IDIAP RESEARCH REPORT

IDIAP

Martigny - Valais - Suisse

SYNCHRONOUS ALIGNMENT

Johnny Mariéthoz[IDIAP] Chafic Mokbel [IDIAP]

IDIAP-RR-99-06

APRIL 1999

Dalle Molle Institute for Perceptual Artificial Intelligence • P.O.Box 592 • Martigny • Valais • Switzerland

phone +41 – 27 – 721 77 11
fax +41 – 27 – 721 77 12
e-mail secretariat@idiap.ch
internet http://www.idiap.ch

Table of Contents

1. Introduction 1
2. Synchronous Alignment 2
Criteria for synchronous alignment approach 2
Decoding 4
2.1.1 Discriminative criterion 4
2.1.2 Joint likelihood criterion 5
Training 5
2.1.3 Discriminative criterion 6
2.1.4 Joint likelihood criterion 7
2.1.5 Convergence properties 8
Scoring with synchronous alignment 10
3. Experiments and results 11
Databases 11
3.1.1 SESP database 11
3.1.2 PolyVar database 12
Scoring and decision module parameters 12
Experimental results 13
3.1.3 Fool Proof for the training on PolyVar database 13
3.1.4 Scoring factor 14
3.1.5 Discriminative synchronous alignment 15
3.1.6 Joint likelihood synchronous alignment 16
3.1.7 Speaker dependent synchronous factor 17
4. Conclusions and perspectives 18
5. References 18
Appendix A – Complementary Results with Different Normalisation Methods 19
Appendix B – Polyvar Protocol 24

Table of Figures

Figure 1: Principle of the synchronous alignment approach compared to the classical approach. 2
Figure 2: Illustration how the sharing of the path can be done in two strategies: joint likelihood and discriminative likelihood ratio. 3
Figure 3: Comparison of the “fool proof” and the classical training when the number of parameters of the model is reduced (1 gauss/state) for Z-norm normalisation. 13
Figure 4a: Comparison of the “fool proof” (3 gauss/state) and the classical training (1 gauss/state) for Sum normalisation. 13
Figure 5: Influence of the scoring factor on the Clients/Impostors scores distributions. Scoring factors are 0.5 (left), 0.3 (middle) and 0.7 (right). Reference distributions relative to the classical LLR are also plotted. 15
Figure 6: Results for different discriminative synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding. 15
Figure 7: Results for different discriminative synchronous alignment factors. Synchronous alignment training is done. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding. 16
Figure 8: Results for different joint likelihood synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding. 16
Figure 9: Results for different joint likelihood synchronous alignment factors. Synchronous alignment training is performed. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding. 17
Figure 10: Results for speaker dependent synchronous factor chosen *a posteriori* (left) or *a priori* (right). 17
Figure 11: Results for different discriminative synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding. 19
Figure 12: Results for different discriminative synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding. 19
Figure 13: Results for different discriminative synchronous alignment factors. Synchronous alignment training is done. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding. 20
Figure 14: Results for different discriminative synchronous alignment factors. Synchronous alignment training is done. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding. 20
Figure 15: Results for different joint likelihood synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding. 21
Figure 16: Results for different joint likelihood synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding. 21
Figure 17: Results for different joint likelihood synchronous alignment factors. Synchronous alignment training is performed. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding. 22
Figure 18: Results for different joint likelihood synchronous alignment factors. Synchronous alignment training is performed. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding. 22
Figure 19: Results for speaker dependent synchronous factor chosen *a posteriori* (left) or *a priori* (right). 23
Figure 20: Results for speaker dependent synchronous factor chosen *a posteriori* (left) or *a priori* (right). 23

Abstract

In speaker verification, the maximum Likelihood between criterion is generally used to verify the claimed identity. This is done using two independent models, i.e. a Client model and a World model. It may be interesting to make both models share the same topology, which represent the phonetic underlying structure, and then to consider two different output distributions corresponding to the Client/World hypotheses. Based on this idea, a decoding algorithm and the corresponding training algorithm were derived. The first experiments show, on a significant telephone database, a small improvement with respect to the reference system, we can conclude that at least synchronous alignment provides equivalent results to the reference system with a reduced complexity decoding algorithm. Other important perspectives can be derived.

1. Introduction

Several applications might use a speaker verification system to secure the private information of the clients. Such systems verify the identity of a claimed client on the basis of some speech utterances. In the last years, IDIAP has been largely involved in the development of speaker verification systems. These developments are carried within several national and European projects. Important part of the work described here has been done in the context of the PICASSO[1] project. To do the verification, client and anti-client (world) models are generally computed in a training phase. These models help to discriminate between a client and the impostors regarding some acoustic realisations.

Speech signal carries different information such as the pronounced words or the speaker characteristics. It is very difficult to measure one information by discarding the others. Thus, speaker verification systems are generally classified following their degree of dependence on the pronounced text. We distinguish:

- Text dependent systems: Verification of the claimed identity is based on a predefined password (or expression or sentence). This password can be fixed or chosen by the user. The last case is also a research topic of the PICASSO project.
- Text prompted systems: To do verification the system proposes a word or a sequence of words to repeat. In this case, the system should only know the speaker's models of several units of speech (digits, syllables or phonemes…).
- Text independent systems: Verification is based on acoustic utterances independently of the underlying text. This is the most practical form of verification.

Even if the three previous classes overlap, the performance generally increases when introducing, in the speaker recognition system, more information about the underlying text, i.e. while going from text independent to text dependent systems. In this work we are particularly focusing on text dependent speaker recognition where client and world HMMs are used to model the passwords for the client and the anti-client speakers.

This combination of text information and speaker information in the acoustic speech signal motivates the work described in this document. Standard speaker recognition systems verify the claimed identity of a speaker by using two stochastic models to describe the acoustic utterances: a client model and an anti-client (world) model. The Maximum Likelihood criterion is generally used or more precisely the log-likelihood ratio is compared to a fixed threshold. Given an input utterance and one model the likelihood is not directly computed but replaced by the joint likelihood of the input observation and the corresponding optimal path[1] (Viterbi decoding). This might be interpreted that the underlying (phonetic or) text structure is first identified and then the corresponding likelihood is computed. In this document we investigate the idea of sharing underlying text structure between the client and the anti-client models. This motivates the definition of a modeling structure where the hidden components are shared between the client and anti-client models and only the output distributions differ. Besides the theoretical motivation, such structure has an important practical advantage since only one decoder is used instead of two classically.

