EUSIPCO 2013 1569739163

# MULTI-CLASS UBM-BASED MLLR M-VECTOR SYSTEM FOR SPEAKER VERIFICATION

*A. K. Sarkar and C. Barras*

LIMSI-CNRS, Université Paris-Sud, BP 133, 91403 Orsay, France

{sarkar,barras}@limsi.fr

## ABSTRACT

In this paper, we extend the recently introduced Maximum Likelihood Linear Regression (MLLR) super-vector based m-vector speaker verification system to a multi-class MLLR m-vector system. In the conventional case, global class MLLR transformation is estimated with respect to Universal Background Model (UBM) for a given speech data, which is then used in the form of super-vector for m-vector system. In the proposed system, Gaussian mean vectors of the UBM are first clustered into several classes. Then, MLLR transformations are estimated (of a speech data) for each class, and are used in the form of super-vectors for speaker characterization using the m-vector technique. We consider two clustering approaches: one is based on the conventional K-means and the other is proposed based on Expectation Maximization (EM) and Maximum Likelihood (ML). Both systems yield better performance than the conventional m-vector system and allow for multiple MLLR transforms without additional temporal alignment of the data with respect to UBM. Furthermore, we show that, contrary to conventional K-means, the proposed clustering is not affected by the random initialization, and also provides equal or comparable system performance. The system performances are shown on NIST 2008 SRE core condition over various tasks.

***Index Terms*—** Multi-class m-vector, Statistical clustering algorithm, MLLR super-vector, UBM, Speaker verification

## 1. INTRODUCTION

During the last decade, different high-dimensional features have been proposed for use in Speaker Verification (SV) systems, among which is the *Maximum Likelihood Linear Regression (MLLR) super-vector* [1, 2, 3, 4, 5, 6]. In the MLLR approach, an affine transformation is estimated with respect to a Speaker Independent (SI) model for a given speech segment/utterance. Then, the MLLR transformation matrix is represented in the form of a super-vector and used as a feature for speaker modeling. Depending on the SI model, speaker verification systems based on MLLR super-vector can be broadly divided into two categories, i.e. Hidden Markov Models (HMMs) based [1, 4, 5] and Universal Background Model (UBM) based [3, 6]. In the first approach, each speech utterance is automatically transcribed and aligned against the states of the phonetic models. Then, one or more pre-defined phonetic class-specific MLLR transformations are estimated using this alignment. It is popularly known as Automatic Speech Recognition (ASR) based MLLR SV system. In the former case, commonly, a single (or global) class MLLR transformation is calculated with respect to the UBM without any speech transcription. The limitation of this later approach is that the UBM does not model the temporal successions of phonemes, and ASR based MLLR SV systems perform better than UBM based ones [5]. However, the main drawback of ASR based systems is the complexity of HMM modeling which makes the estimation of MLLR transformations computationally heavy, while UBM system usually consists 512-2048 Gaussian mixture components. Therefore, UBM based MLLR SV systems are better suited for real time applications.

MLLR super-vectors were commonly associated with a Support Vector Machines (SVM) classifier for speaker modeling [1, 2, 4] but alternative approaches have been recently proposed. Following the *m-vector technique* [5, 6], speakers are represented by a *uniform* segmentation of their MLLR super-vectors. Each segment of the MLLR super-vector is called an m-vector. The MLLR super-vector is derived from a *global MLLR transformation* which is estimated with respect to the UBM. The m-vector technique was shown to extract more speaker relevant information from the speaker MLLR super-vector than the conventional way of using the *full* super-vectors. It also gives promising performance when compared to a standard i-vector system in speaker verification. Later, the effectiveness of the m-vector technique has also been shown in [5] for speaker verification with ASR based MLLR super-vectors.

Our objective in this paper is to extend the conventional UBM based global (i.e. single class) MLLR m-vector system into a UBM based multi-class MLLR m-vector system, where Gaussian components of the UBM are first clustered into different groups. An MLLR transformation is estimated with respect to each cluster for a given speech segment. Finally, MLLR transforms are used as super-vectors for speaker characterization with the m-vector technique. We consider two clustering approaches: the first one is based on the conventional K-means and the other one relies on a proposed *statistical clustering* algorithm based on the concepts of Expectation Maximization (EM) and Maximum Likelihood (ML). The salient features of the proposed algorithm are that:

- the clustering is very robust to random initialization compared to conventional K-means;
- the performances are either equal or comparable to the best ones obtained with several experimental runs of K-means.

