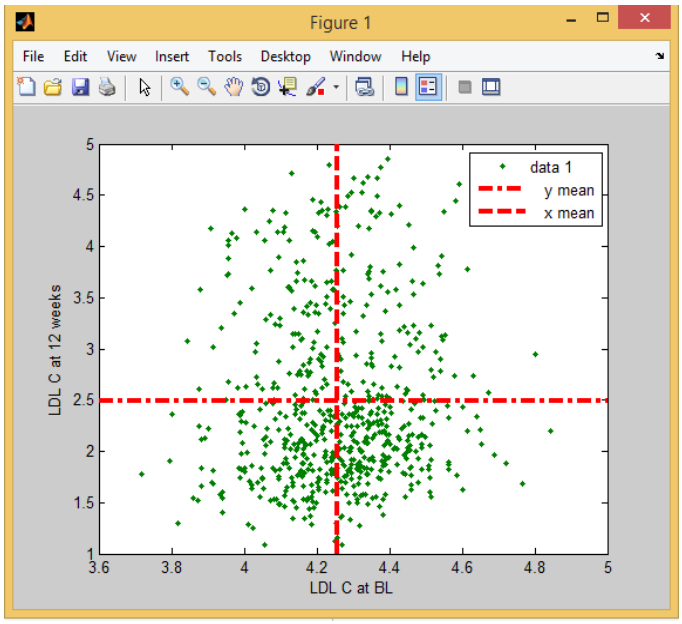

Figure 2: LDL C at 12 weeks

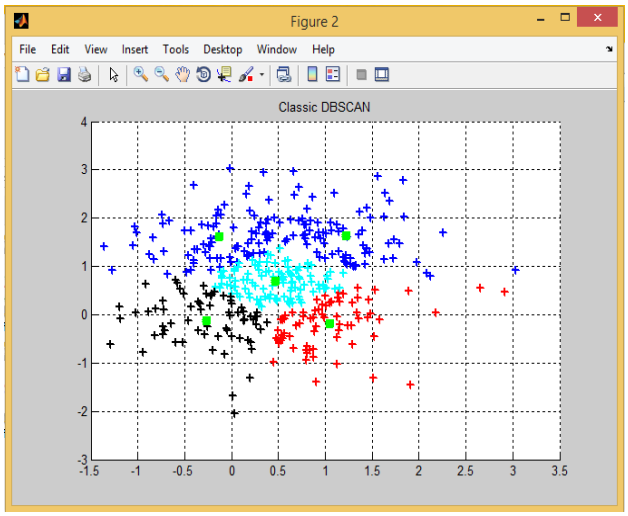

Figure 3: classic DBSCAN

In this figure 3 we applied classic DBSCAN on the data set and calculated the centroid values. The clusters formed are four in number but there are five centroid values. This shows that there is some misplacement of some data points while putting them in a cluster based on similar properties and clustering process needs to be refined to achieve more accuracy in placement of data points.

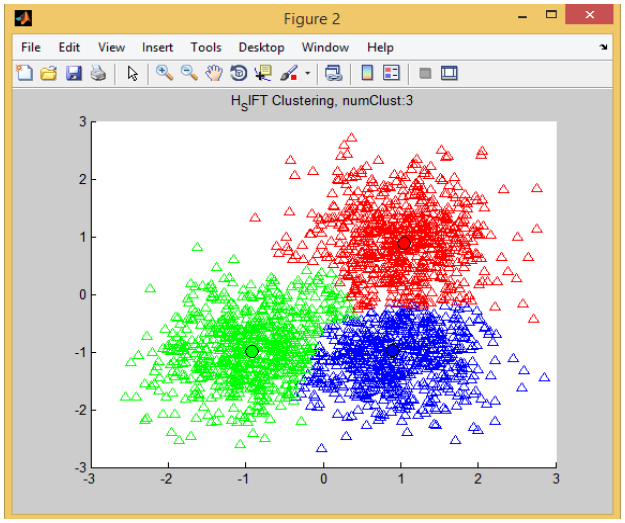

Figure 4: SIFT clustering

As shown in the above figure 4, result shows the number of cluster groups are 3 with its centroid. Using SIFT clustering we get the better result as compare to cure DBSCAN due to reason of SIFT work as a hierarchical that contain features of male and female Drug. Using SIFT we can take into account set of unique attributes to make clustering process more specific from the start. This helps to differentiate two very similar data points and help in reducing the calculation overhead.