In the following section we define the problem and the proposed solution called "synchronous alignment". A new decoder has been developed to implement this approach. A training algorithm that optimises the modelling parameters in order to satisfy the same criterion used for decoding. In the section 3 a large experimental set and the corresponding results are described. These experiments are

[1] A path is an association between the frames of the signal and the hidden component of the model (including the states and the mixture components for mixture of Gaussian distributions).

conducted on a state of the art system, the PICASSO/CAVE[2] system. Finally, the main conclusions and the principal perspectives are drawn in the section 4.

2. Synchronous Alignment

The state of the art speaker verification system used in the PICASSO project is based on HMM modelling of speech utterances corresponding to the clients' passwords. When a speaker makes an access to a secure system, the system has to take a decision on the claimed client identity. This must be done regarding the password uttered by the speaker. Since there is no analytic solution to this problem, the Bayesian approach is generally used. Two separate HMMs are used to model a client password; one for the client and the other for the anti-client (the world model). The most probable hypothesis permits to decide if the claimed identity is correct. If no *a priori* is available for the client/anti client hypotheses, the Bayesian criterion becomes equivalent to the *maximum likelihood* criterion.

Given the speaker utterance of the password, a likelihood score is computed for each client and world HMMs. This is generally done using the Viterbi algorithm. Viterbi decoding assumes that the likelihood of a HMM given an utterance is equal to the likelihood computing along the optimal path in the model. The optimal path is defined as the correspondence between the observed acoustic sequence and the hidden components of the model leading to the highest likelihood score. This optimal path corresponds physically to the segmentation of the observed utterance into states that correspond to the underlying text. Considering the two HMMs separately assumes complete independence between the models. In fact, on the most case, the topology of the both models is identical, representing the phonetic structure of the underlying text of the password. The main idea of synchronous alignment is to make the two models share the same topology and differ in the output distributions following it is or it is not the client utterance. This means that the underlying text structure is identical for both cases and the unique difference exists in the acoustic realisation following it is or not the client. This idea has been called synchronous alignment in opposite to the classical approach where even if the two models has the same topology the optimal alignment differs and can be considered as asynchronous. The synchronous alignment principle is shown in comparison with the classical approaches in the Figure 1 where H_0 represents the Client hypothesis and H_1 represents the anti-client (world) hypothesis.

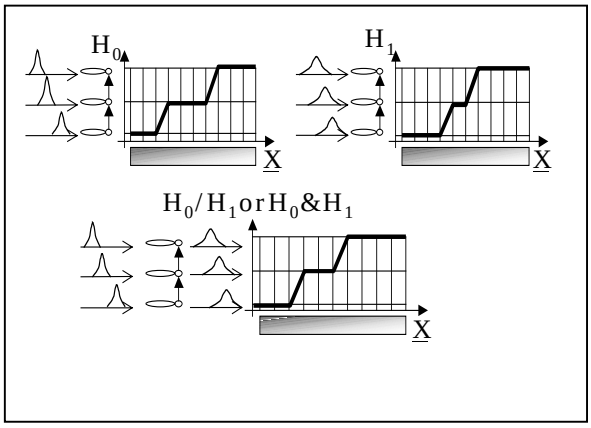

Figure 1: Principle of the synchronous alignment approach compared to the classical approach.

Even if the idea of sharing the underlying structure between the client and anti-client (world) is very simple, we need to define a criterion for decoding, to develop a specific decoder to satisfy this criterion and finally to propose a new training algorithm that is consistent with the decoding criterion. In this section we first define the two possible criteria for the decoding. Then the decoding algorithm that has been developed is described. A specific training algorithm for both criteria is derived from the classical EM algorithm. The convergence properties of such algorithm are studied and the results are presented here.

Criteria for synchronous alignment approach

If the paths are shared between the models of the two hypotheses a new scoring measure should be defined in order to determine the optimal path. Two main directions might be studied:

- The score must reflect how much the path is good for both the client and the anti-client models. A joint likelihood function can be used for this purpose. This idea assumes that the text information

is predominant in the acoustic signal. Thus, an optimal path should be optimal for both the client and the anti-client models.

- The score must reflect how much the client model is likely with respect to the anti-client model. This is a discriminant approach where the optimal path corresponds to the highest client likelihood and lowest anti-client likelihood. This criterion assumes that the speaker information is predominant in the observed signal.

The Figure 2 illustrates both ideas. First schematic log-likelihood as function of the possible paths (discrete axis) is plotted for both the client and the world models. Considering the maximum of both curves corresponds to the classical approach in speaker verification. Two other function are also shown corresponding to joint likelihood (sum of the log-likelihoods) and to the discriminant approach (difference of the log-likelihood). The maxima of these functions correspond to the optimal paths for the synchronous alignment approach in the joint likelihood and the discriminant cases respectively. As it can be seen on the Figure 2 the discriminant criterion is more sensitive and should be handled with care. The experimental results described in the following section confirm this sensitivity.