Compared to other multi-class MLLR based speaker verification found in the literature [1, 2, 4, 5], our system *did not use any* phonetic knowledge for the multi-class MLLR transformation. The system performance are shown on NIST 2008 SRE core condition over various tasks. Experimental results show that this multi-class MLLR m-vector system performs better than the conventional m-vector system.

The paper is organized as follows: Sec. 2 describes the MLLR super-vector concept. Sec. 3 describes the m-vector technique. The proposed system is described in Sec. 4. Sec. 5 describes the session variability compensation and scoring. Experimental setup is

Work was partly realized as part of the QUAERO Program and the QCOMPERE project, respectively funded by OSEO (French State agency for innovation) and ANR (French national research agency)

presented in Sec. 6. Sec. 7 describes results and discussion, before the conclusion in Sec. 8.

## 2. MLLR SUPER-VECTOR

MLLR [7] is a commonly used for speaker adaptation in Automatic Speech Recognition (ASR) systems. It estimates an affine transformation with respect to Speaker Independent (SI) HMMs for a given speech segment in the Maximum Likelihood (ML) sense. The affine transformation is then applied to the Gaussian mean vectors of the SI model to get the speaker adapted model parameters. The MLLR transformation $(W, b)$ can be expressed as,

$$\hat{\mu} = W\mu + b; \quad \hat{\Sigma} = \Sigma \tag{1}$$

where $\mu$ and $\Sigma$ indicates respectively, the Gaussian mean vectors and covariance matrices of the SI model, and $\hat{\mu}$ and $\hat{\Sigma}$ are the adapted speaker model parameters. In our experiments, a UBM is considered as the SI model for MLLR transformation. Estimation of the MLLR transformation $W$ given the feature vectors $X = \{x_1, \ldots x_T\}$ for the $r^{th}$ speaker involves the following steps:

**Step 1:** Determine the probabilistic alignment, $\gamma_j\,(t)$ of feature vector $X$ with respect to UBM for the $j^{th}$ Gaussian as:

$$\gamma_j\,(t) = p(j|x_t) = \frac{\omega_j b_j(x_t)}{\sum_{k=1}^{c} \omega_k b_k(x_t)} \tag{2}$$

where $c$ and $b_k$ indicate the number of Gaussians and density function of $k^{th}$ Gaussian of the UBM, respectively.

**Step 2:** Compute the following two sufficient statistics for $i^{th}$ *components (dimension) of feature vectors*,

$$K^{(i)} = \sum_{j=1}^{c}\sum_{t=1}^{T} \gamma_j(t)\, \frac{1}{\sigma_{ji}^2}\, x_i(t)\, \mu_j^{'} \tag{3}$$

$$G^{(i)} = \sum_{j=1}^{c} \frac{1}{\sigma_{ji}^2} \mu_j \mu_j^{'} \sum_{t=1}^{T} \gamma_j\,(t) \tag{4}$$

$\mu_j$ and $\sigma_{ji}^2$ are the $j^{th}$ mean and the $i^{th}$ component of $j^{th}$ covariance matrix of UBM, respectively. The symbol $(.)'$ indicates matrix transpose operation.

**Step 3:** $i^{th}$ *row of the MLLR transformation* of the $r^{th}$ speaker is obtained as,

$$W_i = K^{(i)} {G^{(i)}}^{-1} \tag{5}$$

**Step 4:** Repeat Step 2 to 3 upto feature vector dimension

MLLR super-vector is then formed by stacking the elements of the MLLR transformation one by one [1], as illustrated in Fig. 1. The bias $b$ is not considered in our experiments since it does not provide any significant system performance gain. We use 47 dimensional feature vectors resulting in $47 * 47 = 2209$ dimensional MLLR super-vectors.

