Clustering: Applied to Data Structuring and Retrieval

Ogechukwu N. Iloanusi
Department of Electronic Engineering
University of Nigeria, Nsukka
Enugu State, Nigeria

Charles C. Osuagwu
Department of Electronic Engineering
University of Nigeria, Nsukka
Enugu State, Nigeria

Abstract—**Clustering is a very useful scheme for data structuring and retrieval behuhcause it can handle large volumes of multi-dimensional data and employs a very fast algorithm. Other forms of data structuring techniques include hashing and binary tree structures. However, clustering has the advantage of employing little computational storage requirements and a fast speed algorithm. In this paper, clustering, k-means clustering and the approaches to effective clustering are extensively discussed. Clustering was employed as a data grouping and retrieval strategy in the filtering of fingerprints in the Fingerprint Verification Competition 2000 database 4(a). An average penetration of 7.41% obtained from the experiment shows clearly that the clustering scheme is an effective retrieval strategy for the filtering of fingerprints.**

Keywords-component; Clustering; k-means; data retrieval; indexing.

I. Introduction

A collection of datasets may be too large to handle and work on hence may be better grouped according to some data structure. Large datasets are encountered in filing systems in digital libraries, access to and caching of data in databases and search engines. Given the high volume of data there is need for fast access and retrieval of required or relevant data. Several of the existing data structures are hashing [1, 2, 3, 4, 5, 6], search trees [7, 8], and clustering [9]. Hashing is a technique that utilizes a hash function to convert large values into hash values and maps similar large values to the same hash values or keys in a hash table. Clustering is however a useful and efficient data structuring technique because it can handle datasets that are very large and at the same time n-dimensional (more than 2 dimensions) and similar datasets are assigned to the same clusters [9]. A 2D or 3D point can be imagined and illustrated however it will be difficult to imagine or illustrate a 9-dimensional data. When datasets are clustered, the clusters can be used rather than the individual datasets.

Clustering is a process of organizing a collection of data into groups whose members are similar in some way [9, 10, 11, 12] According to Jain et al. [13] "Cluster analysis is the organization of a collection of patterns (usually represented as a vector of measurements, or a point in a multidimensional space) into clusters based on similarity". Similarity is determined using a distance measure and objects are assigned and belong to the same cluster if they are similar according to some defined distance measure. Cluster analysis differs from classification because in clustering the data are not labeled and hence are naturally partitioned by the clustering algorithm whereas in classification the data are labeled and partitioned according to their labels. The former is hence an unsupervised mode of data structuring while the later is supervised [13]. Jain [14] identifies three main reasons while data clustering is used; to understand the underlying structure of the data; to determine degree of similarity amongst the data in their natural groupings and to compress data by summarizing the data by cluster groups.

Clustering has a vast application in the life sciences, physical and social sciences and especially in the disciplines of Engineering and Computer Science. Clustering is used for pattern analysis, recognition and classification, data mining and decision making in areas such as document retrieval, image processing and statistical analysis and modeling [13]. Documents may be clustered for fast information access [15] or retrieval [16]. Clustering is used in image processing to segment images [17] as well as in marketing, biology, psychiatry, geology, geography and archeology [13]. Figure 1 shows a general data clustering illustration. The data are grouped in clusters. Each cluster has a collection of data that are similar.

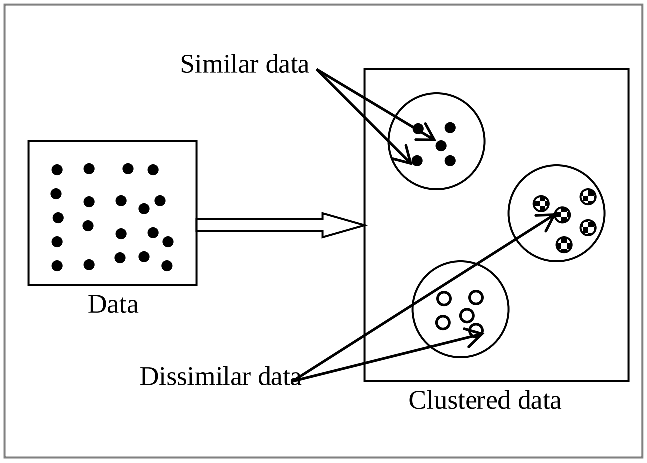


Figure 1. Data Clustering

A cluster is a group of similar datasets represented by an n-dimensional value given by the cluster centroid. Clusters may also be defined as "high density regions separated by low density regions in the feature space" [13].