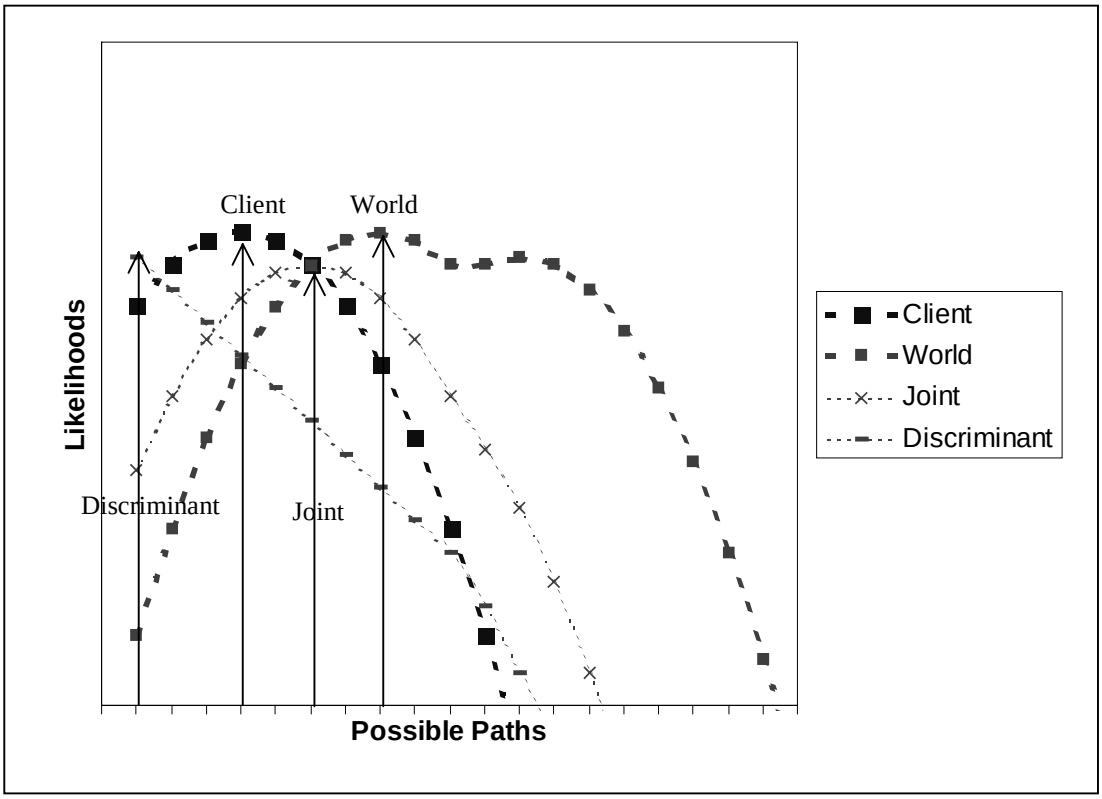

Figure 2: Illustration how the sharing of the path can be done in two strategies: joint likelihood and discriminative likelihood ratio.

Let $\underline{X}$ denote the sequence of input feature vectors of length T corresponding to the utterance pronounced by the speaker to be verified, λ denote the underlying common model structure, θ_{client} and θ_{world} denote the client's parameters and the world's parameters respectively and, finally S denote a possible path in the model. The discriminant criterion is based on a weighted likelihood ration. The optimal path can be found following this equation:

EQ 1 $$\hat{S} = \arg\max_{S} \left(\frac{p(\underline{X}, S/\theta_{client}, \lambda)^{\alpha}}{p(\underline{X}, S/\theta_{world}, \lambda)^{\beta}} \right) \quad with \quad \begin{vmatrix} \alpha + \beta = 1 & for\ 0 \le \alpha \le 1 \\ \alpha + \beta = -1 & for\ -1 \le \alpha \le 0 \end{vmatrix}$$

where α and β are the weighting factors.

It is obvious that for positive α value, the optimal path is identified such as the client's output distributions are the most likely and the world's output distributions are the less likely. These are two opposite behaviours of the algorithm. It is possible to study a criterion that combines both behaviours. If the combination is a simple linear combination, the decoding process becomes very complex. If the

combination is a simple product, the criterion becomes equivalent to the joint likelihood criterion presented in the following with arbitrary weighting factors. No further investigation in this direction was done in this study.

For the joint likelihood criterion the optimal path must be found in order to satisfy:

EQ 2 $$\hat{S} = \arg\max_{S} \left(p(\underline{X}, S/\theta_{client}, \lambda)^{\alpha} \cdot p(\underline{X}, S/\theta_{world}, \lambda)^{\beta} \right) \quad with \; \alpha + \beta = 1 \; and \; 0 \le \alpha \le 1$$

where α and β are the weighting factors.

Decoding

Here, the decoding algorithm is described. This algorithm has been developed to compute the optimal path following the two criteria described in the previous subsection 0. It is clear that we cannot maximise the EQ 1 and EQ 2 by an exhaustive search. A variant of the Viterbi decoding algorithm must be developed. This decoding issue will be discussed for both discriminative and joint likelihood criteria.

2.1.1 Discriminative criterion

For the discriminative criterion, the argument to maximise in the EQ 1 can be written:

EQ 3
$$\begin{aligned}
\frac{p(\underline{X}, S/\theta_{client}, \lambda)^{\alpha}}{p(\underline{X}, S/\theta_{world}, \lambda)^{\beta}} &= \frac{p(\underline{X}/S, \theta_{client}, \lambda)^{\alpha} . p(S/\theta_{client}, \lambda)^{\alpha}}{p(\underline{X}/S, \theta_{world}, \lambda)^{\beta} . p(S/\theta_{world}, \lambda)^{\beta}} \\
&= \frac{\prod_{t=1}^{T} p(\underline{X}_t/S_t, \theta_{client}, \lambda)^{\alpha} . a_{S_{t-1}S_t}{}^{\alpha}}{\prod_{t=1}^{T} p(\underline{X}_t/S_t, \theta_{world}, \lambda)^{\beta} . a_{S_{t-1}S_t}{}^{\beta}} \\
&= \prod_{t=1}^{T} a_{S_{t-1}S_t}{}^{\alpha-\beta} . \frac{b_{client,S_t}(\underline{X}_t)^{\alpha}}{b_{world,S_t}(\underline{X}_t)^{\beta}}
\end{aligned}$$

$$with \quad \begin{cases} \alpha + \beta = 1 & for \; 0 \le \alpha \le 1 \\ \alpha + \beta = -1 & for \; -1 \le \alpha \le 0 \end{cases}$$

where $a_{S_{t-1}S_t}$ represents the transition probability in the model and is supposed to be identical for the speaker and the world parameters and, $b_{client,S_t}()$ and $b_{world,S_t}()$ are the client's and respectively the world's output distributions relative to the state S_t.