## 3. M-VECTOR TECHNIQUE

In the m-vector technique [5, 6], speakers are represented by a *uniform* segmentation of their MLLR super-vectors using an *overlapped window*. (As Eqn.(5), rows of MLLR transformation i.e. super-vector is associated with the components of feature vectors.) Each

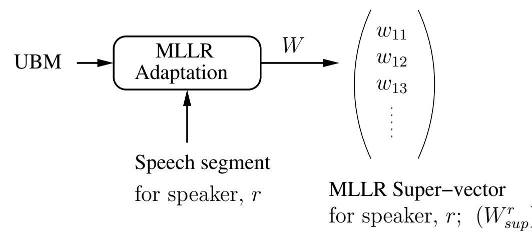


**Fig. 1**. *Estimation of MLLR super-vector for the $r^{th}$ speaker with respect to the UBM.*

segment is called an m-vector as shown in Fig.2. A speaker is thus characterized by several m-vectors which are processed separately and hence constitute several sub-systems. Before scoring, m-vectors are post-processed for session variability compensation; m-vectors of the test utterance are then scored against the claimant specific m-vectors obtained during the training phase.

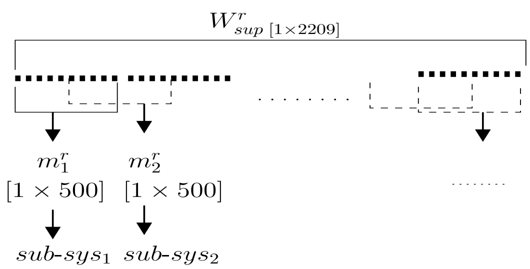


**Fig. 2**. *m-vector extraction for the $r^{th}$ speaker from his/her MLLR super-vector using an overlapped window of* 500 *elements.*

While dimension of MLLR super-vector is not integer divisible by window size, we extract an additional m-vector by putting the window at the end of the super-vector to cover the remaining elements.

## 4. UBM-BASED MULTI-CLASS M-VECTOR SYSTEM

In this section, we describe the proposed UBM based multi-class MLLR m-vector system. We consider two clustering algorithms for multi-class MLLR transformation, resulting in two systems.

### 4.1. Multi-class MLLR m-vector system with statistical clustering

In this case, Gaussian mean vectors of the UBM are first clustered using the concept of Expectation Maximization (EM) [8] and Maximum Likelihood (ML) as described in *Algorithm 1*.

Algorithm 1: *Proposed* statistical clustering algorithm

***Initial:*** Load the UBM and the chose number of clusters $L$

**Step 1:** Use the Gaussian mean vectors of the UBM as feature vectors, $Y = \{\mu_1, \mu_2, \ldots \mu_c\}$

**Step 2:** Train a $L$ components Gaussian Mixture Model (GMM) $\sim \mathcal{N}(\tilde{\omega}_i, \tilde{\mu}_i, \tilde{\Sigma}_i)$, $i = 1 \ldots L$, using the feature vectors $Y$ with EM algorithm after a *random partition initialization*

**Step 3:** Iterate EM algorithm in *Step 2* several times

**Step 4:** Separate *each Gaussian component* of the GMM obtained in *Step 2* as a *single Gaussian model* and discard the weights $\tilde{\omega}_i$ in order to give *equal importance* to all the models:

$$\lambda_i \sim \mathcal{N}(\tilde{\mu}_i, \tilde{\Sigma}_i) \tag{6}$$

**Step 5:** Assign the $c^{th}$ Gaussian mean vector of the UBM, i.e. $\mu_c$ to cluster $k$ in ML sense as,

$$k = \arg \max_{1 \le j \le L} p(\mu_c | \lambda_j) \tag{7}$$

In our experiment, we use 1000 iterations in *Step* 3 of *Algorithm 1* (with constraints on initial and final variance ceiling, flooring of global data). Though the parameters of the models $\lambda_1, \ldots \lambda_L$ are slightly different each run of the algorithm for a particular number of clusters, however it yields the same final clustering output, showing that this clustering algorithm is not affected by the random initialization.

Then, a MLLR transformation is estimated for a given speech data with respect to *each cluster* using the *sufficient statistics* accumulated from the *Gaussian components assigned to that particular cluster*. *Algorithm 2* describes in detail multi-class wise MLLR transformations for given $r^{th}$ target speaker data $X = \{x_1, \ldots x_T\}$ with respect to UBM.

