# FUZZY GAUSSIAN MIXTURE MODELS FOR SPEAKER RECOGNITION

*Dat Tran, T. Van Le and Michael Wagner*

Human-Computer Communication Laboratory
School of Computing, Faculty of Information Sciences and Engineering
University of Canberra, ACT 2601, Australia
E-mail: (dat, vanl, miw)@hcc1.canberra.edu.au

## ABSTRACT

A fuzzy clustering based modification of Gaussian mixture models (GMMs) for speaker recognition is proposed. In this modification, *fuzzy mixture weights* are introduced by redefining the distances used in the fuzzy *c*-means (FCM) functionals. Their reestimation formulas are proved by minimising the FCM functionals. The experimental results show that the fuzzy GMMs can be used in speaker recognition and it is more effective than the GMMs in tests on the TI46 database.

## 1. INTRODUCTION

In speaker recognition, the GMMs are used to model the distribution of spectral feature vectors of speakers. The model parameters which are *mean vectors, covariance matrices* and *mixture weights* are trained in an unsupervised classification using the expectation maximisation (EM) algorithm [6,8,11]. This algorithm provides an iterative maximum likelihood estimation technique. Experiments have shown that GMMs are effective models capable of achieving high identification accuracy for short utterance lengths from unconstrained conversational speech [8].

FCM clustering is the most widely used approach in both theory and practical applications of fuzzy clustering techniques to unsupervised classification. It is an extension of the hard *c*-means algorithm and was first introduced by Dunn [2]. From the classical within groups sum of squared errors function Dunn first generalised, the infinite family of FCM functionals were generalised by Bezdek [4], where a weighting exponent *m* on each fuzzy membership and a distance in *A* norm (*A* is any positive definite matrix) were introduced. The FCM algorithms are used to minimise the FCM functionals, where *fuzzy mean vectors* are iteratively updated. Gustafson and Kessel [3] proposed a modification of the FCM algorithms which attempts to recognise the fact that different clusters in the same data set may have differing geometric shapes. These algorithms were referred to as fuzzy covariance clustering algorithms where *fuzzy covariance matrices* of clusters were defined.

A fuzzy clustering based modification of GMMs is proposed in this paper. To obtain this modification, the distances in the FCM functionals are redefined as the negative of logarithms of density functions, which are products of mixture weights and Gaussian functions. These distances are used in *entropy constrained vector quantisation* (VQ) or *generalised k-means* VQ approaches in speech and speaker recognition [5]. In this modification, *fuzzy mixture weights* are defined and are proved together with *fuzzy mean vectors*, *fuzzy covariance matrices* in the reestimation formulas. The GMMs in this modification could be named *fuzzy Gaussian mixture models* (FGMMs).

## 2. GAUSSIAN MIXTURE MODELS

Let $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$ be a data set of $T$ vectors $\mathbf{x}_t$, each of which is a $d$-dimensional feature vector extracted from digital speech processing. Since the distribution of these vectors is unknown, it is approximately modelled by a Gaussian mixture density, which is a weighted sum of $c$ component densities, given by the equation

$$p(\mathbf{x}_t \mid \lambda) = \sum_{i=1}^{c} p_i N(\mathbf{x}_t, \mu_i, C_i) \qquad (2.1)$$

where $p_i$, $i = 1,\ldots, c$, are the mixture weights, $N(\mathbf{x}_t, \mu_i, C_i)$, $i = 1,\ldots, c$, are the $d$-variate Gaussian component densities with mean vector $\mu_i$ and covariance matrix $C_i$

$$N(\mathbf{x}_t, \mu_i, C_i) = \frac{exp\left\{-\frac{1}{2}(\mathbf{x}_t - \mu_i)' C_i^{-1} (\mathbf{x}_t - \mu_i)\right\}}{(2\pi)^{d/2} |C_i|^{1/2}} \qquad (2.2)$$

where $(\mathbf{x}_t - \mu_i)'$ is the transpose of $(\mathbf{x}_t - \mu_i)$ and $\lambda$ is a set of all parameters contained in the probability model, $\lambda = \{p_i, \mu_i, C_i\}$, $i = 1,\ldots, c$. In training the GMM, these parameters are estimated such that in some sense, they best match the distribution of the training vectors. The most popular estimation method is the maximum likelihood (ML) estimation. For a sequence of training vectors $X$, the likelihood of the GMM is

$$p(X \mid \lambda) = \prod_{t=1}^{T} p(\mathbf{x}_t \mid \lambda) \qquad (2.3)$$

