Mobile phone embedded digit-recognition

Christophe Lévy[1,2]*, Georges Linarès*[1]*, Jean-François Bonastre*[1]

[1] LIA-CNRS, Avignon (France)
[2] Stepmind SA, Le Cannet (France)
{christophe.levy, georges.linares, jean-francois.bonastre}@lia.univ-avignon.fr

Abstract

Speech recognition applications are known to require a significant amount of memory. However, the targeted context of this work - mobile phone embedded speech recognition system - only authorizes less than 100kB of memory. In order to fit the memory resource, a global codebook of Gaussians is learned to derive state-dependent probability density functions. This strategy aims at storing only the transformation function parameters for each state. In this paper, two upper limits (concerning the acoustic model size) are set to 50kB and 100kB.

The proposed approches are evaluated on the French corpus VODIS (digit recognition - recorded into car with or without fan/opened window/radio - with a very low Signal/Noise Ratio). This preliminary study allows to build systems fitting the memory constraint with a DER (Digit Error Rate) around 10.9% (for model less than 100kB) which represents a DER absolute increase less than 1% compared to an HMM-based baseline system respecting the same memory constraint. Despite this increase, performance of both approaches remains comparable since the DER is still in the confident interval.

1. Introduction

The amount of services offered by the last generation of mobile phones was significantly increased compared to regular mobile phones. Nowadays, phones propose new kind of services like organizer, phone book, e-mail/fax, or games. During the same period, the mobile phone size was largely reduced (when the first generation of mobile phones measured about 13cm, today their size is about 8cm). These two evolutions raise an important question: "How could we use a mobile phone with all its services without a large keyboard ?". Voice based human-to-computer interfaces supply a friendly solution to this problem but require to embed a speech recognizer into the mobile phone.

Since the last decade, performance of Automatic Speech Recognition (ASR) systems were improved in such way that the level reached authorizes to build efficient vocal human-to-computer interfaces. Moreover, the gain (in performance) is linked to the computer evolution: a last generation computer with a lot of memory is generally required. The main problem to embed ASR in a mobile phone is the low level of resource available in this context which classically consists of a 50/100 MHz processor, a 50/100 MHz DSP, and less than 100kB of memory.

State-of-the-art speech recognition systems are mainly related to statistical methods (like Hidden Markov Model). For this kind of systems, a large training data set is required in order to reach good performance. The training data should be as close as possible to the targeted application. For mobile phone embedded speech processing, few speech corpora are available (moreover the main part of collected speech material is not directly recorded in a mobile phone which adds coding and transmission problems). In order to cope with this problem, the acoustic models are generally learned using large available corpora recorded in different contexts before being adapted to the targeted context using the smaller amount of data available.

The targeted context involves a large environment variability as clients use their mobile phone in several locations (office, car, street,...). In order to improve the speech robustness in these adverse conditions, large acoustic models (trained with enough data) and/or acoustic-model adaptation are needed. Nevertheless, mobile phone resource constraints emphasize the need of new solutions.

In this paper, we mainly focus on the memory constraints even if the proposed solutions allow a significative gain in terms of computational cost.

Two approaches are presented which are able to deal both with the few (training) data available and with the memory constraint. Both methods are based on the same idea: to learn a codebook of Gaussians and to use it to derive each state probability density function by applying a simple transformation (*cf.* figure 1). In this context, transformation parameters for each state are only needed to be stored.

In section 2, we present the corpora used in order to validate the proposed approaches. Then, a baseline HMM system corresponding to the application context is presented, for comparison, in section 3. Section 4 describes the proposed approaches. Section 5 shows some experimental results and finally, some ongoing works and conclusions are provided in section 6 and 7.

2. Corpora

In this paper, two corpora are used in order to evaluate the proposed approaches: BREF and VODIS. Both are French database collected from different acoustic environments. BREF is a clean speech corpus while VODIS signal is recorded into cars.

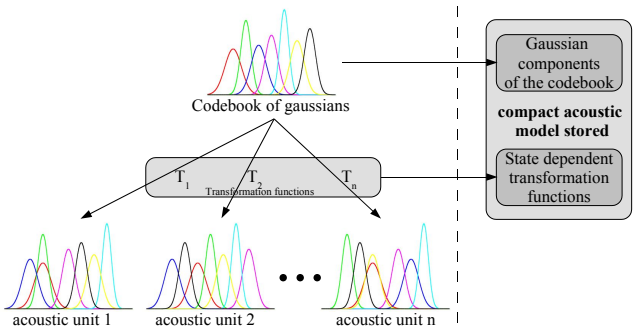


Figure 1: *General schema of the proposed approaches.*

2.1. BREF

BREF [1] is a large read-speech corpus composed of sentences selected from the French newspaper "*Le Monde*". This corpus contains about 100 hours of speech material from 120 speakers.