Algorithm 2: Estimation of cluster-wise MLLR transformation

**Step 1:** Estimate $\gamma_j(t)$ for the feature vector $X$ with respect to the UBM as in Eqn.(2)

**Step 2:** For the $L^{th}$ class, compute the sufficient statistics using *Gaussian components* $\epsilon$ $L$ as in Eqn.(3-4),

$$K_L^{(i)} = \sum_{j \epsilon L} \sum_{t=1}^{T} \gamma_j(t) \, \frac{1}{\sigma_{ji}^2} \, x_i(t) \, \mu_j^{'} \tag{8}$$

$$G_L^{(i)} = \sum_{j \epsilon L} \frac{1}{\sigma_{ji}^2} \mu_j \mu_j^{'} \sum_{t=1}^{T} \gamma_j(t) \tag{9}$$

**Step 3:** $i^{th}$ row of the MLLR transformation for $L^{th}$ class is obtained,

$$W_i^L = K_L^{(i)} {G^{(i)}}_L^{-1} \tag{10}$$

**Step 4:** Repeat *Step 2 to 3* upto the number of classes

Fig. 3 graphically illustrates the UBM based multi-class MLLR transformations. The number of MLLR transformations and MLLR super-vectors per speaker depends on the number of classes chosen for the clustering. Finally, target speakers are represented by their m-vectors extracted from their MLLR super-vectors as described in Sec. 3. *It is to be noted that alignment of data i.e. Step 1 in Algorithm* 2 *is required* only once *irrespective estimation of number of class wise MLLR transformations with respect to UBM*.

### 4.2. K-means based multi-class MLLR m-vector system

The K-means based system is similar to *statistical clustering based multi-class* MLLR m-vector system. The only difference is that the *conventional K-means* algorithm with random initialization is used for clustering, associated with the euclidean distance measure. Clustering is stopped (i.e. converge) when cluster associated elements are not altered. We use the terms "conventional K-means" and "K-means" interchangeably throughout the paper.

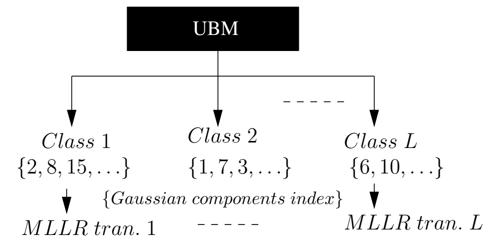


**Fig. 3**. *Clustering of UBM Gaussian components and estimation of an MLLR transformation with respect to each cluster.*

## 5. SESSION VARIABILITY COMPENSATION AND SCORING

Linear Discriminant Analysis (LDA) is applied on the m-vectors in order to reduce the dimensions of the vectors and increase the discrimination between speakers. LDA projected m-vectors are then post-processed for speaker session variability compensation. Several post-processing techniques are available in literature, namely LDA followed by Within-Class Covariance Normalization (WCCN), Eigen Factor Radial (EFR) [9] and Probabilistic (P)-LDA [10] etc. We chose the EFR recently introduced for i-vector based speaker verification systems [9]. In our case, iterative normalization of the length of the m-vectors is performed as,

$$\hat{m} \leftarrow \frac{V^{-\frac{1}{2}}(m - \overline{m})}{\sqrt{(m - \overline{m})' V^{-1} (m - \overline{m})}} \tag{11}$$

where $V$ and $\overline{m}$ are the covariance matrix and mean vector of the training m-vectors respectively for successive iterations. $V$ and $\overline{m}$ are calculated from development data set collected over many non-target speakers.

During the test, a Mahalanobis-based scoring function described in Eqn.(12) is used for scoring between two m-vectors (i.e. $\hat{m}_1, \ \hat{m}_2$):

$$score(\hat{m}_1, \hat{m}_2) = (\hat{m}_1 - \hat{m}_2)^{'} \Omega^{-1} (\hat{m}_1 - \hat{m}_2) \tag{12}$$

where $\Omega$ is the within-class covariance matrix computed using development data with non-target speakers. It is to be noted that LDA and EFR are implemented separately for each sub-system (as in Fig.2) i.e. each sub-system has its own LDA transformation, $\Omega$ and $V$ etc. Two iterations of EFR are used during post-processing for all systems presented in the paper. Finally, scores of the m-vectors i.e. sub-systems are fused together for a particular LDA dimension. For fusion, equal weights are given to all sub-systems,

$$fused\ score = \frac{1}{N_{sub}} \sum_{i=1}^{N_{sub}} score(\hat{m}_i^{test}, \hat{m}_i^{r}) \tag{13}$$

where $\hat{m}_i^{test}$ and $\hat{m}_i^{r}$ denote respectively, the LDA-EFR processed m-vectors of test utterance and claimant, $r$ for $i^{th}$ subsystem.