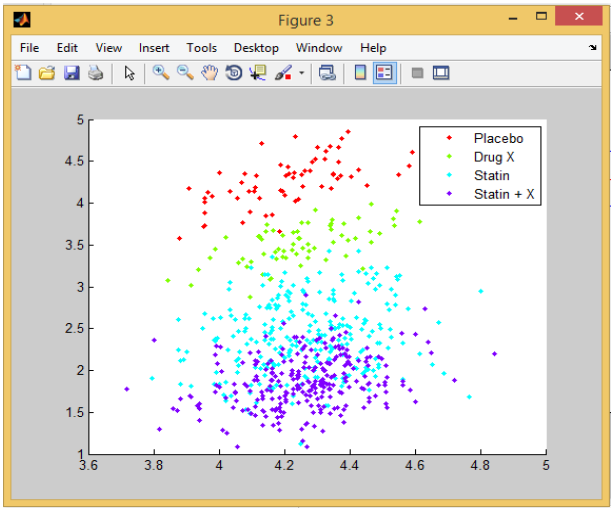

Figure 5: Visualize effect of treatment and statin dose on percentage LDL reduction

First, we will extract percent change in LDL C level for the Statin and the Statin + X groups only. We will test the null hypothesis that the percent change in LDL C level for the "Statin + X" groups is greater than that in the "Statin + X" using pooled data. We use a 2 sample t-test to test this hypothesis.We performed a detailed hypothesis to see if statin + X group (grp2) is better than the Statin group (grp1). We test against the alternative that the mean LDL change of grp1 (Statin only) is less than mean LDL change of grp2 (Statin + X). The null hypothesis is rejected ($p < 0.01$), implying that grp1 mean is less that grp2 mean, i.e. the Statin group is less effective at lowering LDL C levels than the Statin + X group.The pooled analysis shows that coadministering drug X with statin is more effective than statin mono therapy.Our analysis so far was done on pooled data. We analyzed the effect of treatment (statin alone (X = 0) vs. statin + 10 mg X) on the LDL C levels. We ignored levels of statin dose within each treatment group. Next, we wouldbe performing a 2-way ANOVA (analysis of variance) to simultaneously understand the effect of both factors - statin dose (4 levels – 10, 20, 40, 80 mg) and Treatment (2 level - statin only or Statin + 10 mg X) - on the percentage change of LDL C levels.

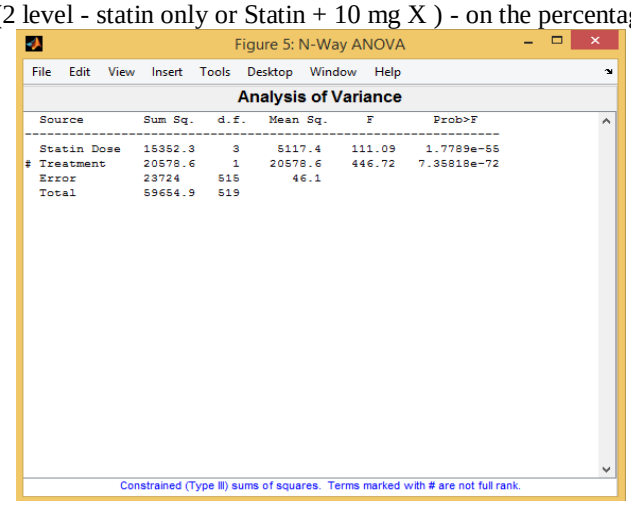

Figure 6: Effect of Statin Dose on incremental increase in percentage LDL reduction

The ANOVA results indicate that statin dose is a significant factor, but it doesn't compare means across individual dose-treatment level combination. We look at the individual cell means.