This corpus is used for the training phase only, not for the test phase.

2.2. VODIS

VODIS [2] is a French corpus dedicated to car embedded applications. It includes signals from 200 speakers recorded in 3 different cars. It contains a large variety of data: letters, digits, vocal commands, spelled words... Recordings are made with close-talk and far-talk microphones. The acoustic environment varies for every recording session (the window is opened or not, the radio is turned on or not, the AC is turned on or not).

These data are very close to realistic data: the Speech/Noise Ratio (SNR), estimated with the system presented in [3], is around 3.5dB[1]. The SNR is performed as follow:

$$SNR = 10 * \log_{10}\left(\frac{\sum_{i=0}^{M}(\tilde{S}_i - \tilde{N}_i)}{\sum_{i=0}^{M}\tilde{N}_i}\right)$$

where $\tilde{S}_i$ is the signal spectral amplitude and $\tilde{N}_i$ the noise spectral amplitude at frequency i. $\tilde{N}_i$ is estimated using histograms of energy [5].

We use only the subset containing the isolated digits, under the close-talk condition. It was divided into two parts:

- one for the application context adaptation (ADAPT_SET): it includes 693 digits pronounced by 40 speakers;
- one for testing (TEST_SET): composed of 2689 utterances of digits pronounced by 160 speakers.

As we performed digit recognition the evaluation measure used is the Digit Error Rate (DER).

The speakers of ADAPT_SET are different from the speakers of TEST_SET (and are also different from the BREF speakers).

3. Baseline HMM system

The baseline HMM system used for comparison is composed of 38 phonemic models, 108 emitting states (for French non-contextual model), 128 Gaussians per state and 39 PLP coefficients per frame (13 statics and first and second derivatives). The acoustic model size is about 10MB[2].

HMM is firstly trained using BREF. Then a model adaptation (using MAP [6]) is performed with ADAPT_SET.

For reducing the acoustic model size, different approaches are available like reducing the number of Gaussian components by state, decreasing the number of states in the HMM, reducing the size of acoustic coefficients or sharing Gaussians (into states or between states) [7].

For the baseline system, we propose a basic solution in order to fit the memory constraint. It consists in decreasing the number of parameters. The number of Gaussians is decreased from 128 to 4 and the order of the acoustic vector from 39 to 13 (by removing the dynamic features) which leads to a model size less than 100 kB. A second model (which requires less than 50kB) is also learned with only two Gaussians per state and 13 acoustic coefficients.

Table 1 shows a DER around 4%[3] when using the full size baseline system (128 Gaussians per state and 39 coefficients for acoustic vectors). The DER increases from 9.8% to 11.0% when the acoustic model size is fitted to the targeted context, 100kB and 50kB respectively (model size is decreased by a factor of 100 or 200).

[1] for comparison the SNR of French corpus ESTER ([4]) is around 6dB and around 15dB for the French corpus BREF ([1])

[2] for simplification, in all this work, we use "double" precision to store the values. Some techniques are known to easily reduce the model size as using "float" instead of "double" or using quantification. Since these techniques propose the same reduction factor for both the baseline HMM system and the approaches proposed in this paper, this memory reduction is ignored in this work.

[3] results are obtained with a mono-phone model. It can be noted that a tri-phone model allow a DER around 3.4% in the same experimental conditions

Table 1: *Evolution of Digit Error Rate according to the number of parameters (number of Gaussian components and the order of acoustic vectors). 2689 isolated digits are decoded for this test (TEST_SET).*

	DER	model size
128g. (39 coef.)	3.7 %	8737 kB
4g. (13 coef.)	9.8 %	93 kB
2g. (13 coef.)	11.0 %	47 kB

4. Codebook based approaches

For embedded speech recognition, the main issues could be related to memory occupation or computational cost constraints (which could be independent). In this paper, we focus on the memory occupation aspect.

In this sense, two approaches are proposed, both based on a Gaussian codebook used to derive the state dependent Probability Density Functions (PDF).

The first step (the codebook building which is common to both approaches) consists in using all the Gaussian components gathering from the baseline HMM and then reducing, if needed, the number of components by a merging approach. Gaussian merge is based on a bottom-up hierarchical clustering method following a minimum likelihood loss metrics.