## 6. EXPERIMENTAL SETUP

The baseline system in our experiment is similar to [6], where global MLLR transformations (i.e. super-vectors) are estimated for each speaker and are processed through the m-vector technique. The global MLLR transformation is estimated with respect to a UBM for a given speech segment without any speech transcriptions.

Experiments are performed on NIST 2008 SRE core condition for all male speakers following the evaluation plan [11]. There are 1270 speech utterances for training 1270 male target models. Each utterance is around 5 minutes long and contains in average 2.5 minutes of speech.

For signal processing, 47 dimensional Perceptual Linear Predictive (PLP) feature vectors (15 static with their $\Delta$, $\Delta\Delta$, $\Delta$E and $\Delta\Delta$E) are extracted from the speech signal at 10 ms rate over the 0-3800 Hz bandwidth. Voice activity detection is then applied on the feature vectors to discard the silent or of low energy frames. Finally, selected frames are normalized to zero mean and unity variance at utterance level.

A UBM of 512 Gaussian components with diagonal covariance matrices is trained using male data from NIST 2004 SRE. LDA and EFR are implemented with 12399 utterances which are collected over 890 non-target speakers (NIST 2004-05, Switchboard II part 1, 2 & 3; Switchboard cellular part 1 & 2, about 15 sessions per speaker).

MLLR transformations are estimated with a single iteration in all systems. We impose a constraint that if estimation of the MLLR transformation is not possible for a class due to singularity problem during inverse of $G^{(i)}$ matrix for lack of sufficient frames, we consider the global MLLR transformation as a substitute transformation for that particular class. Equal Error Rate (EER) and Minimum Detection Cost Function (MinDCF) are used for system performance measurement as per NIST 2008 SRE plan [11].

## 7. RESULTS AND DISCUSSION

### 7.1. Comparison of performance of m-vector system with conventional approach of speaker characterization

In this section, we compare the performance of m-vector technique (using the overlapped window method) with the system where speakers are characterized by their *full* MLLR super-vectors (called *full* system). Table 1 presents the system performance on NIST 2008 SRE core condition det 7 task. In the case of m-vector, system performance is shown for m-vector dimension of 500 elements (i.e. overlap window size 500).

**Table 1**. *Comparison of performance of the m-vector technique with the system of speakers characterization by their full MLLR super-vector on NIST 2008 SRE core condition (det 7 task).*

| System | m-vector dim. | LDA Opt. dim. | EER(%)/(MinDCF) |
|---|---|---|---|
| (A) Full | 2209 | 200 | 4.37 /(0.0272) |
| (B) m-vector (overlapped) | 500 | 400 | **3.46 /(0.0237)** |
| (A+B)‡ | - | - | **3.45** /(**0.0193**) |

‡ linear fusion. (A+B) refers as (full+overlapped)

From Table 1, similarly to [6], it can be observed that m-vector system shows significantly better performance than the *full* system. It reflects the fact that m-vectors extract more speaker relevant information from the MLLR super-vector than the *full* approach. Further, fusion of system A with B reduce speaker verification error rate, especially the MinDCF. This shows that the *full* system also contains some speaker related information complementary to the m-vector system, which is not covered by the overlapped window method. In rest of the paper, other all system performances are compared in the fused (across a particular LDA dimension) *full+overlapped (m-vector dimension of* 500*)* framework.

### 7.2. Analysis performance of the proposed multi-class MLLR m-vector system with proposed statistical and conventional K-means algorithms

In this section, we first show the effect of the number of class-wise MLLR transformations on the proposed UBM based multi-class MLLR m-vector system for speaker verification. Table 2 presents the system performance for the respective clustering algorithms on NIST 2008 SRE core condition det 7 task in terms of EER. For simplicity, optimal LDA projected dimension is not shown in the table. In case of K-means, we perform 10 *pass of experiments* with random initialization of clustering for a particular number of clusters and provide the system performance of a run which yields *the best* result.