The aim of ML estimation is to find a new parameter model $\bar{\lambda}$ such that $p(X \mid \bar{\lambda}) \geq p(X \mid \lambda)$. Maximising $p(X \mid \lambda)$ in applications is not easy, hence an auxiliary

function $Q$ is used

$$Q(\lambda,\overline{\lambda}) = \sum_{i=1}^{T} p(i \mid \mathbf{x}_t, \lambda) log[\, \overline{p}_i N(\mathbf{x}_t, \overline{\mu}_i, \overline{C}_i)] \qquad (2.4)$$

where $p(i \mid \mathbf{x}_t, \lambda)$ is the a posteriori probability for acoustic class $i$, $i = 1,\ldots,c$ and satisfies

$$p(i \mid \mathbf{x}_t, \lambda) = \frac{p_i N(\mathbf{x}_t, \mu_i, C_i)}{\sum_{k=1}^{C} p_k N(\mathbf{x}_t, \mu_k, C_k)} \qquad (2.5)$$

Maximising the $Q$ function is performed using the EM algorithm. The basis of the EM algorithm is that if $Q(\lambda,\overline{\lambda}) \geq Q(\lambda,\lambda)$ then $p(X \mid \overline{\lambda}) \geq p(X \mid \lambda)$ [6,8]. Setting derivatives of the $Q$ function with respect to $\overline{\lambda}$ to zero, the following reestimation formulas are found

$$\overline{p}_i = \frac{1}{T} \sum_{t=1}^{T} p(i \mid \mathbf{x}_t, \lambda) \qquad (2.6)$$

$$\overline{\mu}_i = \frac{\sum_{t=1}^{T} p(i \mid \mathbf{x}_t, \lambda) \mathbf{x}_t}{\sum_{t=1}^{T} p(i \mid \mathbf{x}_t, \lambda)} \qquad (2.7)$$

$$\overline{C}_i = \frac{\sum_{t=1}^{T} p(i \mid \mathbf{x}_t, \lambda)(\mathbf{x}_t - \overline{\mu}_i)(\mathbf{x}_t - \overline{\mu}_i)'}{\sum_{t=1}^{T} p(i \mid \mathbf{x}_t, \lambda)} \qquad (2.8)$$

The algorithm for training the GMM is described as follows

**Algorithm 1:**

*Step 1:* Generate the a posteriori probability $p(i \mid \mathbf{x}_t, \lambda)$ at random satisfying (2.5)
*Step 2:* Compute the mixture weight, the mean vector, and the covariance matrix following (2.6), (2.7) and (2.8)
*Step 3:* Update the a posteriori probability $p(i \mid \mathbf{x}_t, \lambda)$ following (2.5) and compute the $Q$ function following (2.4)
*Step 4:* Stop if the increase in the value of the $Q$ function at the current iteration relative to the value of the $Q$ function at the previous iteration is below a chosen threshold, otherwise go to step 2.

## 3. FUZZY CLUSTERING

Consider the above-mentioned data set $X$ of $d$-dimensional vectors $\mathbf{x}_t$, $t = 1,\ldots, T$. Its structure can be analysed by means of *cluster analysis technique*. Clustering also known as unsupervised learning or self-organisation in $X$ is a partitioning of $X$ into $c$ subsets or $c$ clusters, $1 < c < T$. The most important requirement is to find a suitable measure of clusters, referred to as a clustering criterion. Objective function methods allow the most precise formulation of the clustering criterion. The most well known objective function for fuzzy clustering in $X$ is the least-squares functionals, the infinite family of fuzzy c-means (FCM) functionals, generalised from the classical within groups sum of squared errors function by Bezdek [4]

$$J_m(U, \mu; X) = \sum_{t=1}^{T} \sum_{i=1}^{c} u_{it}^m d_{it}^2 \qquad (3.1)$$

where $U = \{u_{it}\}$ is a fuzzy $c$-partition of $X$, each $u_{it}$ represents the degree of vector $\mathbf{x}_t$ belonging to the $i$th cluster, for $1 \leq i \leq c$ and $1 \leq t \leq T$, we have

$$0 \leq u_{it} \leq 1 \text{ and } \sum_{i=1}^{C} u_{it} = 1; \qquad (3.2)$$

$m \geq 1$ is a weighting exponent on each fuzzy membership $u_{it}$ and is called the degree of fuzziness;
$\mu = (\mu_1, \ldots, \mu_c)$ are cluster centers and, $d_{it}$ is the distance in the $A$ norm from $\mathbf{x}_t$ to $\mu_i$, known as a measure of dissimilarity

$$d_{it}^2 = \| \mathbf{x}_t - \mu_i \|_A^2 = (\mathbf{x}_t - \mu_i)' A (\mathbf{x}_t - \mu_i) \qquad (3.3)$$