For the second step (deriving the state dependent PDF), two methods are proposed:

- re-estimating only the weight parameters. Weights of all Gaussian components are re-estimated with ML criterion. The unique difference for each state dependent PDFs is the weight of Gaussian components;
- using an unique linear transformation for all the Gaussian components of the codebook. The state PDF are derived from the codebook by a linear transformation.

4.1. Weight Re-Estimate (WRE)

In this approach, for a given state, we estimate (and store) the weight of each Gaussian in the codebook only.

The Gaussian weights (w_i) are re-estimated with a Maximum Likelihood criterion[4]. The estimate function is:

$$w'_i = \frac{w_i * likelihood(fr|g_i)}{\sum_{g_j=1}^{nb_g} w_j * likelihood(fr|g_j)}$$

where $likelihood(fr|g_x)$ is the likelihood for the state-related frames (fr) given a Gaussian component g_i.

This approach allows to work with the low memory resource. Furthermore, likelihood for each component is only computed once and then the state likelihood is issued from a simple weighted combination of component likelihoods.

4.2. Unique Linear Transformation (ULT)

The method presented in [8] (LIAMAP) allows to estimate the state dependent PDF by applying a simple transformation to a codebook. This transformation (applied both on the mean and the variance) is a linear adaptation of the codebook components:

$$\mu(StateGMM) = \alpha * \mu(GaussianCodebook) + \beta$$

$$\sigma(StateGMM) = \alpha^2 * \sigma(GaussianCodebook)$$

where α (which is common for $\mu(StateGMM)$ and $\sigma(StateGMM)$) and β are given as follows.

The main idea of this adaptation (as illustrated by figure 2) is to estimate a linear transformation between two Gaussians obtained by:

1. merging the Gaussian components of the initial codebook. The final Gaussian is defined by μ and Σ, respectively the mean and the covariance matrix
2. adapting the Gaussian components of the codebook using state-specific data (using MAP) and then merging adapted Gaussians to obtain a unique Gaussian defined by $\widetilde{\mu}$ and $\widetilde{\Sigma}$.

The state-dependent Gaussian components (defined by its mean μ'_m and its covariance matrix Σ'_m) are computed as follows:

$$\mu'_m = \widetilde{\Sigma}^{1/2}\Sigma^{-1/2}(\mu_m - \mu) + \widetilde{\mu} \quad (1)$$

$$\Sigma'_m = \widetilde{\Sigma}\Sigma^{-1}\Sigma_m \quad (2)$$

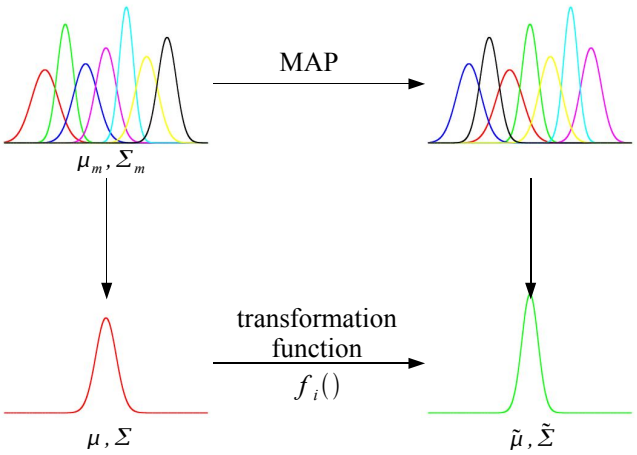
Equation 1 could be expanded as:

$$\mu'_m = \widetilde{\Sigma}^{1/2}\Sigma^{-1/2}\mu_m - \widetilde{\Sigma}^{1/2}\Sigma^{-1/2}\mu + \widetilde{\mu} \quad (3)$$