Every cluster is assumed to have a centroid, which is the arithmetic mean of all data in that cluster. The mean is what is common to data assigned to a cluster and creation of clusters builds from the arithmetic mean. A similarity measure is used for the assignment of patterns or features to clusters.

II. Cluster Similarity Measures

Similarity is fundamental to the definition of a cluster hence a measure for the similarity otherwise known as the distance measure is essential. The dissimilarity or similarity between points in the feature space is commonly calculated in cluster analysis [13]. Some of the distance measures used are:

- Euclidean distance
- Manhattan distance
- Chebyshev distance
- Hamming distance

The distance metric is used for computing the distance between two points and cluster centers. For the distance measures explained in the following sections, two points, a and b, are defined in an n-dimensional space as:

$$\mathbf{a = (w_0, x_0, y_0 \ldots z_0)\ coordinates} \qquad \mathbf{(1)}$$

$$\mathbf{b = (w_1, x_1, y_1 \ldots z_1)\ coordinates} \qquad \mathbf{(2)}$$

A. *Euclidean distance*

Euclidean distance is the distance between two points, a and b, as the crow flies in an n-dimensional space.

$$\boldsymbol{D(a,b) = \sqrt{((w_0 - w_1)^2 + \cdots + (z_0 - z_1)^2)}} \qquad \mathbf{(3)}$$

$$= \sqrt{\boldsymbol{\sum_{i=1}^{n}(a_i - b_i)^2}} \qquad (4)$$

where n is the number of dimensions. The Euclidean distance is the most commonly used metric because it is appealing to use in an n-dimensional space and it works well with isolated clusters [13].

B. *Manhattan distance*

In the Manhattan distance, the distance between two points is the absolute difference of their coordinates.

$$\boldsymbol{D(a,b) = (w_0 - w_1) + \cdots + (z_0 - z_1)} \qquad (5)$$

The difference between the Euclidean distance and the Manhattan distance is that the Euclidean is a squared distance while the Manhattan is not squared.

C. *Chebyshev distance*

In the Chebyshev distance metric the distance between two points is the greatest of their differences along any coordinate dimension [18]. This distance is named after Pafnuty Chebyshev.

$$\boldsymbol{D(a,b) = \max(a_i - b_i)} \qquad (6)$$

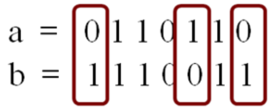
This is also known as the chessboard distance. In the chessboard the length of side of a chess square may be assumed as one unit. In this case the minimum number of moves needed by a king to go from one chess square to another equals the Chebyshev distance between the centers of the squares.

D. *Hamming distance*

The Hamming distance is a way of determining the similarity of two strings of digits of equal lengths by measuring the number of substitutions required to change a string into another. It is the number of positions at which corresponding digits in the two strings are different [19].

Given two strings a and b where

a = 0110110 and b = 1110011, the difference between the two strings a and b, D(a,b), where



D(a,b) = 3, as the corresponding digits differ in three places.

III. CLASSIFICATION OF CLUSTERING ALGORITHMS

Clustering algorithms may be classified as:

- Exclusive clustering
- Overlapping clustering
- Hierarchical clustering

A. *Exclusive clustering*

In exclusive clustering, data that belongs to a particular cluster cannot belong to another cluster. An example is K-means clustering.

B. *Overlapping clustering*

Data may belong to two or more clusters. Example of this in fuzzy-c-means clustering.

C. *Hierarchical clustering*

In this case clusters are represented in tree from. Two close clusters are derived from the top-level cluster. The hierarchy is built by individual elements progressively merging into bigger clusters.