The basic idea in FCM is to minimise $J_m$ over the variables $U$ and $\mu$, on the assumption that matrices $U$ that are part of optimal pairs for $J_m$ identify good partitions of the data. Minimising the fuzzy objective function $J_m$ in (3.1) gives

$$u_{it} = \left[ \sum_{k=1}^{c} (d_{it} / d_{kt})^{2/(m-1)} \right]^{-1} \qquad (3.4)$$

$$\mu_i = \sum_{t=1}^{T} u_{it}^m \mathbf{x}_t \Big/ \sum_{t=1}^{T} u_{it}^m \qquad (3.5)$$

The FCM algorithm is known as the *fuzzy vector quantisation* (FVQ) algorithm in speech and speaker recognition and is used to train codebooks in the VQ approach. This algorithm is described as follows

**Algorithm 2:**

*Step 1:* Choose any inner product norm metric for $\mathbf{R}^d$, fix $c$ and $m$, $2 < c < T$, $m > 1$. Generate matrix $U$ at random satisfying (3.2)
*Step 2:* For $i = 1,\ldots, c$, compute the *c fuzzy mean vectors* $\{\mu_i\}$ with (3.5) and the distances $d_{it}$ with (3.3). If $d_{it} = 0$ for some $t$, set $u_{it} = 1$, $u_{is} = 0$, $\forall s \neq t$
*Step 3:* Update matrix $U$ using (3.4)
*Step 4:* Stop if the decrease in the value of the fuzzy objective function $J_m$ at the current iteration relative to the value of the $J_m$ at the previous iteration is below a chosen threshold, otherwise go to step 2.

An interesting modification of the FCM algorithm was proposed by Gustafson and Kessel [3,4]. It attempts to

recognise the fact that different clusters in the same data set $X$ may have differing geometric shapes. A generalisation to a metric which appears more natural was made, through the use of a fuzzy covariance matrix. Replacing (3.3) by an inner product induced norm metric of the form

$$d_{it}^2 = (\mathbf{x}_t - \mu_i)' M_i (\mathbf{x}_t - \mu_i) \tag{3.6}$$

with $M_i$ symmetric and positive definite. Define a *fuzzy covariance matrix* $C_i$ by

$$C_i = \sum_{t=1}^{T} u_{it}^m (\mathbf{x}_t - \mu_i)(\mathbf{x}_t - \mu_i)' \Big/ \sum_{t=1}^{T} u_{it}^m \tag{3.7}$$

then $M_i^{-1} = (|M_i||C_i|)^{-1/d} C_i$ (3.8)

where $|M_i|$ and $|C_i|$ are the determinants of $M_i$ and $C_i$, respectively and $d$ is the feature space dimension. $|M_i|$ is constrained by a fixed parameter for each $i$ [3].

Step 2 in the algorithm 2 is now generalised by computing the $c$ *fuzzy mean vectors* $\{\mu_i\}$ with (3.5), the *fuzzy covariance matrix* $C_i$ with (3.7) and the distances $d_{it}$ with (3.6). If $d_{it} = 0$ for some $t$, set $u_{it} = 1$, $u_{is} = 0$, $\forall s \neq t$.

## 4. FUZZY GAUSSIAN MIXTURE MODELS

A further modification of the FCM algorithm is proposed in this paper. Our goal is to apply FCM estimate to an Bayesian classifier in the particular case of a mixture of $c$ Gaussian distributions. It attempts to recognise the fact that different clusters in the same data set $X$, beyond differing geometric shapes, may have differing data densities, denoted by mixture weights (class *a priori* probabilities). A generalisation to a metric is made, through the use of a *fuzzy covariance matrix* and a *fuzzy mixture weight*. To obtain these, since the density of the data in cluster $i$ is proportional to the joint mixture density function $f(\mathbf{x}_t, i \mid \lambda)$, we can define the dissimilarity denoted by the distance in (3.3) as

$$d_{it}^2 = -\log f(\mathbf{x}_t, i \mid \lambda) = -\log[p_i N(\mathbf{x}_t, \mu_i, C_i)] \tag{4.1}$$