By replacing the EQ 3 into the EQ 1, it seems that the Viterbi algorithm can be directly used for decoding by:

- taking the transition probabilities at a power of α - β,
- replacing for each frame the log-likelihood of an output distribution by the difference between the weighted log-likelihoods of the client and the world output distributions.

Since the discriminative criterion is mainly based on the idea that the predominant information in the measured features is relative to the speaker, a problem exists when decoding with a model including silence, pause and/or noise models. These parts of the signal do not include any information about any speaker and the discriminative criterion is not justified. In order to avoid such problem, we propose to first decode the signal on the world model and cut-off the parts relative to silence or other non-speech parts in the signal. Only the speech parts of the signal will be decoded using the discriminative synchronous alignment algorithm. In our experiments this procedure will be referenced as segmental "**Seg**" discriminant synchronous alignment decoding as opposed to the standard "**Std**" discriminative synchronous alignment decoding.

2.1.2 Joint likelihood criterion

For the joint client/anti-client likelihood decoding, the argument to maximise in the EQ 2 can be written:

$$
\begin{aligned}
& p(\underline{X}, S/\theta_{client},\lambda)^{\alpha} \cdot p(\underline{X}, S/\theta_{world},\lambda)^{\beta} \\
&= p(\underline{X}/S,\theta_{client},\lambda)^{\alpha}.p(S/\theta_{client},\lambda)^{\alpha} \cdot p(\underline{X}/S,\theta_{world},\lambda)^{\beta} \cdot p(S/\theta_{world},\lambda)^{\beta} \\
&= \prod_{t=1}^{T} a_{S_{t-1}S_t}{}^{\alpha}.p(\underline{X}_t/S_t,\theta_{client},\lambda)^{\alpha} \prod_{t=1}^{T} a_{S_{t-1}S_t}{}^{\beta}.p(\underline{X}_t/S_t,\theta_{world},\lambda)^{\beta} \\
&= \prod_{t=1}^{T} a_{S_{t-1}S_t}{}^{\alpha+\beta}.b_{client,S_t}(\underline{X}_t)^{\alpha}.b_{world,S_t}(\underline{X}_t)^{\beta} \\
&= \prod_{t=1}^{T} a_{S_{t-1}S_t}.b_{client,S_t}(\underline{X}_t)^{\alpha}.b_{world,S_t}(\underline{X}_t)^{\beta} \\
& \qquad\qquad with\,\alpha+\beta=1\,and\,0\le\alpha\le 1
\end{aligned}
$$

EQ 4

where the notations are the same as for the EQ 3.
When replacing the EQ 4 in the EQ 2, it appears that the classical Viterbi algorithm can be used for decoding in the joint likelihood synchronous alignment approach. The only modification consists in replacing, at each frame, the log-likelihood of an output distribution by a linear combination of two log-likelihoods. These two log-likelihoods correspond to the client and the world output distributions.

In opposite to the discriminative criterion, the non-speech parts of the models do not need any specific attention. Actually, the output distributions of the client and the world hypotheses must be identical for the non-speech parts of the model.

In summary, for both discriminative and joint likelihood criteria, the decoding can be performed using the classical Viterbi algorithm. This section shows that few modifications must be introduced in the decoder to include the synchronous alignment approach. Regarding this simple decoding process, the synchronous alignment procedure offers an important practical advantage with respect to the classical verification method since decoding using a unique model can be performed instead of two models even if the same number of output distributions is used.

Training

For synchronous alignment, the models can be trained as classically. However, this is not consistent with the decoding process. Thus, a specific training algorithm has been developed. This training algorithm permits to compute the client's parameters given some utterances of the predefined password from the client. The parameters relative to the anti-client or the world are supposed to be known and are not changed during the enrolment.
In the synchronous alignment approach, the main idea is based on the fact that the underlying Markov automaton is shared between the client and anti-client models of the password. This hypothesis is supposed to be true for all the paths and not only the optimal path. However, we are mostly interested in the optimal path. Thus a variant of the segmental "Estimate Maximise" (EM) algorithm, often called the Viterbi training algorithm, is developed. A similar development can be performed in order to define a variant of the classical EM algorithm for training the model in the synchronous alignment approach.

The training algorithm has to satisfy a predefined criterion. The same criterion used during the decoding must be employed to train the client's parameters. Two cases are thus distinguished following the decoding criterion. Let *K* be the number of available utterances from the client for the training. In the case of the discriminative synchronous alignment, the optimal client's parameters must satisfy:

EQ 5

$$\begin{cases} \hat{\theta}_{client} = \arg\max_{\theta_{client}} \prod_{k=1}^{K} \max_{S^{(k)}} \left(\frac{p(\underline{X}^{(k)}, S^{(k)} / \theta_{client}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, S^{(k)} / \theta_{world}, \lambda)^{\beta}} \right) & for\ \alpha + \beta = 1\ and\ 0 \le \alpha \le 1 \\ \hat{\theta}_{client} = \arg\min_{\theta_{client}} \prod_{k=1}^{K} \max_{S^{(k)}} \left(\frac{p(\underline{X}^{(k)}, S^{(k)} / \theta_{client}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, S^{(k)} / \theta_{world}, \lambda)^{\beta}} \right) & for\ \alpha + \beta = -1\ and\ -1 \le \alpha \le 0 \end{cases}$$

In the case of joint likelihood synchronous alignment, the optimal client's parameters must satisfy:

EQ 6

$$\hat{\theta}_{client} = \arg\max_{\theta_{client}} \prod_{k=1}^{K} \max_{S^{(k)}} \left(p(\underline{X}^{(k)}, S^{(k)} / \theta_{client}, \lambda)^{\alpha} \cdot p(\underline{X}^{(k)}, S^{(k)} / \theta_{world}, \lambda)^{\beta} \right) \quad with\ \alpha + \beta = 1\ and\ 0 \le \alpha \le 1$$

Since, only the client parameters are to be trained, the two criteria of the EQ 5 and EQ 6 are similar (but not equivalent) to the Maximum likelihood criterion generally used to train the parameters of classical hidden Markov models. The world's parameters do not influence directly the client's parameters in this enrolment process. However, they indirectly influence the estimation since they are involved in the determination of the optimal paths for each training utterance.