Figure 2 shows the types of data clustering algorithms.

Jain [13] classifies clustering algorithms as hierarchical and partitional. In hierarchical clustering each cluster arises from and depends on the parent cluster. A typical partitional clustering algorithm is the K-means algorithm.

IV. Cluster Similarity Measures

K-means clustering algorithm was first proposed over 50 years ago [14] and is commonly preferred to other clustering algorithms because of its ease of implementation and efficiency in cluster analysis.

K-means clustering is a type of cluster analysis that partitions n observations into k disjoint clusters, k<<n, such that the number of clusters are much less than the number of observations [18, 20]. The k-means algorithm partitions n observations $\{O_i \mid i=1, 2 \ldots n\}$ into k number of clusters, $\{C_j \mid j=1,2 \ldots k\}$, as follows

$$\boldsymbol{\{C_j \in O_i\}\ if\ \{argmin \sum_{j=1}^{k} \sum \left\| O_i - C_j \right\|^2\}} \qquad (7)$$

This is illustrated in Figure 3.

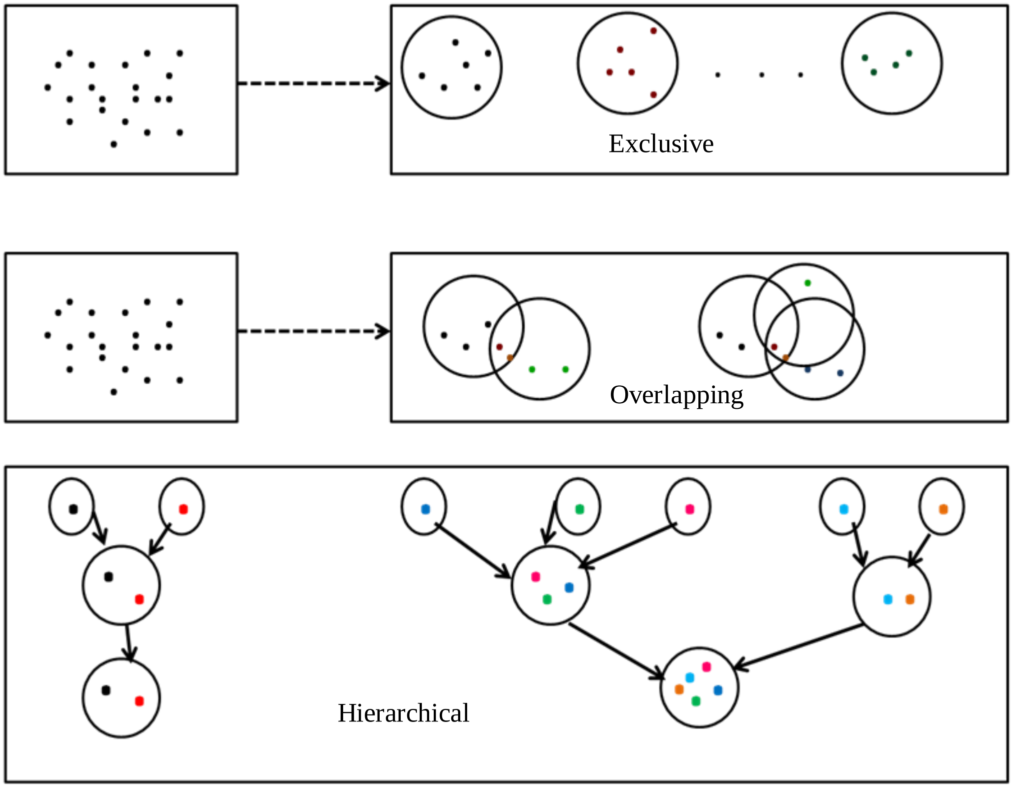


Figure 2. Types of Data Clustering Algorithms

Any of the distance metrics which include the Euclidean, Manhattan, Chebyshev or Hamming may be used as the distance measure for determining the similarity of the datasets, though the Euclidean is most preferred and widely used [14].

The K-means algorithm basically follows these steps.