Using (2.2), we have

$$\begin{aligned} d_{it}^2 = &-\log p_i + \tfrac{1}{2}\log(2\pi)^d |C_i| \\ &+ \tfrac{1}{2}(\mathbf{x}_t - \mu_i)' C_i^{-1} (\mathbf{x}_t - \mu_i) \end{aligned} \tag{4.2}$$

An approximation of this distance was used in *entropy constrained* VQ algorithm or *generalised k-means* VQ algorithm to train codebooks in the VQ approach from the training data set $X$ [4,5]. The argument list of $J_m$ is extended using $C = \{C_1, \ldots, C_c\}$ and $p = \{p_1, \ldots, p_c\}$ and we still have

$$J_m(U, \mu, C, p) = \sum_{t=1}^{T} \sum_{i=1}^{c} u_{it}^m d_{it}^2 \tag{4.3}$$

Substituting (4.1) to (4.3) gives

$$\begin{aligned} J_m(U, \mu, C, p) = &-\sum_{t=1}^{T} \sum_{i=1}^{c} u_{it}^m \log p_i \\ &- \sum_{t=1}^{T} \sum_{i=1}^{c} u_{it}^m \log N(x_t, \mu_i, C_i) \end{aligned} \tag{4.4}$$

Minimising $J_m$ is performed by minimising each term on the right hand side of (4.4). For minimising the first term, using the Lagrange multiplier $\lambda$ [6], the following augmented objective function is maximised

$$f(p) = \sum_{t=1}^{T} \sum_{i=1}^{c} u_{it}^m \log p_i + \lambda \sum_{i=1}^{c} p_i \tag{4.5}$$

we have

$$\bar{p}_i = \frac{\sum_{t=1}^{T} u_{it}^m}{\sum_{i=1}^{c} \sum_{t=1}^{T} u_{it}^m} \tag{4.6}$$

$\bar{p}_i$ in (4.6) is defined the *fuzzy mixture weight*. The minimisation of the second term of (4.4) is obtained by setting its derivatives with respect to $\mu_i$ and $C_i$ to zero, $i = 1, \ldots, c$.

$$\sum_{t=1}^{T} u_{it}^m C_i^{-1} (\mathbf{x}_t - \mu_i) = 0 \tag{4.7}$$

$$\sum_{t=1}^{T} u_{it}^m [C_i - (\mathbf{x}_t - \mu_i)(\mathbf{x}_t - \mu_i)'] = 0 \tag{4.8}$$

To get (4.8), the following identities are used

$$\nabla_b (b' A b) = Ab + A' b, \quad \nabla_A (b' A b) = bb' \quad \text{and}$$

$$\nabla_A |A| = A^{-1} |A| \tag{4.9}$$

where $A$ and $b$ are a $d$-by-$d$ matrix and a $d$-dimensional column vector, respectively. From (4.7) and (4.8) we have

$$\bar{\mu}_i = \frac{\sum_{t=1}^{T} u_{it}^m \mathbf{x}_t}{\sum_{t=1}^{T} u_{it}^m} \tag{4.10}$$

$$\bar{C}_i = \frac{\sum_{t=1}^{T} u_{it}^m (\mathbf{x}_t - \mu_i)(\mathbf{x}_t - \mu_i)'}{\sum_{t=1}^{T} u_{it}^m} \tag{4.11}$$

where $u_{it}$ is computed using (3.4) since it is derived from minimising $J_m$ with $\{u_{it}\}$ as variables. The algorithm based on these estimation formulas could be named the *fuzzy Gaussian mixture model* (FGMM) algorithm and is stated as follows

**Algorithm 3:**

*Step 1:* Fix $c$ and $m$, $2 < c < T$, $m > 1$. Generate matrix $U$ at random satisfying (3.2)
*Step 2:* For $i = 1,\ldots, c$, compute the *c fuzzy mixture weights* $\{p_i\}$ with (4.6), the *c fuzzy mean vectors* $\{\mu_i\}$ with (4.10), the *c fuzzy covariance matrices* $\{C_i\}$ with (4.11) and the distances $d_{it}$ in (4.3). If $d_{it} = 0$ for some $t$, set $u_{it} = 1$, $u_{is} = 0$, $\forall s \neq t$
*Step 3:* Update matrix $U$ using (3.4)
*Step 4:* Stop if the decrease in the value of the fuzzy objective function $J_m$ at the current iteration relative to the value of the $J_m$ at the previous iteration is below a chosen threshold, otherwise go to step 2.

