Speaker specific feature based clustering and its applications in language independent forensic speaker recognition

Satyanand Singh[1], Pragya Singh[2]
[1]School of Electrical and Electronics Engineering, Fiji National University, Fiji Island
[2]School of Public Health and Primary Care, Fiji National University, Fiji Island

Article Info

Article history:

Received Dec 5, 2018
Revised Dec 18, 2019
Accepted Jan 11, 2020

Keywords:

Alternative dunn index (ADI)
Classification entropy (CE)
Dunn index (DI)
Fuzzy maximum likelihood (FML)
Partition coefficient (PC)
Partition index (SC)
Separation index(S)
Xie and Beni index (XB)

ABSTRACT

Forensic speaker recognition (FSR) is the process of determining whether the source of a questioned voice recording (trace) is of a specific individual (suspected speaker). Most existing methods measure inter-utterance similarities directly based on spectrum-based characteristics, the resulting clusters may not be well related to speaker's, but rather to different acoustic classes. This research addresses this deficiency by projecting language-independent utterances into a reference space equipped to cover the standard voice features underlying the entire utterance set. Then a clustering approach is proposed based on the peak approximation in order to maximize the similarities between language-independent utterances within all clusters. This method uses a K-medoid, Fuzzy C-means, Gustafson and Kessel and Gath-Geva algorithm to evaluate the cluster to which each utterance should be allocated, overcoming the disadvantage of traditional hierarchical clustering that the ultimate outcome can only hit the optimum recognition efficiency. The recognition efficiency of K-medoid, Fuzzy C-means, Gustafson and Kessel and Gath-Geva clustering algorithms are 95.2%, 97.3%, 98.5% and 99.7% and EER are 3.62%, 2.91 %, 2.82%, and 2.61% respectively. The EER improvement of the Gath-Geva technique based FSRsystem compared with Gustafson and Kessel and Fuzzy C-means is 8.04% and 11.49% respectively.

Corresponding Author:

Satyanand Singh,
School of Electrical and Electronics Engineering,
Fiji National University, Fiji Island.
Email: satyanand.singh@fnu.ac.fj

1. INTRODUCTION

Speaker recognition is the general term used to include all the many different tasks of discrimination based on the sound of their voices between one person and another [1]. Forensics means the use of science or technology in investigating and finding in the court of law facts or evidence. The role of forensic science is to provide information (in fact or opinion) to assist investigators and law courts in answering questions of importance. Forensic speaker recognition is the method of determining whether the origin of a questioned voice recording (trace) is a particular person (suspected speaker). This process involves comparing an unidentified voice recording (questioned recording) with one or more recordings of a known voice (the alleged speaker's voice) [1]. Forensic Automatic Speaker Recognition (FASR) is an established term used in the adaptation of automatic speaker recognition methods to forensic applications. For automated speaker identification, the deterministic or predictive models of the voice of the speaker's acoustic characteristics are contrasted with the acoustic characteristics of the recordings for question [1].

The clustering of speaker's refers to the function of grouping together unidentified speech expressions based on the voice characteristics of a speaker. The concerns and needs of the speaker recognition community have been a major motivation for the research on speaker clustering for more than a decade [2, 3], in which the main purpose is to band together speech data generated by the same speaker or speaker’s with similar voices so that the adaptation of acoustic models can be more effectively carried out. Because speech clustering simply serves as a supplementary process in speech recognition, however, there is still a dearth of studies dedicated to this subject. More recently, speaker-clustering work has experienced a renaissance [4], powered by research into spoken document indexing to handle burgeoning collections of accessible voice data. The main purpose of such an emerging research topic is that the human effort needed for documentation can be dramatically reduced by grouping speech data from the same speaker’s.

Speaker clustering can be described as an unsupervised speaker-recognition problem in which the speaker recognition process [5] is concerned with determining a speaker recognition or whether a speaker is who he/she claims. Contrary to the traditional speaker-recognition approach, however, which assumes that some contextual information or speech details is accessible and can be modeled on the speaker’s concerned, speaker clustering will function without any awareness of who the potential speaker’s are and how many are involved in the language to be clustered. Solutions to the speaker-clustering issue should therefore be able to extract and compare the speech characteristics that underlie the utterance collections in an unattended manner. Contrary to the traditional speaker-recognition approach, however, which assumes that some contextual information or speech details is accessible and can be modeled on the speaker’s concerned, speaker clustering will function without any awareness of who the potential speaker’s are and how many are involved in the language to be clustered. Solutions to the speaker-clustering issue should, therefore, be able to extract and compare the speech characteristics that underlie the utterance collections in an unattended manner. A similar activity is a segmentation of speaker’s [6], which seeks to identify the boundaries when a speaker change occurs in an audio stream containing the speech expressions of multiple people. Together with speaker clusters, the segmentation of speaker’s breaks the continuous input into discrete statements that are easy to process in speech/speaker recognition and is, therefore, an essential step in the indexing of spoken documents. Speaker segmentation can be accomplished from another angle with the help of speaker clustering. There may be a shift in speaker’s between two adjacent short regions with different cluster indices.

Most speaker-clustering methods currently follow a hierarchical clustering framework, consisting of three main components: computing inter-utterance similarities, generating a cluster tree and determining the number of clusters. The similarity equation was designed to generate higher values for similarities between the same speaker's utterances and lower values for similarities between different speaker's utterances. Many similarity tests, such as the Kullback Leibler (KL) distance [6-8], the cross probability ratio (CLR) [8], and the generalized probability ratio (GLR) are analyzed and contrasted in many works of literature. A cluster tree is created either in a bottom-up (agglomerative) or a top-down (divisive) fashion, according to some criteria derived from the measure of similarity. The bottom-up method starts as a single cluster with each utterance and then merges the clusters in a fair manner until all the utterances are found in one cluster. Nevertheless, all the utterances start in a single cluster in the top-down method, and the clusters are separated successively until each cluster has exactly one utterance. The resulting cluster tree is then split to maintain the best partition by estimating the number of clusters. Representative methods are based on the BBN Metric and the Bayesian Information Criterion to estimate the optimum number of clusters [8, 9].