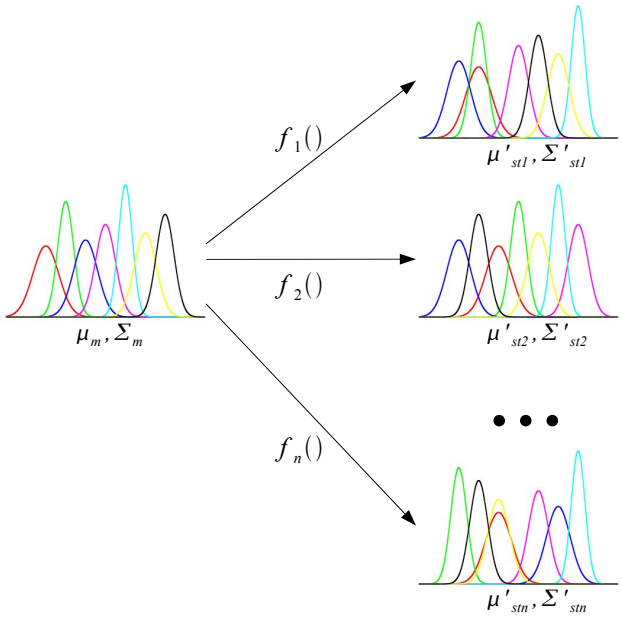
let us set

$$\alpha = \widetilde{\Sigma}^{1/2}\Sigma^{-1/2} \quad (4)$$

[4] a MAP criterion, instead of a ML criterion, could be used for weight re-estimate. In our work ML criterion is used for weight re-estimate as enough adaptation data are available.



(a) learning the transformation function



(b) applying the adaptation function

Figure 2: *LIAMAP: Method to estimate a single linear transformation for all Gaussians of a codebook. During learning phase (2(a)), one transformation per state is learned, for all Gaussians from the codebook.*

and

$$\beta = -\tilde{\Sigma}^{1/2}\Sigma^{-1/2}\mu + \tilde{\mu} \tag{5}$$

equations 1 and 2 become:

$$\mu'_m = \alpha\mu_m + \beta \tag{6}$$

and

$$\Sigma'_m = \alpha^2\Sigma_m \tag{7}$$

Equations 11 and 12 correspond to a linear adaptation function defined only by the vectors α and β (the transformation is shared by all the codebook components).

Several experiments show that saving only the linear transformation and using an equiprobable weight give worse results compared to performing first ULT and then re-estimating all weights. Nevertheless to fit the memory constraint, it was not possible to store all weights. An alternative method was used: selecting the 20 gaussian components which maximise the likelihood and then applying the ULT on this subset. The weights were then re-estimated from the subset transformed (the ML criterion was also used for this weight re-estimation).

During the test, this approach requires to apply the transformation state by state before computing the corresponding likelihood.

4.3. Evaluation of memory occupation

For each approach - HMM baseline system (eq. 8), weight re-estimate (eq. 9) and linear transformation (eq. 10) - we estimate the acoustic model size (in terms of number of parameters) as follows:

$$nbes * nbg * \underbrace{(2 * nbp + 1)}_{one\ Gaussian} \tag{8}$$

$$\underbrace{nbg * 2 * nbp}_{codebook} + \underbrace{nbes * nbg}_{pdf\ weight} \tag{9}$$

$$\underbrace{nbg * (2 * nbp + 1)}_{codebook\ with\ weight} + \underbrace{nbes * (2 * nbp + 20)}_{linear\ transf.\ and\ weight} \tag{10}$$

where nbg is the number of Gaussians, $nbes$ the number of emitting states (fixed to 108) and nbp the number of acoustic features (fixed to 13).

To respect the memory constraint of 100kB, according to equations 8, 9 and 10, the number of Gaussians is limited to 4 Gaussians per state for the baseline system (432 Gaussians totally), to 87 Gaussians for WRE method and, finally, to 257 Gaussians for ULT approach.

When the constraint is set to 50kB, the number of Gaussian components becomes 2 for baseline system, 44 for WRE approach and 34 for ULT.

5. Results

Experiments the on isolated-digit recognition task were carried out with VODIS corpus to evaluate the proposed approaches. For comparison, the same experiment was carried out using the baseline system (with the same memory occupation).

Considering the confident interval, table 2 shows similar results for both classical-HMM based and ULT based approaches. In absolute, the DER is slightly highest for the ULT method.

The WRE also performs worse than both the baseline system and the ULT system (with two model sizes). Nevertheless, this approach could allow an important gain in terms of computing resources. A unique computation for each Gaussian component is performed followed by a simple weighted sum for the state likelihood computation. As opposed, the two others approaches need a computation for each Gaussian components of each state.

Table 2: *Comparison (in DER) between baseline system and the two presented approaches. Tests are performed with 2689 utterances of isolated digits. The "conf. int." column represents the confident interval.*

(a) Results obtain with a model size limit to 100 kB

	DER	conf. int.	model size
HMM	9.8 %	1.2%	93.3 kb
WRE	12.5 %	1.2%	93.2 kB
ULT	10.9 %	1.2%	93.3 kB

(b) Results obtain with a model size limit to 50 kB

	DER	conf. int.	model size
HMM	11.0 %	1.2%	46.7 kB
WRE	14.6 %	1.2%	47.2 kB
ULT	11.7 %	1.2%	46.7 kB

The results are very preliminary and several evolutions will be proposed in the two last sections.