- A similarity or distance measure is chosen and used throughout.
- K number of centroids are chosen.
- The distance between each dataset from each of the k centroids is determined.
- Then a dataset is assigned to the centroid for which it had the minimum distance.
- All datasets are hence assigned to a particular centroid. Figure 4 shows a very simple illustration using 5 datasets and 2 clusters.
- The arithmetic mean is recalculated for each of the k centroids and the distance of each dataset from the new means is recalculated for each of the k centroids. This is the second iteration.
- The datasets are reassigned again to the new k centroids. In other words, a dataset assigned for instance to centroid 2 in the first iteration may be reassigned to centroid 1 in the second iteration.

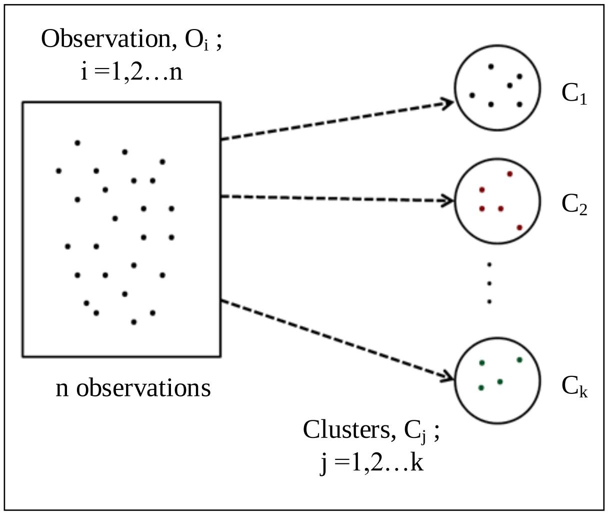
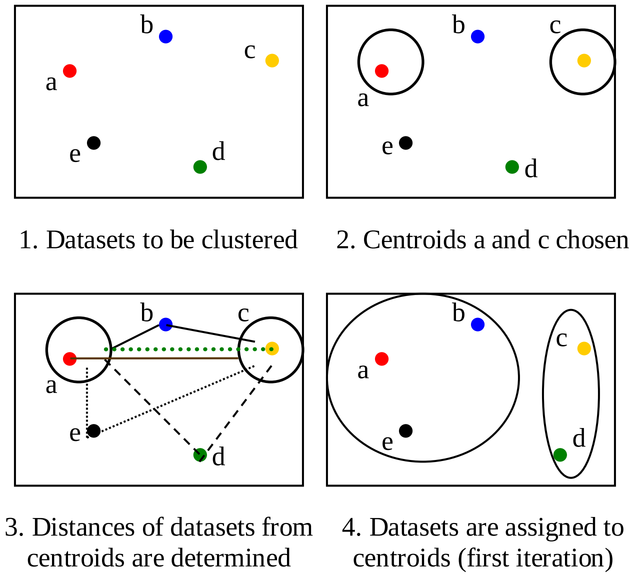


Figure 3. K-means Clustering

- The arithmetic means is recalculated and the data-sets reassigned again.
- This continues to i number of iterations, and the iteration stops when there is no change in the assignments between the ith iteration and the (i-1)th iteration.
- The last k centroids are the k clusters.



1. Datasets to be clustered

2. Centroids a and c chosen

3. Distances of datasets from centroids are determined

4. Datasets are assigned to centroids (first iteration)

Figure 4. K-means clustering algorithm

V. Approaches to Effective Cluster Analysis

Several challenges to cluster analysis include the difficulties in choosing an appropriate clustering algorithm; representing the data to be clustered; choosing a suitable similarity measure; determining which data should be used and choosing a suitable number of clusters that would yield maximum success. A user is not faced with these problems in hierarchical clustering analysis since all the datasets are related. These problems arise when a dataset is to be classified into unique clusters.

There are many partitional clustering algorithms and the user may be faced with the dilemma of choosing an appropriate algorithm. What guides in choosing an appropriate algorithm is to know the purpose and goal of the clustering exercise and this consequently would guide in representing that data to be clustered. It is also necessary to know if the dataset has a clustering tendency [18] and if it should be normalized. A data set that does not have a clustering tendency should not be clustered as it would yield invalid clusters. For example, if a data set that has all similar data and consequently has no variance is clustered it would result in invalid clusters. On the other hand a data set with high variance has a clustering tendency.