In addition to developing a more accurate measurement of inter-utterance similarity, we also investigate how the clusters can be optimally produced so that all the utterances within the cluster are from the same speaker. Conventional methods based on either top-down or bottom-up hierarchical clustering use a nearest neighborhood selection rule to decide which pronouncements should be assigned to the same class. However, the closest neighborhood selection rule is applied in a cluster-by-cluster manner during the procedure of splitting one cluster or merging two clusters, rather than in a global manner that considers all the clusters. Consequently, hierarchical clustering can only make each individual cluster as homogeneous as possible, but the ultimate goal of maximizing overall homogeneity can not be achieved. To solve this problem, we are proposing a new clustering approach specifically aimed at maximizing the total number of statements from the same speaker’s within the cluster. This is achieved by estimating the so-called cluster purity in combination with a genetic algorithm-based optimization process [10] to find the best utterance partitioning to achieve maximum cluster purity.

2. FUNDAMENTAL OF SPEAKER SPECIFIC FEATURE BASED CLUSTERING ANALYSIS

The goal of cluster analysis is to categorize objects based on similarities between them and to organize data into groups. Among the unsupervised approaches, clustering techniques do not use prior class identifiers. Different classifications may be associated with the clustering techniques algorithmic approach. It is possible to distinguish partitioning, hierarchical, graph-theoretical methods and methods based on the objective function.

2.1. Speaker specific training data

It is possible to apply clustering techniques to information that is quantitative (numerical), qualitative (categorical), or a mixture of both. The clustering of quantitative data is being applied in language independent FSRsystem. Specific data from the speaker are typically physical process observations. Each observation of the speaker consists of n measured variables grouped into a n-dimensional line vector $X_k = [x_{k1}, x_{k2}, \dots\dots.., x_{kn}]^T$, $X_K \in R^n$. X denotes a set of N observations and is represented as a matrix of NXn:

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots \dots & x_{1n} \\ x_{21} & x_{22} & \dots \dots & x_{2n} \\ \dots & \dots \dots & \dots \dots & \dots \\ x_{N1} & x_{N2} & \dots \dots & x_{Nn} \end{bmatrix} \tag{1}$$

In the language of pattern recognition, X rows are called patterns or objects, the columns are called the characteristics or attributes, and X is called the pattern matrix. X is often referred to as the data matrix in this study. The significance of X's columns and rows in relation to reality depends on the context. For example, the rows of X may represent a speaker in application, and the columns are speaker specific measurements. As clustering is applied to dynamic system modeling and classification, the X rows contain time signals measurements, and the columns are, for example, speaker specific variables observed in the system (sentiment, emotion, ethnicity, etc.). The purpose of clustering in language independant forensic application is to find relationships between language independent system variables, called regressors, and speaker specific feature dependent variables values, called regressands [11].

2.2. Clustering algorithms in FSR application

Based on the purpose of clustering, various concepts of a cluster can be formulated. In general, the view that a cluster is a group of objects that are more similar to each other than members of other clusters can be embraced. In some well-defined sense, the term similarity should be understood as mathematical similarity. The similarity is often described in metric spaces by means of a distance standard. Distance to some cluster prototypical object can be measured between the data vectors themselves, or as a distance form a data vector.The prototypes are usually not known in advance and are searched simultaneously with data partitioning by the clustering algorithms. The models may be vectors of the same size as the data objects, but they may also be described as geometrical “higher-level” objects, such as linear or non-linear subspaces or functions.

2.2.1. Application of K-means and K-medoid algorithms

The methods of hard partitioning are simple and popular, although their results are not always reliable and these algorithms also have numerical problems. K-means and K-medoid algorithms allocate each data point to one of the c clusters from a NXn dimensional data set to minimize the sum of squares within the cluster:

$$\sum_{i=1}^{c} \sum_{k \in A_i} \| x_k - v_i \|^2 \tag{2}$$

where A_i is a set of objects in the I cluster (data points) and v_i is the mean over cluster I for that point in (2) in fact denotes ac distance standard. The cluster prototypes are called in K-means clustering v_i, i.e. the cluster centers:

$$v_i = \frac{\sum_{k=1}^{N_i} x_k}{N_i}, X_k \in A_i \tag{3}$$

where N_i is an entity number in A_i. The cluster centers are the nearest objects to the mean of information in one cluster $V = \{v_i \in X | 1 \leq i \leq c\}$ in K-medoid clustering.

2.2.2 Application of fuzzy C-means algorithm

The clustering algorithm for Fuzzy C-means is based on minimizing an objective function called functional C-means. Dunn defines it as:

$$J(X; U, V) = \sum_{i=1}^{c} \sum_{k=1}^{N} (\mu_{ik})^m \| X_k - V_i \|_A^2 \tag{4}$$

where $V = [V_1, V_2, \dots\dots, V_c], V_i \in R^n$ is a cluster model vector (centers) to be established and $D_{ikA}^2 = \| X_k - V_i \|_A^2 = (X_K - V_i)^T A (X_K - V_i)$ is a squared inner-product distance norm.

Statistically, as shown in (4) is the maximum variance of X_k from V_i can be seen as an indicator. Minimizing the functional c-means from as shown in (3), and (4) is a nonlinear problem of optimization that can be solved by using a variety of methods available, ranging from grouped coordination minimization to simulated annealing to genetic algorithms. Nevertheless, the most popular method is a simple Picard iteration, known as the fuzzy c-means (FCM) algorithm, through the first-order conditions for stationary points of (4). Using Lagrange multipliers, the stationary points of the objective function of (4) can be identified by applying the limit $\sum_{k=1}^{c}\mu_{ik} = 1, 1 \leq i \leq N$ to J:

$$\bar{J}(X;U,V,\lambda) = \sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^m D_{ikA}^2 + \sum_{i=1}^{N}\lambda_k\left(\sum_{i=1}^{c}\mu_{ik} - 1\right) \qquad (5)$$

Figure 1 shows the results of K-medoid algorithm of four speaker's of 6 sec voice data with normalization.