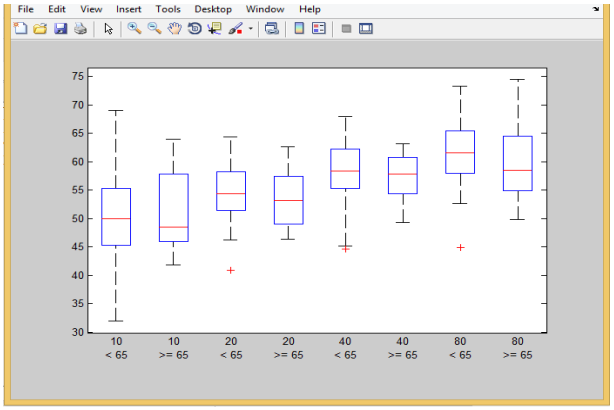

Figure 7: Consistency of effect across subgroups, age and gender

Finally, we will plot a graph using RMSE method to ensure that the efficacy of the Statin + X treatment at various statin doses is consistent across gender and age subgroups. We will perform this check for only the Statin + X treatment group. We will convert the continuous age variable into a categorical variable, with 2 categories: Age < 65 and Age >= 65

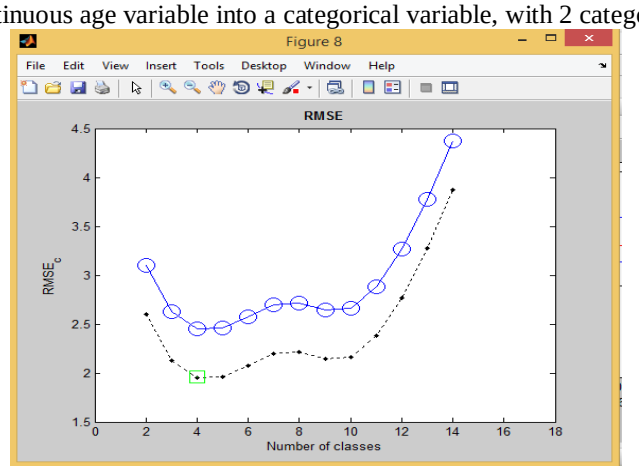

Figure 8: Comparison of existing and proposed method w.r.t RMSE and speed

The above figure 8, shows that the performance of our proposed work which isHierarchical SIFT is better as compare to existing method. Because of RMSE (Root mean square value) is 2.9 with the number of class of 2 and existing method has approx. 4.2 RMSE value. So finally when RMSE is less than on number of class then it is better for clustering. Thus the H-SIFT are performing better due to less time.

V. CONCLUSION

DB SIFT algorithm capable to reduce the redundancy which are found in the result, generated by DBSCAN. This technique helps in shifting the different data-points to right clusters and improve the accuracy of generated result.

We have used the medical dataset to check the accuracy of our proposed algorithm at boundary conditions in generating the final clusters. Further results is compared with the DBSCAN algorithm to show the improved time complexity that is achieved by reducing the redundancies.Our work can be extended by implementing proposed algorithm on distributed parallel computational environment over different nodes.

REFERENCE

[1] Ankerst, M., Breunig, M.M., Kriegel, H.P. and Sander, Jh., 1999, June. OPTICS: ordering points to identify the clustering structure. In ACM Sigmod Record (Vol. 28, No. 2, pp. 49-60). ACM.

[2] Birant, D. and Kut, A., 2007. ST-DBSCAN: An algorithm for clustering spatial–temporal data. Data & Knowledge Engineering, 60(1), pp.208-221.

[3] Borah, B. and Bhattacharyya, D.K., 2007, February. A clustering technique using density difference. In Signal Processing, Communications and Networking, 2007. ICSCN'07. International Conference on (pp. 585-588). IEEE.

[4] Borah, B. and Bhattacharyya, D.K., 2004. An improved sampling-based DBSCAN for large spatial databases. In Intelligent Sensing and Information Processing, 2004. Proceedings of International Conference on (pp. 92-96). IEEE.

[5] Elbatta, M.N., 2012. An Improvement for DBSCAN Algorithm for Best Results in Varied Densities (Doctoral dissertation, The Islamic University-Gaza Palestine).