It is obvious that no direct analytic solution exists for the EQ 5 or the EQ 6. Actually, this is the same problem as for the classical training of HMMs. The observed utterances do not form sufficient statistics to compute the client parameters. It is a problem of incomplete data. The association between the training feature vectors and the hidden components of the model are needed to complete the data. Since an easy analytic solution can be found when data are completed, the segmental EM algorithm can be used to solve the problem. This algorithm is iterative. Each iteration the training algorithm proceeds in two different steps. The first step consists in estimating the optimal paths that complete the data given the current parameters relative to the client (Estimate step). The following step (Maximise) computes new values of the client's parameters given the estimated complete data. Some details concerning this training algorithm are given in the following for the two previous criteria. One final remark concerns the optimisation criterion of the EQ 5 when α is negative. This optimisation is formed of a minimisation following a maximisation. In this case, it cannot be proven that the segmental EM algorithm permits to decrease the discriminative function while iterations progress.

2.1.3 Discriminative criterion

At the iteration *n*, the optimal client's parameters at the preceding iteration $\hat{\theta}_{client}^{(n-1)}$ are known. The corresponding optimal paths can be obtained using the Viterbi algorithm as explained in the subsection 2.1.1. This corresponds to the estimate stage. The optimal path for the k^{th} utterance verifies:

EQ 7

$$\hat{S}^{(k)(n-1)} = \arg\max_{S^{(k)}} \left(\frac{p(\underline{X}^{(k)}, S^{(k)} / \hat{\theta}_{client}^{(n-1)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, S^{(k)} / \theta_{world}, \lambda)^{\beta}} \right) \quad with \begin{cases} \alpha + \beta = 1 & for\ 0 \le \alpha \le 1 \\ \alpha + \beta = -1 & for\ -1 \le \alpha \le 0 \end{cases}$$

Given these optimal paths, new values of the client's parameters can be obtained by maximising the likelihood ratio for positive α values and minimising it for negative α values (the convergence cannot be proved for the negative α values).

For positive α values, the re-estimation equations can be derived from the optimisation:

$$\begin{aligned}\hat{\theta}_{client}^{(n)} &= \arg\max_{\theta_{client}} \prod_{k=1}^{K} \left(\frac{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{client}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{world}, \lambda)^{\beta}} \right) \\ &= \arg\max_{\theta_{client}} \prod_{k=1}^{K} p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{client}, \lambda)^{\alpha} \\ &= \arg\max_{\theta_{client}} \prod_{k=1}^{K} p(\underline{X}^{(k)} / \hat{S}^{(k)(n-1)}, \theta_{client}, \lambda)\end{aligned}$$

EQ 8

$$for\ \alpha + \beta = 1\ and\ 0 \le \alpha \le 1$$

For negative α values, the optimisation equation is given by:

$$\begin{aligned}\hat{\theta}_{client}^{(n)} &= \arg\min_{\theta_{client}} \prod_{k=1}^{K} \left(\frac{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{client}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{world}, \lambda)^{\beta}} \right) \\ &= \arg\min_{\theta_{client}} \prod_{k=1}^{K} p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{client}, \lambda)^{\alpha} \\ &= \arg\max_{\theta_{client}} \prod_{k=1}^{K} p(\underline{X}^{(k)} / \hat{S}^{(k)(n-1)}, \theta_{client}, \lambda)\end{aligned}$$

EQ 9

$$for\ \alpha + \beta = -1\ and\ -1 \le \alpha \le 0$$

Looking to the EQ 8 and the EQ 9, the "Maximise" step is equivalent to the classical one in the HMM training. This is true for all the possible values of α even if convergence is not guaranteed for negative α. Thus, in the case of discriminative training, the re-estimation equations are the same as those of classical training with the segmental EM algorithm.

2.1.4 Joint likelihood criterion

As for the discriminative criterion, given the estimate of the client's parameters $\hat{\theta}_{client}^{(n-1)}$ at the end of iteration $n-1$ the optimal paths can be found for the training utterances. This is done using the synchronous alignment Viterbi decoding as described in subsection 2.1.2. The optimal path for the k^{th} utterance is the solution of:

$$\hat{S}^{(k)(n-1)} = \arg\max_{S} \left(p(\underline{X}^{(k)}, S^{(k)} / \hat{\theta}_{client}^{(n-1)}, \lambda)^{\alpha} \cdot p(\underline{X}^{(k)}, S^{(k)} / \theta_{world}, \lambda)^{\beta} \right)$$

EQ 10

$$with \alpha + \beta = 1 and\ 0 \le \alpha \le 1$$

Given these estimated optimal paths, new estimate of the client's parameters can be obtained in the "Maximise" step. Maximising the joint likelihood provides the re-estimation equations:

$$\begin{aligned}\hat{\theta}_{client}^{(n)} &= \arg\max_{\theta_{client}} \prod_{k=1}^{K} p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{client}, \lambda)^{\alpha} \cdot p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{world}, \lambda)^{\beta} \\ &= \arg\max_{\theta_{client}} \prod_{k=1}^{K} p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{client}, \lambda)^{\alpha} \\ &= \arg\max_{\theta_{client}} \prod_{k=1}^{K} p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{client}, \lambda)\end{aligned}$$

EQ 11

$$with \alpha + \beta = 1 and\ 0 \le \alpha \le 1$$

As for the discriminative criterion, the EQ 11 shows that the re-estimation equations are equivalent to those of classical training with segmental EM. Thus, we can conclude, that for both criteria the re-estimation equations are the same as for the classical training. The only difference with classical training of a standard client model resides in the estimate step where the optimal paths are found using the synchronous alignment Viterbi decoding algorithm.

2.1.5 Convergence properties

Asymptotic convergence properties are generally studied for the segmental EM algorithm. However, it can be proved that for the training utterances, the joint likelihood on the optimal paths increases when the number of iterations increases. This can also be shown for the training within the synchronous alignment approach.