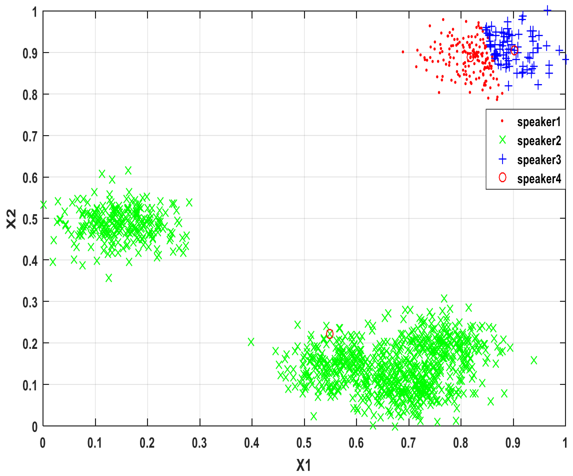

Figure 1. Result of K-medoid algorithm of four speaker's of 6 sec voice data with normalization

Moreover by setting the U;V and zero gradients of (J). If $D_{ikA}^2 \geq 0, m > 1$, then $(U,V) \in M_{fc}XR^{nXc}$ may minimize (3) only if;

$$\mu_{ik} = \frac{1}{\sum_{j=1}^{c}\left({}^{D_{ikA}}/_{D_{jka}}\right)^{2/(m-1)}}\quad 1 \leq i \leq c, 1 \leq k \leq N \qquad (6)$$

and

$$V_i = \frac{\sum_{k=1}^{N}\mu_{ik}^m x_k}{\sum_{k=1}^{N}\mu_{ik}^m}, 1 \leq i \leq c \qquad (7)$$

the remaining constraints $\mu_{ij} \in [0,1], 1 \leq i \leq N, 1 \leq k \leq c$ and $0 < \sum_{i=1}^{N}\mu_{ik} < N, 1 \leq k \leq c$ are also fulfilled by this solution. Note that (7) gives vi as the weighted mean of the data items belonging to a cluster where the weights are the degrees of membership.

The FCM algorithm is based on the standard Euclidean distance standard, which induces hyperspheric clusters. Therefore, clusters with the same form and orientation can only be identified, because the typical choice of norm inducing matrix is: $A = I$ or it can be chosen as a nXn diagonal matrix which accounts for different variances in the direction of the X coordinate axes:

$$A_D = \begin{bmatrix} \left(\frac{1}{\sigma_1}\right)^2 & 0 & \dots & 0 \\ 0 & \left(\frac{1}{\sigma_2}\right)^2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \left(\frac{1}{\sigma_n}\right)^2 \end{bmatrix} \qquad (8)$$

or A can be described as the inverse of nXn matrix of covariance: $A = F^{-1}$, with $F = \frac{1}{N}\sum_{k=1}^{N}(X_k - \bar{X})\,(X_k - \bar{X})^T$. Here $\bar{X}$ denotes the data sample mean. A induces the Mahalanobis standard on R^n in this case. Figure 2 shows the results of Fuzzy C-means algorithm of four speaker's of 6 sec voice data with normalization.

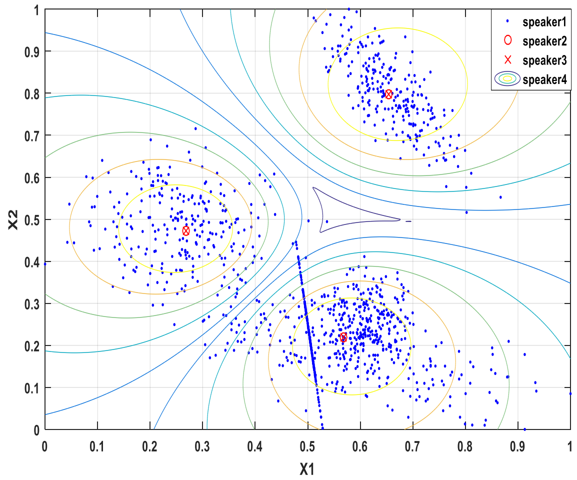

Figure 2. Result of fuzzy C-means algorithm of four speaker's of 6 sec voice data with normalization

2.2.3. Application of gustafson and kessel algorithm

By using an adaptive distance method, Gustafson and Kessel expanded the standard fuzzy c-means algorithm to detect clusters of various geometric shapes in one array of data [12].

$$D_{ikA}^2 = (X_k - V_i)^T A_i (X_k - V_i), 1 \le i \le c, 1 \le k \le N \quad (9)$$