The choice of number of clusters may pose a problem because the performance of the clustering algorithm is affected by the number of clusters. It is usually difficult to determine the best number of partitions that will give the best and valid clustered groups. Some factors that may be considered while choosing the number of clusters are the size of the data set and the variance of the data in the data set. If the dataset is widely varied such that the data set may need to be classified by many groups then it may make sense to use more clusters.

In feature classification, the success of the cluster analysis is largely dependent on the feature set. The clustering algorithm would have a good performance and give compact, isolated and valid clusters if the choice of features is good [18]. If for instance a database of face images of a multi-racial group comprising African, Chinese, Latin American, Indian and European faces need to be clustered into five different groups, the features that would be used would be such that the faces can be effectively separated into five valid clusters. The success of this task is clearly dependent on the features used for the separation. The features making up the data set play a vital role in clustering analysis.

A similarity measure is required for separating data into clusters. The choice of the similarity measure is a challenging problem because the valid clustering of the data also depends on the similarity metric. The performance of the cluster analysis varies according to the similarity metric used and hence it may be difficult to determine the similarity metric that would give the best performance. But this problem can be overcome by having a good understanding of the data to be clustered.

VI. Clustering Used as a Fingerprint Indexing Retrieval Strategy

An indexing technique must include a retrieval strategy. A retrieval strategy defines the method for which data within the same class as the query or input data are retrieved. In fingerprint indexing, the retrieval strategy ensures that fingerprints with similar index codes [21] to that of the query fingerprint are retrieved from the database of enrolled fingerprints.

In this work, a modified Ross's partitional clustering scheme [22] is used as a fingerprint retrieval strategy by compressing the numerous fingerprint features into similar groups of data and hence limiting the search for similar fingerprints to only a few clusters that are identical to the cluster of the query fingerprint. This requires the following

- First the creation of an index space of k clusters for the indexing using the k-means algorithm and the Euclidean distance similarity measure.

- Secondly the assignment of the features of the fingerprints in the enrolled database to the k clusters.

- Thirdly the determination of the clusters, c << k, that have the features of the fingerprints similar to a query fingerprint.

A query fingerprint should have a matching identity in a list of fingerprints outputted by the indexing algorithm. This list is otherwise known as the candidate list. The ratio of the fingerprints in the candidate list to the database size gives the penetration rate of a query fingerprint. The penetration rate is the fraction of fingerprint identities, including the genuine fingerprint, retrieved from the database upon presentation of an input fingerprint. The penetration rate determined for a number of tests, T, in a database of size, N, is [23].

$$\boldsymbol{penetration\ rate} = \frac{\mathbf{1}}{\boldsymbol{T}}\sum_{\boldsymbol{j=1}}^{\boldsymbol{T}}\frac{\boldsymbol{C_j}}{\boldsymbol{N}} \qquad (8)$$

Where $\{C_j \mid j = 1, 2\ldots T\}$ is the size of candidate list of the fingerprints. The less the penetration rate the better the performance of the algorithm.

In this newly created file, highlight all of the contents and import your prepared text file. You are now ready to style your paper; use

VII. Experiments and Results

- The Fingerprint Verification Competition (FVC) 2000 database 4(a) and FVC 2002 database 4(a) were used for this experiment. Each database has 800 fingerprints from 100 subjects at 8 impressions per subject.
- The fingerprint features were extracted using the minutiae quadruplets technique [24].
- 30 clusters were created in the index space using fingerprints from FVC 2002 database 4(a).
- The FVC 2000 database 4(a) was divided into two equal groups – Group A and B.
- Group A had 400 fingerprints of the first four impressions of a subject
- Group B had 400 fingerprints of the last four impressions of a subject.
- The fingerprint features of group A were assigned to the 30 clusters in the index space.
- The fingerprints of group B were used to query the index space to find a matching identity determined by the penetration of the database. Every query resulted in a penetration rate. Majority of the queries had little penetration rates while some had long penetration rates.