Consider the case of the discriminative criterion. For positive α values, the EQ 8 permits to write that:

$$\prod_{k=1}^{K} \frac{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \hat{\theta}_{client}^{(n)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{world}, \lambda)^{\beta}} \geq \prod_{k=1}^{K} \frac{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{client}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{world}, \lambda)^{\beta}}$$

EQ 12

$$\Rightarrow \prod_{k=1}^{K} \frac{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \hat{\theta}_{client}^{(n)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{world}, \lambda)^{\beta}} \geq \prod_{k=1}^{K} \frac{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \hat{\theta}_{client}^{(n-1)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{world}, \lambda)^{\beta}}$$

$$for\ \alpha + \beta = 1\ and\ \ 0 \leq \alpha \leq 1$$

Considering the EQ 7 into the inequality of the EQ 12 gives:

EQ 13

$$\prod_{k=1}^{K} \frac{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \hat{\theta}_{client}^{(n)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{world}, \lambda)^{\beta}} \geq \prod_{k=1}^{K} \max_{S} \frac{p(\underline{X}^{(k)}, S / \hat{\theta}_{client}^{(n-1)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, S / \theta_{world}, \lambda)^{\beta}}$$

$$for\ \alpha + \beta = 1\ and\ \ 0 \leq \alpha \leq 1$$

Moreover, one can write:

EQ 14

$$\prod_{k=1}^{K} \max_{S} \frac{p(\underline{X}^{(k)}, S / \hat{\theta}_{client}^{(n)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, S / \theta_{world}, \lambda)^{\beta}} \geq \prod_{k=1}^{K} \frac{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \hat{\theta}_{client}^{(n)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, \hat{S}^{(k)(n-1)} / \theta_{world}, \lambda)^{\beta}}$$

$$for\ \alpha + \beta = 1\ and\ \ 0 \leq \alpha \leq 1$$

The inequalities in the EQ 13 and the EQ 14 can be combined:

EQ 15

$$\prod_{k=1}^{K} \max_{S} \frac{p(\underline{X}^{(k)}, S / \hat{\theta}_{client}^{(n)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, S / \theta_{world}, \lambda)^{\beta}} \geq \prod_{k=1}^{K} \max_{S} \frac{p(\underline{X}^{(k)}, S / \hat{\theta}_{client}^{(n-1)}, \lambda)^{\alpha}}{p(\underline{X}^{(k)}, S / \theta_{world}, \lambda)^{\beta}}$$

$$for\ \alpha + \beta = 1\ and\ \ 0 \leq \alpha \leq 1$$

The inequality in the EQ 15 shows that while iterations progress, the entity to maximise in the criterion of the EQ 5 increases. The convergence to a local maximum is thus expected.

For negative α values, the EQ 9 permits to write:

$$\prod_{k=1}^{K}\left(\frac{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\hat{\theta}_{client}^{(n)},\lambda)^{\alpha}}{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta}}\right)\leq\prod_{k=1}^{K}\left(\frac{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{client},\lambda)^{\alpha}}{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta}}\right)$$

$$\Rightarrow\prod_{k=1}^{K}\left(\frac{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\hat{\theta}_{client}^{(n)},\lambda)^{\alpha}}{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta}}\right)\leq\prod_{k=1}^{K}\left(\frac{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\hat{\theta}_{client}^{(n-1)},\lambda)^{\alpha}}{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta}}\right) \quad \textbf{EQ 16}$$

$$\Rightarrow\prod_{k=1}^{K}\left(\frac{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\hat{\theta}_{client}^{(n)},\lambda)^{\alpha}}{p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta}}\right)\leq\prod_{k=1}^{K}\max_{S}\left(\frac{p(\underline{X}^{(k)},S/\hat{\theta}_{client}^{(n-1)},\lambda)^{\alpha}}{p(\underline{X}^{(k)},S/\theta_{world},\lambda)^{\beta}}\right)$$

$$for\ \alpha+\beta=-1\ and\ -1\leq\alpha\leq 0$$

Unfortunately, we cannot prove that when the new best path is chosen at the end of iteration *n*, the inequality of the EQ 16 will hold. Thus, for negative α values we cannot confirm the convergence of the algorithm.

Now let us consider the case of the joint likelihood criterion. The EQ 11 permits to write:

$$\prod_{k=1}^{K}p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{client}^{(n)},\lambda)^{\alpha}\cdot p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta}$$

$$\geq\prod_{k=1}^{K}p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{client},\lambda)^{\alpha}\cdot p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta}$$

$$\Rightarrow\prod_{k=1}^{K}p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{client}^{(n)},\lambda)^{\alpha}\cdot p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta}$$

$$\geq\prod_{k=1}^{K}p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{client}^{(n-1)},\lambda)^{\alpha}\cdot p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta} \quad \textbf{EQ 17}$$

$$\Rightarrow\prod_{k=1}^{K}p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{client}^{(n)},\lambda)^{\alpha}\cdot p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta}$$

$$\geq\prod_{k=1}^{K}\max_{S}p(\underline{X}^{(k)},S/\theta_{client}^{(n-1)},\lambda)^{\alpha}\cdot p(\underline{X}^{(k)},S/\theta_{world},\lambda)^{\beta}$$

$$with\ \alpha+\beta=1\ and\ 0\leq\alpha\leq 1$$

Given the client's parameters after the iteration *n*, the new optimal path can be found. We can write:

$$\prod_{k=1}^{K}\max_{S}p(\underline{X}^{(k)},S/\theta_{client}^{(n)},\lambda)^{\alpha}\cdot p(\underline{X}^{(k)},S/\theta_{world},\lambda)^{\beta}$$

$$\geq\prod_{k=1}^{K}p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{client}^{(n)},\lambda)^{\alpha}\cdot p(\underline{X}^{(k)},\hat{S}^{(k)(n-1)}/\theta_{world},\lambda)^{\beta} \quad \textbf{EQ 18}$$

$$with\ \alpha+\beta=1\ and\ 0\leq\alpha\leq 1$$

Combining the inequalities in the EQ 17 and the EQ 18 it can be found:

$$\prod_{k=1}^{K} \max_{S} p(\underline{X}^{(k)}, S/\theta_{client}^{(n)}, \lambda)^{\alpha} \cdot p(\underline{X}^{(k)}, S/\theta_{world}, \lambda)^{\beta}$$