The A_i matrices are used in the c-means functional as optimization variables, allowing the cluster to adapt the distance norm to the data's local topological structure. Let A denote a c-tuple of matrices that induce norm: $A = (A_1, A_2, A_2, \ldots A_C)$. The GK algorithm's functional objective is defined by:

$$J(X; U, V, A) = \sum_{i=1}^{c}\sum_{k=1}^{N}(\mu_{ik})^m D_{ikA_i}^2 \quad (10)$$

conditions $\mu_{ij} \in [0,1], 1 \le i \le N, 1 \le k \le c,\ \ \sum_{k=1}^{c}\mu_{ik} = 1, 1 \le i \le N$ and $0 < \sum_{i=1}^{N}\mu_{ik} < N, 1 \le k \le c$ may be applied directly to a fixed A. The objective function (9) in relation to A_i however, cannot be directly minimized as it is linear in A_i. This means that by simply making A_i less optimistic, J can be rendered as low as desired. A_i must be constrained in some way in order to find a feasible solution. The usual way to do this is to limit A_i determinant. Allowing the matrix A_i to differ with its defined determinant means optimizing the shape of the cluster while staying constant in volume $\|A_i\| = \rho_i, \rho > 0$. Where ρ_i for every cluster is set. The following expression for A_i is obtained using the Lagrange multiplier method as $A_i = [\rho_i det(F_i)]^{\frac{1}{n}} F_i^{-1}$. Where F_i is the I cluster's fuzzy covariance matrix defined by:

$$F_i = \frac{\sum_{k=1}^{N}(\mu_{ik})^m (X_k - V_i)(X_k - V_i)^T}{\sum_{k=1}^{N}(\mu_{ik})^m} \quad (11)$$

Remember that replacing $A_i = [\rho_i det(F_i)]^{\frac{1}{n}} F_i^{-1}$ and (11) with (9) gives a generalized square Mahalanobis distance norm between xk and the mean V_i cluster where the covariance is weighted by the membership degrees. Figure 3 shows the results of the Gustafson and Kessel clustering algorithm of four speaker's of 6 sec voice data with normalization.

2.2.4. Application of gath-geva algorithm

The clustering algorithm fuzzy maximum likelihood (FML) uses a distance standard based on the fuzzy maximum likelihood Estimates suggested by Bezdek and Dunn [13]:

$$D_{ik}(X_k, V_i) = \frac{\sqrt{det(F_{wi})}}{\alpha_i} exp\left(\frac{1}{2}\left(X_k - V_i^{(l)}\right)^T F_{wi}^{-1}\left(X_k - V_i^{(l)}\right)\right) \tag{12}$$

notice that this distance norm includes an exponential concept, contrary to the Gustafson and Kessel algorithm, and therefore decreases faster than the internal product standard. F_{wi} denotes the fuzzy covariance matrix of ith cluster, given by:

$$F_{wi} = \frac{\sum_{k=1}^{N}(\mu_{ik})^w (X_k - V_i)(X_k - V_i)^T}{\sum_{k=1}^{N}(\mu_{ik})^w}, 1 \leq i \leq \tag{13}$$

where w=1 is used in the original FML algorithm, but the w=2 weighting exponent is used to make the partition more fuzzy to compensate for the exposure of the distance standard. The difference between the matrix F_i in Gustafson and Kessel algorithm and the F_{wi} mentioned above is that the latter does not include the weighting exponent m, but consists of w=1 instead. This is because the two weighted matrices of covariance derive from two different concepts as generalizations of the classical covariance. The α_i is the prior probability of choosing cluster *i*, that is given by:

$$\alpha_i = \frac{1}{N}\sum_{k=1}^{N}\mu_{ik} \tag{14}$$

According to the speaker data point X_k the membership degrees μ_{ik} are interpreted as the posterior probability of selecting the *I* cluster. Gath and Geva [12] stated that clustering algorithms were capable of detecting clusters of varying shapes, sizes and densities from the fuzzy maximum likelihood estimates. The cluster covariance matrix is used in combination with an "exponential" length, and there is no volume constraint on the clusters. This algorithm, however, is less robust in the sense that it needs a good initialization because it converges to a near-local optimum due to the exponential distance norm. Figure 4 shows the results of the Gustafson-Kessel clustering algorithm by three speaker's of 6 sec voice data.

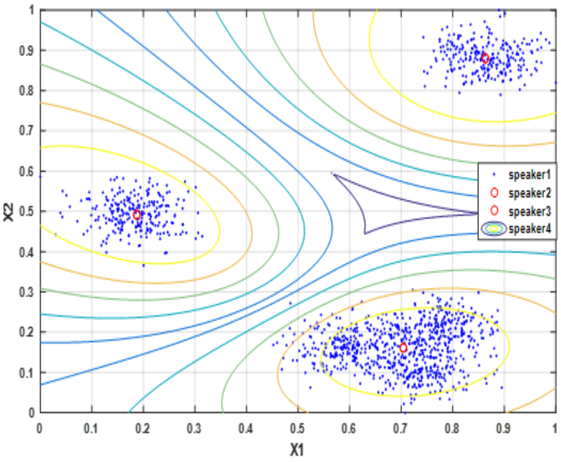

Figure 3. The results of the gustafson and kessel clustering algorithm of four speaker's of 6 sec voice data with normalization

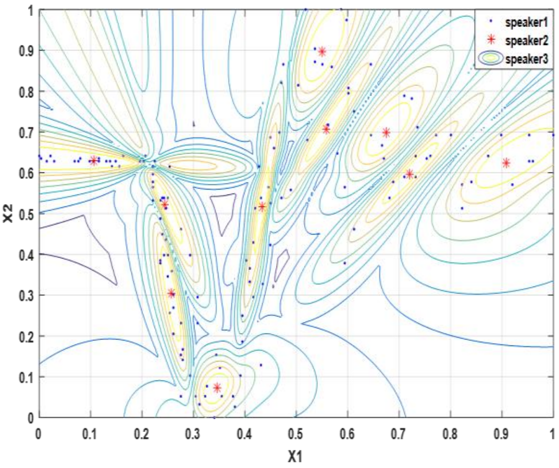

Figure 4. The results of the gustafson-kessel clustering algorithm by three speaker's of 6 sec voice data

2.2.5. Cluster validation

For each partition, the validity function provides cluster validity measurements. If the number of clusters is unknown apriori, it is useful. The optimal partition can be determined depending on the number of clusters on the extreme point of the validation indexes. Computed indexes include: Partition Coefficient (PC), Classification Entropy (CE), Partition Index (SC), Separation Index(S), Xie and Beni Index (XB), Dunn Index (DI) and Alternative Dunn Index (ADI).