[6] El-Sonbaty, Y., Ismail, M.A. and Farouk, M., 2004, November. An efficient density based clustering algorithm for large databases. In Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on (pp. 673-677). IEEE.

[7] Ester, M., Kriegel, H.P., Sander, J. and Xu, X., 1996, August. A density-based algorithm for discovering clusters in large spatial databases with noise. In Kdd (Vol. 96, No. 34, pp. 226-231).

[8] Fayyad, U.M., Piatetsky-Shapiro, G. and Smyth, P., 1996, August. Knowledge Discovery and Data Mining: Towards a Unifying Framework. In KDD (Vol. 96, pp. 82-88).

[9] Gaonkar, M.N. and Sawant, K., 2013. AutoEpsDBSCAN: DBSCAN with Eps automatic for large dataset. Published in International Journal on Advanced Computer Theory and Engineering (IJACTE), ISSN (Print), 2, pp.2319-2526.

[10] Guha, S., Rastogi, R. and Shim, K., 1998, June. CURE: an efficient clustering algorithm for large databases. In ACM SIGMOD Record (Vol. 27, No. 2, pp. 73-84). ACM.

[11] Guha, S., Rastogi, R. and Shim, K., 1999, March. ROCK: A robust clustering algorithm for categorical attributes. In Data Engineering, 1999. Proceedings, 15th International Conference on (pp. 512-521). IEEE.

[12] Huang, J., Sun, H., Song, Q., Deng, H. and Han, J., 2013. Revealing density-based clustering structure from the core-connected tree of a network. Knowledge and Data Engineering, IEEE Transactions on, 25(8), pp.1876-1889.

[13] Liu, B., 2006, August. A fast density-based clustering algorithm for large databases. In Machine Learning and Cybernetics, 2006 International Conference on (pp. 996-1000). IEEE.

[14] Liu, P., Zhou, D. and Wu, N., 2007, June. VDBSCAN: varied density based spatial clustering of applications with noise. In Service Systems and Service Management, 2007 International Conference on (pp. 1-4). IEEE.

[15] Mahran, S. and Mahar, K., 2008, July. Using grid for accelerating density-based clustering. In Computer and Information Technology, 2008. CIT 2008. 8th IEEE International Conference on (pp. 35-40). IEEE.

[16] Ram, A., Sharma, A., Jalal, A.S., Agrawal, A. and Singh, R., 2009, March. An enhanced density based spatial clustering of applications with noise. In Advance Computing Conference, 2009. IACC 2009. IEEE International (pp. 1475-1478). IEEE.

[17] Ram, A., Jalal, S., Jalal, A.S. and Kumar, M., 2010. A density based algorithm for discovering density varied clusters in large spatial databases. International Journal of Computer Applications, 3(6), pp.1-4.

[18] Tang, X.Q. and Zhu, P., 2013. Hierarchical clustering problems and analysis of fuzzy proximity relation on granular space. Fuzzy Systems, IEEE Transactions on, 21(5), pp.814-824.

[19] Uncu, O., Gruver, W., Kotak, D.B., Sabaz, D., Alibhai, Z. and Ng, C., 2006, October. GRIDBSCAN: Grid density-based spatial clustering of applications with noise. In Systems, Man and Cybernetics, 2006. SMC'06. IEEE International Conference on (Vol. 4, pp. 2976-2981). IEEE.

[20] Viswanath, P. and Pinkesh, R., 2006, August. l-dbscan: A fast hybrid density based clustering method. In Pattern Recognition, 2006. ICPR 2006. 18th International Conference on (Vol. 1, pp. 912-915). IEEE.

[21] Wu, O., Hu, W., Maybank, S.J., Zhu, M. and Li, B., 2012. Efficient clustering aggregation based on data fragments. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 42(3), pp.913-926.

[22] iaoyun, C., Yufang, M., Yan, Z. and Ping, W., 2008, October. GMDBSCAN: multi-density DBSCAN cluster based on grid. In e-Business Engineering, 2008. ICEBE'08. IEEE International Conference on (pp. 780-783). IEEE.

[23] Xu, R. and Wunsch, D., 2005. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, *16*(3), pp.645-678.