EQ 19

$$\geq \prod_{k=1}^{K} \max_{S} p(\underline{X}^{(k)}, S/\theta_{client}^{(n-1)}, \lambda)^{\alpha} \cdot p(\underline{X}^{(k)}, S/\theta_{world}, \lambda)^{\beta}$$

$$with \alpha + \beta = 1 and\ 0 \leq \alpha \leq 1$$

The inequality in the EQ 19 proves that the likelihood following the optimal path at the end of the n^{th} iteration is greater or equal to the likelihood following the optimal path at the end of the $(n\text{-}1)^{th}$. This proves that while iterations progress, the entity to maximise in the criterion of the EQ 6 increases. Thus the client's parameters converge towards a local optimum.
In summary, it has been shown in this subsection that for the joint likelihood criterion converges towards a local optimum. For the discriminative criterion this convergence is shown only for positive α values. For negative values this convergence cannot be proven.

Since local optima can be reached using the training algorithms that have been developed, these algorithms are very sensitive to the initial conditions. In our work, the client's parameters can be set initially to the corresponding values for the world model or to the values after standard training of a classical client model.

Another practical issue is relative to the training with HTK toolkit. In the HTK toolkit there is no specific care for handling unusual training problems. For example, when training with few utterances, which is the case for computing a client's model, the algorithm stops in HInit if some hidden components do not have associated feature vectors. To handle such problem, we have implemented several procedures. These procedures are called "fool proof". When a hidden component has no associated or sufficient frames within the segmental EM, the main idea is to copy for the corresponding parameters the values from the preceding iteration. This helps us to avoid several problems.

Scoring with synchronous alignment

The decision module is an important module of a speaker verification system. As mentioned previously, decision is generally taken by comparing the likelihood ratio to a predefined threshold. With discriminative synchronous alignment, the likelihood ratio can be obtained directly for $\alpha = 0.5$. This is not generally the case for different values of α or for the joint likelihood criterion. Given the optimal alignment provided at the end of the decoding process, a likelihood ratio can be recomputed with different normalisation methods. This was implemented in our approach. We generally experiment with three different normalisation: Sum, Mean-0, Z-norm. Please referee to [3] & [4] for more details on these normalisations.
The decision threshold must be generally independent from the speaker (*a priori* threshold). However, a speaker dependent threshold can be computed in order to measure the expected limits of the speaker verification algorithm. This is generally called *a posteriori* decision.
In this document, we also studied the possibility of using speaker dependent synchronous alignment factor α and the corresponding speaker dependent decision threshold. A procedure to define automatically, i.e. regarding the training data, the best speaker dependent synchronous alignment factor. These approaches and experiments are described in the following section. They permit to prove the high potential of the synchronous alignment approach when the parameters are chosen as function of the speakers.

3. Experiments and results

Experiments were conducted within the European PICASSO project on the CAVE task [1]. The different protocols and the reference system are the state of the art. Two different databases were used in our experiments: SESP database and Polyvar database. In the following the two databases are described. Then some more details on the experimental protocol are provided. Finally, the results of the different experiments conducted are given.

Databases

3.1.1 SESP database

SESP is a database collected by KPN Research. It contains telephone utterances of 24 male speakers and 24 female speakers calling with different handsets (including some calls from mobile phones) from a wide variety of places (such as restaurants, public phones and airport departure lounge). All the recordings were made between March and May 1994. A substantial proportion of the calls was made from foreign countries. In our experiments, the 21 male and 20 female speakers for whom there is sufficient speech material, are used as clients.

The speech material under focus in this paper is “Scope” (telephone calling-card) numbers; sequences of 14 digits uttered in a more or less continuous fashion. A full session contains 2 utterances of such items. For each speaker, speech recorded in 4 distinct sessions was selected as enrolment material. The other sessions were considered as test sessions. For the World model, we used a small subset of the Dutch PolyPhone database, corresponding to 24 male and 24 female speakers, and consisting of 6 sequences of digits, the length of which ranges from 4 to 16. All speakers are distinct from SESP speakers.

No obvious factor makes the SESP data significantly different from those that could be expected from a field test data collection, except for the lack of intentional impostor attempts. Tests were carried out on a single utterance of the card number. Each trial consisted of 1658 genuine accesses and 1016 impostor attempts.

A state of the art reference system for the SESP database

As mentioned previously, all the experiments presented in this document are conducted using the PICASSO/CAVE speaker verification system. For the PICASSO project this system is called PICASOFT. Each experiment is represented by an alphanumeric code. The code describes the different parameters of the system. The reference experiment for the SESP database is presented by the code:
s11231pWF1_REF:HMM_LR:JMLF:DW:2:3:D:G:wlpcc16.lin:WMP:1.0:2:3:1.3.4:6.8

The Table 1 describes the each term in the preceding code and gives a complete description of the reference experiment on the SEP database.

Fields	**Description**
s11231pWF1_REF	Experiment names for scope card number
HMM_LR	Hidden Markov Model with left right topology
JMLF	JPB version of HVite => A traceback permits to obtain the scores frame by frame to enable the use of several normalisations.
DW	Text dependent using a segmentation at word level
2	Number of states per phoneme for the client models
3	Number of Gaussian mixtures for each state, for the client model
D	Diagonal covariance type
G	Name of the training set here 8 occurrences
wlpcc16.lin	Parameterisation name: 16 LPCC + energy + delta +delta delta
WMP	World model trained with data from the Polyphone database
1.0	Variance flooring factor
2	Number of states per phoneme for the World models
3	Number of Gaussian mixtures for each state, for the World model
1.3.4	Normalisation 1:SUM 2:MEAN-0 3:Z-NORM
6.8	Set of population used for the different steps for the experiment: 6:EXTERNAL(set1),DEVELOPPEMENT(set1),SCORING(set1) 8:EXTERNAL(set3),DEVELOPPEMENT(set1),SCORING(set1) see section 0

Table 1 Description of the SESP reference system

3.1.2 PolyVar database

PolyVar is a database collected by IDIAP. It contains telephone utterances of 143 speakers (85 male speakers and 58 female speakers). Each speaker recorded between 1 and 229 sessions for a total of 3600. The recording are made from the office or the home. The language is the Swiss French. Only a part of this database was used for the experiments described in this document: 17 command words considered as different passwords.
The database was split into four sets. Two sets of 19 speakers (12M/7F) represent the possible clients. One set of 33 speakers (17M/16F) defines the pseudo-impostors. A fourth set to estimate the world model is formed of 56 speakers (28M/28F). For the clients, the first 5th sessions are reserved for the training. The test accesses are chosen uniformly from the remaining clients' data. There are about 18000 test accesses and about 9700 test accesses for the pseudo-impostors. A complete descriptions of the Polyvar protocol is given the Appendix B.