- Partition Coefficient (PC): the amount of "overlap" between clusters is computed by $PC(c) = \frac{1}{N}\sum_{i=1}^{c}\sum_{j=1}^{N}\left(\mu_{ij}\right)^2$.
- Classification Entropy (CE): it only measures the cluster partition's fuzzyness, which is similar to the partition coefficient $CE(s) = -\frac{1}{N}\sum_{i=1}^{c}\sum_{j=1}^{N}\mu_{ij}\log(\mu_{ij})$.

- Partition Index (SC): is the compactness maximum ratio and cluster separation ratio. It is a total of uniform individual cluster validity tests by dividing each cluster's fuzzy cardinality [14], S $SC(c) = \sum_{i=1}^{c} \frac{\sum_{j=1}^{N} (\mu_{ij})^m \|x_j - v_i\|^2}{N_i \sum_{k=1}^{c} \|v_k - v_i\|^2}$. When comparing different partitions with the same number of clusters, SC is useful. A lower SC value shows a better partition.
- Separation Index (S): unlike partition index (SC), the separation index uses a partition validity minimum range separation [14], $S(c) = \frac{\sum_{i=1}^{c} \sum_{j=1}^{N} (\mu_{ij})^2 \|x_j - v_i\|^2}{N min_{i,k} \|v_k - v_i\|^2}$.
- Xie and Beni's Index (XB): aims at quantifying the ratio of total cluster variation and cluster separation, $(c) \frac{\sum_{i=1}^{c} \sum_{j=1}^{N} (\mu_{ij})^2 \|x_j - v_i\|^2}{N min_{i,j} \|x_j - v_i\|^2}$. The optimum number of clusters should minimize the index value.
- Dunn's Index (DI): it was originally suggested that this index be used to classify "compact and well separated clusters." It is therefore important to recalculate the outcome of the clustering as it was a hard partition algorithm, $DI(c) = min_{i \in c} \left\{ min_{j \in c, i \neq j} \left\{ \frac{min_{x \in C_i, C_j} d(x,y)}{max_{k \in c} \{ max_{x,y \in C} d(x,y) \}} \right\} \right\}$.
- Alternative Dunn Index (ADI): the purpose of modifying the original Dunn index was to simplify the calculation when the difference between two clusters works, $ADI(c) = min_{i \in c} \left\{ min_{j \in c, i \neq j} \left\{ \frac{min_{x_i \in C_i, x_j \in C_j} \left| d(y, v_j) - d(x_i - v_j) \right|}{max_{k \in c} \{ max_{x,y \in} d(x,y) \}} \right\} \right\}$. Note, the only difference between SC, S and XB is the cluster separation process. Due to re-partitioning the results with the hard partition process, the values of DI and ADI are not consistent in the case of overlapped clusters. Figure 5 shows the results of the Gustafson-Kessel clustering algorithm by three speaker's of 6 sec voice data. Table 1 illustrate Gustafson and Kessel algorithm based numerical values of validity measures of clustering.

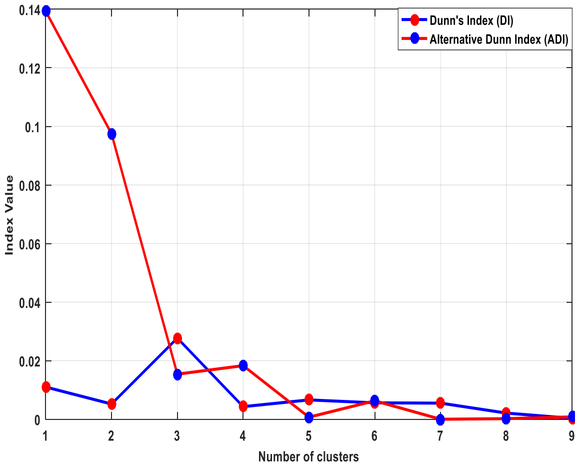

Figure 5. Values of dunn's index and the alternative dunn index of 6 sec voice data

Table 1. Gustafson and kessel algorithm based numerical values of validity measures of clustering

c	1	2	3	4	5	6	7	8	9
PC	0.8728	0.7982	0.8555	0.7573	0.7666	0.7341	0.6693	0.6879	0.6837
CE	0.2219	0.3307	0.2646	0.4139	0.4078	0.4853	0.6441	0.5778	0.6355
SC	35.0909	0.2786	0.3237	0.1612	0.2777	0.1953	0.1405	0.2299	0.2121
S	0.2638	0.0035	0.0037	0.0021	0.0032	0.0021	0.0014	0.0023	0.0022
XB	10.0966	8.4901	4.5690	6.7059	6.8081	11.7229	5.6257	5.7584	5.7868
DI	0.0110	0.0052	0.0277	0.0043	0.0066	0.0056	0.0055	0.0021	0.0003
ADI	0.1393	0.0975	0.0154	0.0183	0.0007	0.0063	0.0000	0.0002	0.0008

3. KERNEL-BASED SPEAKER SPECIFIC FEATURE EXTRACTION ANDT ITS APPLICATION

Classification algorithms must represent the objects to be classified as points in a multidimensional feature space. However, one can apply other vector space transformations to the initial features before running the learning algorithm. There are two reasons for doing this. First, they can improve the performance of classification and second, they can reduce the data's dimensionality. The selection of initial features and their transformation are sometimes dealt with in the literature under the title "feature extraction”. To avoid misunderstanding, this section describes only the latter and describes the first feature set. Hopefully it will be more effective and classification will be faster. The approach to the extraction of features may be either linear or nonlinear, but there is a technique that breaks down the barrier between the two forms in some way. The key idea behind the kernel technique was originally presented in [15] and applied again in connection with the general purpose SVM [16, 17, 18] followed by other kernel-based methods.