**Table 2**. *Effect of number of class-wise MLLR transformations in proposed UBM based multi-class MLLR m-vector system for speaker verification on NIST 2008 SRE core condition (det 7 task).*

| m-vector system | Clustering Algorithm | # of class-wise MLLR trans. | EER (%) |
|---|---|---|---|
| Baseline | - | 1 (global) | 3.45 |
| Proposed Multi-class MLLR | Proposed Statistical | **2**; (358,154) | **3.21** |
| | | 3; (100,100,312) | 3.44 |
| | K-means | 2; (252,260) | **3.22** |
| | | **3**; (170,180,162) | **3.08** |

(,) shows # of Gaussians of respective classes

From Table 2, we can observe that:

- The proposed multi-class MLLR m-vector system shows lower speaker verification error rate than the baseline system.
- Both algorithms show similar system performance for 2 clusters. For 3 clusters, K-means may perform better than the proposed clustering in the best cases, but it can be observed from Table 3, that most experiments results obtained with K-means for 3 classes are similar to those of the proposed clustering. The K-means system performance with 2 or 3 clusters are very close and both can be optimal depending on the case. Therefore, we do not proceed our experiments beyond 3 clusters.
- Proposed *Statistical* clustering algorithm shows equal or comparable performance to the *conventional K-means*.

In order to show the effect of clustering with random initialization in conventional K-means algorithm, we compare the performance (in terms of EER) of the multi-class MLLR m-vector speaker verification system based on K-means and the proposed method in Table 3 on NIST 2008 SRE core condition det 7 task for various number of class wise MLLR transformations.

From Table 3, we can make the following observations:

- Speaker Verification (SV) performance of the multi-class MLLR m-vector system with *K-means* varies across the runs, in contrast to the proposed statistical clustering algorithm. This shows that *K-means* clustering algorithm is affected by the random initialization and gives different clustering output for different runs in contrast to the proposed statistical algorithm.
- The proposed algorithm also yields equal or comparable performance to the *best result obtained* with *K-means*. This indicates that the proposed method always provides optimal clustering. Moreover, it does not need to run many times experiments unlike *K-means* for system judgment.

**Table 3**. *Effect of clustering with random initialization in proposed statistical algorithm and conventional K-means based multi-class MLLR m-vector system for speaker verification on NIST 2008 SRE core condition (det 7 task)*

| Exp. No. | # of class-wise MLLR trans. [% EER] | | | |
|---|---|---|---|---|
| | **2** | | **3** | |
| | Statistical | K-means | Statistical | K-means |
| 1 | **3.21** | 3.25 | 3.44 | 3.57 |
| 2 | nc | 3.37 | nc | 3.83 |
| 3 | nc | 3.52 | nc | 3.30 |
| 4 | nc | 3.45 | nc | **3.08** |
| 5 | nc | 3.52 | nc | 3.35 |
| 6 | nc | 3.48 | nc | 3.79 |
| 7 | nc | **3.22** | nc | 3.83 |
| 8 | nc | 3.44 | nc | 3.32 |
| 9 | nc | 3.47 | nc | 3.31 |
| 10 | nc | 3.63 | nc | 3.62 |

nc=no change

### 7.3. Comparison of performance of baseline system with the proposed multi-class MLLR m-vector system

In this section, we compare the performance of the baseline system with the proposed multi-class MLLR m-vector system using *statistical clustering* on various det tasks (e.g. det 5: telephone-microphone, det 7: telephone-telephone) in NIST 2008 SRE core condition for 2 and 3 class wise MLLR transforms, respectively for the proposed and K-means clustering, which showed best system performance in previous section.