The penetration rates of the 400 query fingerprints used in the experiments are shown in Table 1.

TABLE I. PENETRATION RATES OF 400 QUERY FINGERPRINTS IN AN EVALUATION ON FVC 2000

No of tests (T)	Penetration rates (range)	Average value	fx
57	0.25 - 1.00	0.75	42.75
55	1.25 - 2.00	1.625	89.375
39	2.25 - 3.00	2.625	102.375
49	3.25 - 4.50	3.875	189.875
50	4.75 - 6.50	5.625	281.25
47	6.75 - 8.50	7.625	358.375
55	8.75 - 14.25	11.5	632.5
48	14.5 - 38.25	26.375	1266
T=400			**∑fx=2962.5**

The average penetration for the 400 query fingerprints is obtained using Equation (8) and can also be determined from Table 1 as:

$$Average\ penetration\ = \frac{\sum f_x}{T} \qquad (9)$$

$$= \frac{2962.5}{400} = 7.406\%$$

Where f_x is the product of the first and third columns in Table I and T is the number of queries corresponding to the number of tests in the experiment. There were 400 queries.

The retrieval of a candidate list for a query fingerprint takes 0.592ms.

VIII. Comparison with Other Data Structuring Techniques

In [25], a binary tree based approach was used for matching fingerprints. The work done on this paper is indexing. However, the computational time for a fingerprint match using the binary tree technique in [25] is compared with the computational time for indexing a query fingerprint using the clustering technique described in this paper in Table II.

TABLE II. COMPARISON OF COMPUTATIONAL TIMES OF THE BINARY TREE AND CLUSTERING TECHNIQUES ON FVC 2002 DB1 SET A

Technique	Database size	Computational time
Binary tree [25]	800 fingerprints	34.8ms
Clustering (our approach)	400 fingerprints	0.592ms

IX. Conclusion

In this paper, clustering was discussed extensively. Experiments were conducted by employing a modified clustering scheme as a retrieval strategy for filtering fingerprints. The average penetration, 7.41%, is very small showing clearly that the clustering algorithm employed is an effective scheme for the filtering and retrieval of the candidate fingerprints to a given query fingerprint.

References

[1] H. J. Wolfson, "Geometric Hashing: An Overview" IEEE Computational Science and Engineering, pp. 10 – 21, 1997.

[2] S. Danker, R. Ayers and R. P. Mislan, "Hashing Techniques for Mobile Device Forensics" SMALL SCALE DIGITAL DEVICE FORENSICS JOURNAL, vol. 3, no. 1, June 2009 ISSN:1941-6164

[3] D. Zhang and J. Wang, "Self-Taught Hashing for Fast Similarity Search" SIGIR'10, July 19–23, 2010, Geneva, Switzerland

[4] H. Lim, J. Seo and Y. Jung, "High speed IP address lookup architecture using hashing" IEEE Communication Letters, vol. 7, no. 10, pp. 502 – 504, ISSN: 1089-7798, October 2003

[5] D. Greene, M. Parnas and F. Yao, "Multi-index hashing for information retrieval" Proceedings of the 35th Annual Symposium on the Foundations of Computer Science, 1994, pp. 722 – 731

[6] X. Nie, J. Liu, J. Sun and W. Liu, "Robust Video Hashing Based on Double-Layer Embedding" IEEE Signal Processing Letters, vol. 18, no. 5, pp. 307 – 310, May 2011

[7] B. Pfaff, "Performance Analysis of BSTs in System Software" SIGMETRICS/Performance'04, New York, NY, USA. June 12–16, 2004,

[8] D. D. Sleator and R. E. Tarjan, “Self-Adjusting Binary Search Trees” Journal of the Association for Computing Machinery. vol. 32, no. 3, July 1985, pp. 652-686.

[9] A.K. Baughman, S. Stockt, and A. Greenland, “Large Scale Fingerprint Mining.” 2010 ACM 978-1-4503-0220-3.