3.1. Supplying input variable information into kernel PCA

Additional information to the KPCA representation for interpretability. We have developed a process to project a given input variable into a subspace spanned by feature vectors $\tilde{V} = \sum_{i=1}^{m} \tilde{\alpha}\tilde{\phi}(X_1)$. We can think of our observation as a random vector $X = (X_1, X_2, \ldots\ldots, X_n\ )$ implementation then to represent the prominence of the input variable X_k in the KPCA. Considering a set of points of mathematical forms $y = a + se_k \in \mathbb{R}^n$ where $e_k = (0, \ldots.,1, \ldots.,0)$ of kth component is either 0 or 1. Next, the projection points $\phi(y)$ of these images onto the subspace spanned by the feature vector $\tilde{V} = \sum_{i=1}^{m} \tilde{\alpha}\tilde{\phi}(X_1)$ can be calculated. Considering (12) the row vector gives the induction curve in the Eigen space expressed in matrix form:

$$\sigma(s)_{1Xr} = \left(Z_s^T - \frac{1}{m} 1_m^T K\right)\left(I_m - \frac{1}{m} 1_m 1_m^T\right)\tilde{V} \tag{15}$$

furthermore, by projecting the tangent vector to s = 0, we can express the maximum change direction of $\sigma(s)$ associated with the variable X_k. Matrix form of the expression represented as follows:

$$\left.\frac{d\sigma}{ds}\right|_{s=0} = \left.\frac{dZ_s^T}{ds}\right|_{s=0} \left(I_m - \frac{1}{m} 1_m 1_m^T\right)\tilde{V} \tag{16}$$

where

$$\left.\frac{dZ_s^T}{ds}\right|_{s=0} = \left(\left.\frac{dZ_s^1}{ds}\right|_{s=0}, \ldots\ldots.., \left.\frac{dZ_s^m}{ds}\right|_{s=0}\right)^T$$

and

$$\left.\frac{dZ_s^i}{ds}\right|_{s=0} = \left.\frac{dK(Y,X_i)}{ds}\right|_{s=0} = \left.\left(\sum_{t=1}^{m} \frac{\partial K(Y,X_i)}{\partial Y_t}\frac{dY_t}{ds}\right)\right|_{s=0} = \sum_{t=1}^{m} \left.\frac{\partial K(Y,X_i)}{\partial Y_t}\right|_{Y=a} \delta_t^k = \left.\frac{\partial K(Y,X_i)}{\partial Y_k}\right|_{Y=a}$$

Where delta of Kronecker is represented as δ_t^k and radial basis kernel as $k(Y, X_i) = exp(-c\|Y - X_i\|^2) = exp(-c\sum_{t=1}^{n}(Y_i - X_{it})^2)$. After considering $y = a + se_k \in \mathbb{R}^n$:

$$\left.\frac{dZ_s^i}{ds}\right|_{s=0} = \left.\frac{\partial K(Y,X_i)}{\partial Y_k}\right|_{y=a} = -2cK(a, X_i)(a_k - X_{ik}) = -2cK(X_\beta, X_i)(X_{\beta k} - X_{ik})$$

Where the training point $a = X_\beta$. Thus, by applying (13), it is possible to locally represent any given input variable plot in KPCA. Furthermore, by using (14), it is possible to represent the tangent vector associated with any given input variable at each sample point [19]. Therefore, a vector field can be drawn on KPCA indicating the growth direction of a given variable. There are some existing techniques to compute z for specific kernels [20]. For a Gaussian kernel $(X, Y)) = exp(-\|X - Y\|^2/2\sigma^2)$, z must satisfy the following condition.

$$Z = \frac{\sum_{i=1}^{m} \gamma_i\left(\|Z-X_i\|^2/2\sigma^2\right)X_i}{\sum_{i=1}^{m} \gamma_i(-\|Z-X_i\|^2)/2\sigma^2} \tag{17}$$

4. EXPERIMENTAL SETUP

To evaluate the efficiency of kernel-based speaker-specific feature extraction techniques, a language independent utterances recognition experiment was performed. The experiment includes 520 Japanese words from the ATR Japanese C language set Voice database, 80 speaker’s (40 men and 40 Female). Audio samples of 10 iTaukei speaker’s were collected at random and under unfavourable conditions. The average duration of the training samples was six seconds per speaker for all 10 speaker’s and out of twenty utterances of each speaker just one was used for training purpose [21]. For matching purposes the remaining 19 voice samples were used from the corpus. We have recorded utterances for this investigation were at one sitting for each speaker. The text for the utterances was randomly selected by speaker. The main voice recordings consist of both male and female speaker's of twenty utterance of each using sampling rate of 16 kHz with 16 bits/sample [22].

Speech features, each consisting of 24 Mel-scale frequency cepstral coefficients (MFCCs), were extracted from these utterances for every 20-ms Hamming-windowed frame with 10-ms frame shifts. Prior to MFCC computation, voice active detection was applied to remove salient non-speech regions that may be included in an utterance [23,24]. Our basic strategy is to create an utterance-independent GMM using all the utterances to be clustered, followed by an adaptation of the utterance-independent GMM for each of the utterances using maximum a posteriori (MAP) estimation [25]. This technique comes from the GMM-UBM strategy [26] for FSR in which the necessary speaker-explicit models are made by tuning the parameters of a widespread speaker model pre-prepared by utilizing discourse information from numerous speaker's. We are using Language independent Gaussian mixture modeling followed by MAP adaptation in language independent forensic speaker recognition.