**Table 4**. *Comparison of performance between the baseline system and the proposed multi-class MLLR m-vector system for* statistical clustering *(in case of* 2 *classes) and K-means (in case of* 3 *classes) algorithm on NIST 2008 SRE core condition over various tasks.*

| m-vector system | %EER/(MinDCF) det 5 | det 6 | det 7 | det 8 |
|---|---|---|---|---|
| Baseline | 7.11 (0.0351) | 6.46 (0.0392) | 3.45 0.0193 | 2.92 (0.0155) |
| Multi-class Stat. clus. (2 class) | **5.51** (**0.0298**) | 6.62 (**0.0382**) | **3.21** (**0.0191**) | **2.20** (**0.0121**) |
| K-means (3 class) | **5.55** (**0.0300**) | 6.50 (**0.0380**) | **3.08** (**0.0181**) | **2.16** (**0.0101**) |

From Table 4, it is observed that multi-class MLLR m-vector shows lower error rate in most of the det tasks in terms of EER and MinDCF. Multi-class m-vector with proposed algorithm having 2 classes also shows very comparable performance to the system which is even obtained with 3 classes in K-means. This further also reveals the optimality of the proposed clustering technique. Moreover, the proposed system does not require an additional alignment of data with respect to UBM for estimation of multiple MLLR transformations (see Algorithm 2) compared to the baseline system.

## 8. CONCLUSION

In this paper, we extended the recently introduced global MLLR super-vector based m-vector concept in a UBM framework for speaker verification into a multi-class MLLR m-vector. We addressed two clustering algorithms for multi-class wise MLLR transformations for m-vector system by partitioning the Gaussian components of the UBM into different classes. One is based on conventional K-means and the other proposed statistical algorithm is based on Expectation Maximization (EM) and Maximum Likelihood (ML). We showed that the proposed multi-class MLLR based m-vector system performs better than the conventional m-vector system. Furthermore, it does not require an additional temporal alignment of data with respect to UBM for estimation of multiple MLLR transformations. The system performances are compared on NIST 2008 SRE core condition over various det tasks. Second, we showed that the proposed clustering algorithm is not affected by the random initialization unlike conventional *K-means* algorithm and hence provides stable clustering output compared to *K-means*. Furthermore, it also gives speaker verification performance equal or comparable to the best one obtained with the *K-means* algorithm (over various pass run of experiments).

## 9. REFERENCES


[1] A. Stolcke et al., "MLLR Transforms as Features in Speaker Recognition," in *Proc. of EUROSPEECH*, 2005, pp. 2425–2428.

[2] M. Ferràs, C.C. Leung, C. Barras, and J.-L. Gauvain, "Comparison of Speaker Adaptation Methods as Feature Extraction for SVM-Based Speaker Recognition," *IEEE TASLP*, vol. 18, no. 6, pp. 1366–1378, August 2010.

[3] A. K. Sarkar and S. Umesh, "Eigen-voice Based Anchor Modeling System for Speaker Identification using MLLR Super-vector," in *Proc. of INTERSPEECH*, 2011, pp. 2357–2360.

[4] Z. N. Karam and W. M. Campbell, "A Multi-class MLLR Kernel for SVM Speaker Recognition," in *Proc. of ICASSP*, 2008, pp. 4117–4120.

[5] A. K. Sarkar, C. Barras, and V. B. Le, "Lattice MLLR based *m-vector* System for Speaker Verification," in *Proc. of ICASSP*, 2013, Vancouver, Canada.

[6] A. K. Sarkar, J. F. Bonastre, and D. Matrouf, "Speaker Verification using m-vector Extracted from MLLR Super-vector," in *Proc. of EUSIPCO*, 2012, pp. 21–25.

[7] C. Leggetter and P. Woodland, "Maximum Likelihood Linear Regression for Speaker Adaptation of HMMs," *Computer Speech and Language*, vol. 9, pp. 171–186, 1995.

[8] A. Dempster, N. Laird, and D. Rubin, "Maximum Likelihood from Incomplete Data via EM Algorithm," *J. Roy. statist. Soc.*, vol. 39, pp. 1–38, 1977.

[9] P. M. Bousquet, D. Matrouf, and J. F. Bonastre, "Intersession Compensation and Scoring Methods in the i-vectors Space for Speaker Recognition," in *Proc. of INTERSPEECH*, 2011, pp. 485–488.

[10] M. Senoussaoui et al., "Mixture of PLDA Models in I-Vector Space for Gender-Independent Speaker Recognition," in *Proc. of INTERSPEECH*, 2011, pp. 25–28.

[11] The NIST Year 2008 Speaker Recognition Evaluation Plan., "http://www.itl.nist.gov/iad/mig/tests/sre/2008/sre08_evalplan_release4.pdf," .