[10] A. K. Jain, A. Topchy, M. Law, and J. Buhmann, "Landscape of Clustering Algorithms", in Proceedings of the 17th International Conference on Pattern Recognition, Cambridge UK, August 23-26, 2004 pp. I-260--I-263.

[11] F. Alberto and G. Sergio, "Solving Non-uniqueness in Agglomerative Hierarchical Clustering Using Multidendrograms". Journal of Classification, Vol.25, 2008, pp.43–65.

[12] D. Apetrei, P. Postolache, N. Golovanov, M. Albu and G. Chicco, “Hierarchical Cluster Classification of Half Cycle Measurements in Low Voltage Distribution Networks for Events Discrimination.” International Conference on Renewable Energies and Power Quality (ICREPQ’09).

[13] A. Jain, M. N. Murty and P. Flynn, “Data clustering: A review.” ACM Computing Surveys, Vol.3, no. 13, 1999, pp. 264–323.

[14] A.K. Jain, Data Clustering: “50 Years Beyond K-Means.” Pattern Recognition Letters, Vol. 31, No. 8, 2010, pp. 651-666.

[15] M. Sahami, “Using Machine Learning to Improve Information Access.” Ph.D. Thesis, Computer Science Department, Stanford University. 1998.

[16] S. Bhatia and J. Deogun, “Conceputal clustering in information retrieval.” IEEE Transactions. Systems Man Cybernet. Vol. 28 (B), 1998, pp. 427–436.

[17] A. K. Jain and P. Flynn, “Image segmentation using clustering.” in Advances in Image Understanding. IEEE Computer Society Press, 1996, pp. 65–83.

[18] R.M.C.R. Souza and F.A.T. Carvalho, “Dynamic clustering of interval data based on adaptive Chebyshev distances.” Electronics Letters, Vol 40, Issue 11, ISSN: 0013-5194, 2004, pp. 658 – 660.

[19] L. Gąsieniec, J. Jansson and A. Lingas, “Approximation algorithms for Hamming clustering problems.” ELSEVIER. Journal of Discrete Algorithms, Vol. 2, 2004, pp. 289–301.

[20] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An Efficient k-Means Clustering Algorithm: Analysis and Implementation" IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 24, No. 7, July 2002. pp 881-892.

[21] D. Maltoni, D. Mario, A.K. Jain, and S. Prabhakar, “Handbook of Fingerprint Recognition.” 2nd Ed. Springer-Verlag London Limited. 2009.

[22] A. Ross and R. Mukherjee, “Augmenting Ridge Curves with Minutiae Triplets for Fingerprint Indexing.” In: Proc. of SPIE Conference on Biometric Technology for Human Identification IV. Orlando, USA. (April 2007).

[23] A. Gyaourova and A. Ross, “A Novel Coding Scheme for Indexing Fingerprint Patterns.” In Proc. Joint IAPR International Workshop on Structural, Syntactic, and Statistical Pattern Recognition. Springer-Verlag, 2008, pp 755 -764.

[24] O. Iloanusi, A. Gyaourova, A. Ross, "Indexing Fingerprints Using Minutiae Quadruplets," Proc. of IEEE Computer Society Workshop on Biometrics at the Computer Vision and Pattern Recognition (CVPR) Conference, (Colorado Springs, USA), 20-25 June 2011, pp 127 - 133.

[25] M. D Jain, N. S. Pradeep, C. Prakash and B. Raman, Binary tree based linear time fingerprint matching, IEEE International Conference on Image Processing (ICIP), 2006

AUTHORS PROFILE

Dr. Ogechukwu Iloanusi is a Lecturer at the Department of Electronic Engineering, University of Nigeria, Nsukka. She holds a Ph.D in Digital Electronics and Computer. Her research interests are biometric recognition and algorithm development, web-based applications, micro-processor based system design and e-learning..

Prof. Charles Osuagwu is a Professor of the Department of Electronic Engineering, University of Nigeria, Nsukka. He has an M. Sc and Ph.D in Engineering from the University of Southampton. His major research interests are Digital Signal Processing, Microprocessor-based System Design, Software Design, Computer Design, e-Security and Institutional Improvement.