## 5. EXPERIMENTAL RESULTS

According to the theoretical considerations above, we present in this paper the results of GMM-based and FGMM-based speaker recognition experiments. The commercially available TI46 speech data corpus is used to compare these algorithms. There are 16 speakers, 8 female and 8 male, labelled f1-f8 and m1-m8, respectively. The vocabulary contains a set of ten single-word computer commands which are: *enter, erase, go, help, no, rubout, repeat, stop, start,* and *yes*. Each speaker repeated the words 10 times in a single training session, and then again twice in each of 8 later testing sessions. The corpus is sampled at 12500 samples per second and 12 bits per sample. The data were processed in 20.48 ms frames (256 samples) at a frame rate of 125 frames per second (100 sample shift). Frames were Hamming windowed and preemphasised with $\mu = 0.9$. For each frame, 46 mel-spectral bands of a width of 110 mel and 20 mel-frequency cepstral coefficients (MFCC) were determined [12]. In the training phase, 100 training tokens (10 utterances x 1 training session x 10 repetitions) of each speaker were used to train GMMs and FGMMs of 32, 64 , 128 mixtures.

Speaker identification was carried out by testing all 2560 test tokens (16 speakers x 10 utterances x 8 testing sessions x 2 repetitions) against the GMMs and the FGMMs of all 16 speakers in the database. The experimental results are as follows:

| Number of mixtures | Identification Error Rate for | |
|---|---|---|
| | GMM | FGMM |
| 32 | 22.53 % | 22.05 % |
| 64 | 18.59 % | 16.48 % |
| 128 | 14.97 % | 12.63 % |

Speaker verification in text-dependent mode with 160 tokens for each model (10 short utterances x 8 testing sessions x 2 repetitions) using the similarity normalisation method for speaker verification based on a posteriori probability proposed by Matsui and Furui [9,10] . The experimental results are as follows:

| Number of mixtures | Equal error rate for | |
|---|---|---|
| | GMM | FGMM |
| 32 | 6.45 % | 6.03 % |
| 64 | 4.89 % | 4.12 % |
| 128 | 3.75 % | 3.75 % |

## 6. CONCLUSION

In this paper, the fuzzy gaussian mixture model (FGMM) algorithm has been proposed for speaker recognition. This algorithm has been compared with the well-known GMM algorithm. Results show an error reduction for the new algorithm and show that the FGMM algorithm is applicable in speaker identification and speaker verification applications.

## 7. REFERENCES


[1] R.O. Duda and P.E. Hart (1973), "*Pattern classification and scene analysis*", John Wiley & Sons.

[2] J. Dunn (1974), "*A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Cluster*", J. Cybernetics, Vol. 3, pp. 32-57.

[3] Gustafson, D. E. and Kessel, W. (1979), "*Fuzzy clustering with a Fuzzy Covariance Matrix*", in Proc. IEEE-CDC, Vol.2 (K. S. Fu, ed.), pp. 761-766, IEEE Press, Piscataway, New Jersey.

[4] James C. Bezdek (1987), " *Pattern Recognition with Fuzzy Objective Function Algorithms*", Plenum Press, New York and London.

[5] P. Chou, T. Lookabaugh, and R. Gray (1989), "*Entropy-constrained vector quantisation*", IEEE Trans. Acoustic, Speech, and Signal Processing, vol. ASSP-37, pp. 31-42.

[6] X.D. Huang, Y. Ariki, and M.A. Jack (1990), "*Hidden Markov Models For Speech Recognition*", Edinburgh University Press.

[7] James C. Bezdek and Sankar K. Pal (1992), "*Fuzzy Models for Pattern Recognition*", IEEE Press.

[8] Reynolds, Douglas Alan. (1993), "*A Gaussian Mixture Modeling Approach To Text-Independent Speaker Identification*", PhD thesis.

[9] Tomoko Matsui and Sadaoki Furui (1994), "*A new similarity normalisation method for speaker verification based on a posteriori probability*", ESCA Workshop on Automatic Speaker Recognition, Identification and Verification, pp. 59-62.

[10] Sadaoki Furui (1994), "*An overview of speaker recognition technology*", ESCA Workshop on Automatic Speaker Recognition, Identification and Verification, pp. 1-9.

[11] Reynolds, Douglas Alan. (1995), "*Robust text-independent speaker identification using Gaussian mixture models*", IEEE Transactions on Speech and Audio Processing, Vol. 3, No. 1, January 1995.

[12] Michael Wagner (1996), "*Combined speech-recognition/speaker-verification system with modest training requirements*", Proceedings of the Sixth Australian International Conference on Speech Science and Technology, Adelaide, Australia, 1996, pp. 139-143.