Throughout the experiment, 10400 utterances were used as training data and the remaining 31,200 utterances were used as test data. The sampling rate of the audio signal is 16 kHz. 13 Mel-Cepstral coefficients extracted using 25.6 ms Hamming windows with 10 ms shifts. Figure 6 show the equal error rate (EER) of K-medoid, Fuzzy C-means, Gustafson and Kessel and Gath-Geva clustering algorithms Geva for 6 sec of voice data of ATR Japanese C language. The forensic speaker recognition efficiency of of K-medoid, Fuzzy C-means, Gustafson and Kessel and Gath-Geva clustering algorithms are 95.2%, 97.3%, 98.5% and 99.7% and EER are 3.62%, 2.91%, 2.82% and 2.61% respectively. The EER improvement of Gath-Geva technique based FSRsystem compared with Gustafson and Kessel and Fuzzy C-means is 8.04% and 11.49% respectively.

Table 2 illustrate efficiency and EERof the FSRsystem for of K-medoid, Fuzzy C-means, Gustafson and Kessel and Gath-Geva clustering algorithms respectively for ATR Japanese C language. Figure 7 show the equal error rate (EER) K-medoid, Fuzzy C-means, Gustafson and Kessel and Gath-Geva clustering algorithms Geva for 6 sec of voice data of 10 iTaukei speaker's cross language.The FSRefficiency of of K-medoid, Fuzzy C-means, Gustafson and Kessel and Gath-Geva clustering algorithms are 93.2%, 96.6%, 97.7% and 98.8% and EER are 4.23%, 3.42%, 3.33% and 3.11% respectively. The EER improvement of Gath-Geva technique based FSRsystem compared with Gustafson and Kessel and Fuzzy C-means is 7.07% and 9.96% respectively. Table 3 illustrate Efficiency and EERof the FSRsystem for of K-medoid, Fuzzy C-means, Gustafson and Kessel and Gath-Geva clustering algorithms respectively for 10 iTaukei speaker's cross language.

In addition, to decide how many clusters should be produced, the clustering method has been integrated with the Bayesian information criterion. Experimental results show that the number of clusters automatically determined can approximate the actual population size of the speaker. An experimental evaluation of the performance of the forensic recognition system efficiency of K-medoid, Fuzzy C-means, Gustafson and Kessel and Gath-Geva clustering algorithms are 95.2%, 97.3%, 98.5% and 99.7% and EER are 3.62%, 2.91%, 2.82% and 2.61% respectively. The EER improvement of Gath-Geva technique based FSRsystem compared with Gustafson and Kessel and Fuzzy C-means is 8.04% and 11.49% respectively.

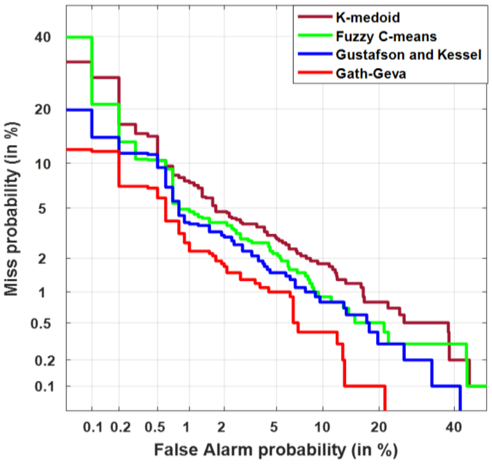

Figure 6. Equal error rate of K-medoid, fuzzy C-means, gustafson and kessel and gath-geva for 6 sec of voice data of ATR japanese C language

Table 2. Efficiency and EERof the FSRsystem for of K-medoid, fuzzy C-means, gustafson and kessel and gath-geva clustering algorithms respectively for ATR japanese C language

	Efficiency in %	EER in %
K-medoid	95.2	3.62
Fuzzy C-means	97.3	2.91
Gustafson and Kessel	98.5	2.82
Gath-Geva	99.7	2.61

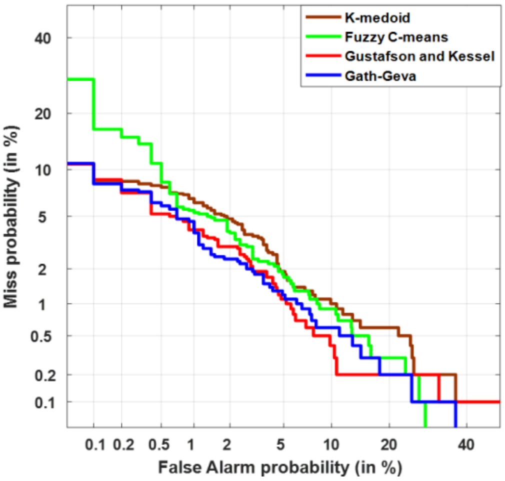

Figure 7. Equal error rate of K-medoid, fuzzy C-means, gustafson and kessel and gath-geva for 6 sec of voice data of itaukei speaker's cross language

Table 3. Efficiency and EERof the FSRsystem for of K-medoid, fuzzy C-means, gustafson and kessel and gath-geva clustering algorithms respectively for 10 itaukei speaker's cross language

	Efficiency in %	EER in %
K-medoid	93.2	4.23
Fuzzy C-means	96.6	3.42
Gustafson and Kessel	97.7	3.33
Gath-Geva	98.8	3.11

5. CONCLUSION

This study examined methods for improving the measurement of inter-utterance similarity for speaker clustering. The relationships of similarity between the utterances to be clustered can be exploited more easily and efficiently by using a voice characteristic reference space. We presented several implementations for the development of reference spaces. In particular, in order to capture the most representative characteristics of the voices of speaker's, the reference space was represented as a set of eigenvectors obtained by applying the technique of self-voice to the set of expressions to be clustered. This has resulted in a significant improvement in speaker-clustering performance compared to the traditional inter-speech similarity measure based on the generalized likelihood ratio. However, we have researched the method for creating clusters outside traditional hierarchical clustering so that all within-cluster utterances can be from a single speaker as far as possible. This criterion was conceived as a problem of calculating and optimizing the overall purity of the cluster. By representing cluster purity as a function of inter-utterance similarity and applying the genetic algorithm to find a solution to this problem, we have demonstrated a further increase in speaker-clustering efficiency compared to conventional agglomeration hierarchical clustering.