6. Ongoing works

The approaches presented in this paper should also be tested with higher order codebooks.

For the ULT approach, several other transformations will be tested. One could be to use a simpler transformation where only the β parameters were stored. The linear transformation becomes simply a mean adaptation:

$$\mu'_m = \mu_m + \beta \tag{11}$$

and

$$\Sigma'_m = \Sigma_m \tag{12}$$

Other ongoing works on WRE approach show that the ML criterion does not seem to be the best criterion. Other criteria could be used as:

$$w_i' = \alpha * \frac{w_i * likelihood(fr|g_i)}{\sum_{g_j}^{nb_g} w_j * likelihood(fr|g_j)} + (1-\alpha) * w_i \tag{13}$$

$$w_i' = \left((1-\alpha) * \frac{w_i * likelihood(fr|g_i)}{\sum_{g_j}^{nb_g} w_j * likelihood(fr|g_j)} + w_i \right) / (1+1-\alpha) \tag{14}$$

The first one is proposed in [9] and the second one in [10].

In this paper, a subset of the VODIS corpus (containing isolated digits) is used to perform tests, but other experiments on French clean corpus (also on isolated digit recognition task) tend to show a major improvement with alternative approaches.

7. Conclusion and perspectives

In this paper, some solutions for embedding ASR in a mobile phone were proposed. The focus was mainly on the memory occupation. This preliminary results are encouraging since the performance were comparable with the classical HMM baseline system. The baseline system (with same memory occupation and 100kB constraint) reaches a DER around 9.8%, while alternative approaches obtained a DER between 10.9% and 12.5% (for comparison, the HMM based system with full size model - 128 Gaussian components and 39 acoustic coefficients - is around 3.7%). With the 50kB constraint, performance of the ULT approach and HMM-based system are close (the DER are respectively 11.7% and 11.0% with a confident interval of 1.2% for both DER).

For further works, we will focus on the codebook training strategy. Indeed, two approaches will be compared: to learn a codebook (GMM) with all available data or to learn a classical HMM and then use Gaussian components gathered from the HMM.

regarding to the ULT approach, the use of more dependent transformations (instead of a unique one) will be studied in order to improve the transformation accuracy.

8. Acknowledgement

The authors want to thank Irina Illina and Yves Laprie for their help.

9. References

[1] L. Lamel, J. Gauvain, and L. Eskénazi, "BREF, a large vocabulary spoken corpus for french," *proceedings of EUROSPEECH*, 1991.

[2] P. Geutner, L. Arevalo, and J. Breuninger, "VODIS - voice-operated driver information systems: a usability study on advanced speech technologies for car environments," in *proceeding of ICSLP*, vol. 4, 2000, pp. 378–382.

[3] D. Istrate, C. Fredouille, S. Meignier, L. Besacier, and J. F. Bonastre, "NIST RT'05S Evaluation : Pre-processing techniques and speaker diarization on multiple microphone meetings," *2nd Joint Workshop on Multimodal Interaction and Related Machine Learning Algorithms*, july 2005.

[4] G. Gravier, J.-F. Bonastre, E. Geoffrois, S. Galliano, K. M. Tait, and K. Choukri, ""ESTER, une campagne d'évaluation des systèmes d'indexation automatique d'émissions radiophoniques en français"," *Journées d'Etude sur la Parole*, pp. 253–256, april 2004.

[5] H. Hirsh, *Estimation of noise spectrum and its applications to SNR-estimation and speech enhancement*, Technical report tr-93-012, Berkley, USA, 1993.

[6] J. Gauvain and C. Lee, "Maximum A Posteriori estimation for multi-variante gaussian mixture observations of markov chains," *IEEE transactions on speech and audio processing*, vol. 2, pp. 291–298, 1994.

[7] J. Park and H. Ko, "Compact acoustic model for embedded implementation," *proceeding of ICSLP*, 2004.

[8] D. Matrouf, O. Bellot, P. Nocera, Linarès, and J. F. Bonastre, "Structural linear model-space transformations for speaker sdaptation," *proceeding of eurospeech*, 2003.

[9] D. Reynolds, T. Quatieri, and R. Dunn, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, vol. 10, 2000.

[10] J. Bonastre, P. Morin, and J. Junqua, ""Gaussian Dynamic Warping method applied to Text-Dependent speaker detection and verification"," in *proceedings of Eurospeech*, 2003.