REFERENCES

[1] S. Singh, "Forensic and Automatic Speaker Recognition System," in *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 5, pp. 2804-2811, October 2018.

[2] S. Singh, "Support Vector Machine Based Approaches For Real Time Automatic Speaker Recognition System," *International Journal of Applied Engineering Research*, vol. 13, no. 10, pp. 8561-8567, 2018.

[3] S. Singh, “The Role of Speech Technology in Biometrics, Forensics and Man-Machine Interface,” *International Journal of Electrical and Computer Engineering*, vol. 9, no. 1, pp. 281-288, February 2019.

[4] S., Kwon and S., Narayanan, “Unsupervised speaker indexing using generic models,” in *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 5, pp. 1004-1013, Sept. 2005.

[5] J. P. Campbell, “Speaker recognition: a tutorial,” in *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437-1462, Sept. 1997.

[6] B. Zhou and J. H. L. Hansen, “Unsupervised audio stream segmentation and clustering via the Bayesian information criterion,” *6th International Conference on Spoken Language Processing (ICSLP 2000)*, October 2000.

[7] D. A. Reynolds, E. Singer, B. A. Carson, G. C. O’Leary, J. J. McLaughlin, and M. A. Zissman, “Blind clustering of speech utterances based on speaker and language characteristics,” *5th International Conference on Spoken Language Processing (ICSLP 98)*, Sydney, Australia, December 1998.

[8] W. H. Tsai, S. S. Cheng, and H. M. Wang, “Automatic speaker clustering using a voice characteristic reference space and maximum purity estimation,” in *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 4, pp. 1461-1474, May 2007.

[9] S. S. Chen and P. S. Gopalakrishnan, “Clustering via the Bayesian information criterion with applications in speech recognition,” *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP '98 (Cat. No.98CH36181)*, Seattle, WA, USA, vol. 2, pp. 645-648, 1998.

[10] D. E. Goldberg, “Genetic Algorithm in Search, Optimization and Machine Learning,” *Addison-Wesley Professional; 1 edition*, New York, pp. 432, January 1989.

[11] S. Singh and P. Singh, “High level speaker specific features modeling in automatic speaker recognition system,” in *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 2, pp. 1859-1867, April 2020.

[12] D. E. Gustafson and W.C. Kessel, “Fuzzy clustering with fuzzy covariance matrix,” *1978 IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes*, San Diego, CA, USA, pp. 761-766, 1978.

[13] J. Abonyi, R. Babuska, F. Szeifert, “Modified Gath–Geva fuzzy clustering for identification of Takagi–Sugeno fuzzy models,” in *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 32, no. 5, pp. 612-621, Oct. 2002.

[14] Maria Halkidi, Yannis Batistakis, Michalis Vazirgiannis, “On Clustering Validation Techniques,” *Journal of Intelligent Information Systems*, vol. 17, no. 3, pp. 107-145, 2001.

[15] Gerazov, B., Ivanovski, Z., “Kernel Power Flow Orientation Coefficients for Noise-Robust Speech Recognition,” in *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 2, pp. 407-419, Feb. 2015.

[16] Geiger, J., Schuller, B.; Rigoll, G., “Large-scale audio feature extraction and SVM for acoustic scene classification,” *2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, pp. 1-4, 2013.

[17] Rabaoui, A., Davy, M., Rossignaol, S., Ellouze, N., “Using One-Class SVMs and Wavelets for Audio Surveillance,” in *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 4, pp. 763-775, Dec. 2008.

[18] Jiang, H., Bai, J., Zhang, S., Xu, B., “SVM-based audio scene classification,” *2005 International Conference on Natural Language Processing and Knowledge Engineering*, Wuhan, China, pp. 131-136, 2005.

[19] Mika, B. Scholkopf, A. J. Smola, K.-R. Muller, M. Scholz, and G. Ratsch, “Kernel PCA and de–noising in feature spaces,” *Proceedings of the 1998 conference on Advances in neural information processing systems II*, pp. 536-542, July 1999.

[20] Kim, K. I., Jung, K., Kim, H. J., “Face Recognition Using Kernel Principal Component Analysis,” in *IEEE Signal Processing Letters*, vol. 9, no. 2, pp. 40-42, Feb. 2002.

[21] S. Singh, “Speaker Recognition System for Limited Speech Data Using High-Level Speaker Specific Features and Support Vector Machines,” *International Journal of Applied Engineering Research*, vol. 12, no. 19, pp. 8026-8033, January 2017.

[22] S. Singh, M. H Assaf and Abhay Kumar, “A Novel Algorithm of Sparse Representations for Speech Compression/Enhancement and Its Application in Speaker Recognition System,” *International Journal of Computational and Applied Mathematics*, vol. 11, no. 1, pp. 89-104, 2016.

[23] S. Singh, “Evaluation of Sparsification algorithm and Its Application in Speaker Recognition System,” *Internationa Journal of Applied Engineering Research*, vol. 13, no. 17, pp. 13015-13021, 2018.

[24] S. Singh and Dr. E. G. Rajan, “MFCC VQ based Speaker Recognition and Its Accuracy Affecting Factors,” *International Journal of Computer Applications*, vol. 21, no. 6, May 2011.

[25] S. Singh and Mansour H. Assaf, “A Perfect Balance of Sparsity and Acoustic hole in Speech Signal and Its Application in Speaker Recognition System,” *Middle-East Journal of Scientific Research*, vol. 24, no. 11, pp. 3527-3541, 2016.

[26] Yi-Hsiang Chao, Wei-HoTsai and Hsin-MinWang, “Improving GMM-UBM speaker verification using discriminative feedback adaptation,” *Computer Speech & Language*, vol. 23, no. 3, pp. 376-388, 2009.