A state of the art reference system for the PolyVar database
As for the SESP database, the PICASOFT was used to build a state of the art reference system. This reference system is also designated by a code:
i41123112345p_CJ:HMM_LR:JMLF:DW:2:1:D:12345:wlpcc16.lin:WMP:1.0:2:1:1.3.4:6.8
The Table 2 explains the different fields in the code of the reference system and thus provides a complete description of the reference system.

Fields	**Description**
i41123112345p_CJ	Experiment name: i for info Martigny simple command words
HMM_LR	Hidden Markov Model with a left right topology
JMLF	JPB version of HVite =>The scores are computed frame by frame using a traceback
DW	Text dependent using a segmentation at word level
2	Number of states per phoneme for the client models
3	Number of Gaussian mixtures for each state, for the client model
D	Diagonal covariance type
12345	Name of the training set here 5 occurrences
wlpcc16.lin	Parameterisation name: 16 LPCC + energy + delta +delta delta
WMP	World model trained with data from the Polyphone database
1.0	Variance flooring factor
2	Number of states per phoneme for the World models
3	Number of Gaussian mixtures for each state, for the World model
1.3.4	Normalisation: 1:SUM 2:MEAN-0 3:Z-NORM
6.8	Set of population used for the different steps for the experiment: 6:EXTERNAL(set1),DEVELOPPEMENT(set1),SCORING(set1) 8:EXTERNAL(set3),DEVELOPPEMENT(set1),SCORING(set1) see section 0

Table 2 Description of the PolyVar reference system

Scoring and decision module parameters

The log-likelihood ratios between the client and the world models were computed frame by frame. As noted in the subsection 0, three different normalisation were used in our the experiments:

- SUM : sum of the partial scores for each sequence
- MEAN-0 : mean of the non-zero partial scores for each sequence
- MEAN-0 Z-NORM : z-norm of non-zero mean of the partial scores for each sequence

For the mean 0 z-norm, a distribution of the impostor's scores is generally estimated on the external population, typically "**set3**" (pseudo-impostors access) while the "**set1**" is used as client population.

Given the measured score, there are tree different methods to compute the threshold implemented on the reference system:

- Speaker dependant threshold a posteriori (ENST method)
- Speaker independent threshold a posteriori (JMLF SOFT)
- Speaker independent threshold a priori (JMLF HARD). The threshold is fixed at “0”.

Experimental results

In this section, a large set of experimental results will be presented leading to a better understanding of the synchronous alignment approach. The performance of speaker verification system on a database is shown as the false rejection rate function of the false acceptance rate. To draw such functions, the decision threshold has been varied.

3.1.3 Fool Proof for the training on PolyVar database

The first set of experiments concerns the “fool proof” method. As described in the section 0, the training with the HTK tools may not be completed if no sufficient training data is available to estimate each hidden parameter. Actually, the Hinit implements a segmental EM algorithm and stops execution if a Gaussian mixture has no feature vectors associated after the “Estimate” step. In the previous CAVE project, the adopted solution consists in replacing for the unestimated client’s digit the corresponding world model. This is not useful for the Polyvar database since the password is formed of a single word and not a sequence of digits. The first approach investigated in the PICASSO project consists in reducing the number of Gaussian components in the mixtures. This approach has limited performances. IDIAP proposed to use the “fool proof” method described in the section 0.
First it was verified that the “fool proof” method does not deteriorate the results for a model with reduced number of parameters that can be trained with classical HTK tools. The results of the experiments conducted on the Polyvar database are shown in the Figure 3. Looking to this figure proves that the “fool proof” permits to obtain similar models than classical training for models that have sufficiently small number of parameters to be trained with the available number of utterances.

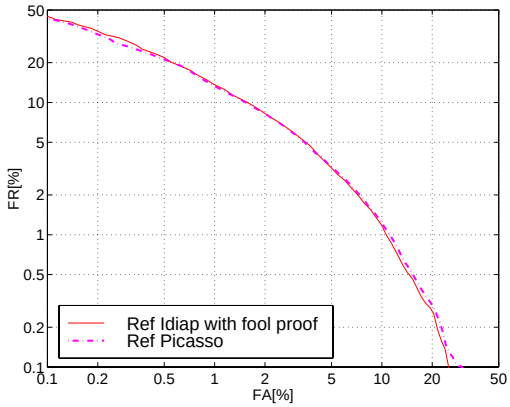

Figure 3: Comparison of the “fool proof” and the classical training when the number of parameters of the model is reduced (1 gauss/state) for Z-norm normalisation.

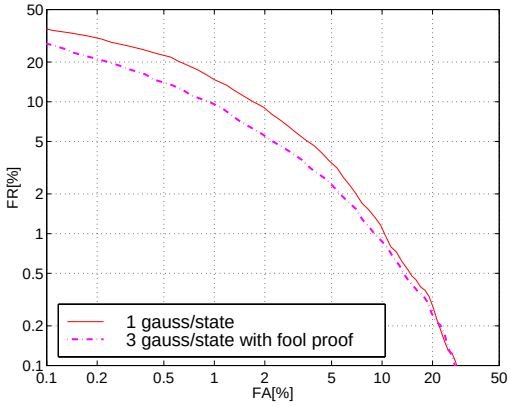

Figure 4a: Comparison of the “fool proof” (3 gauss/state) and the classical training (1 gauss/state) for Sum normalisation.

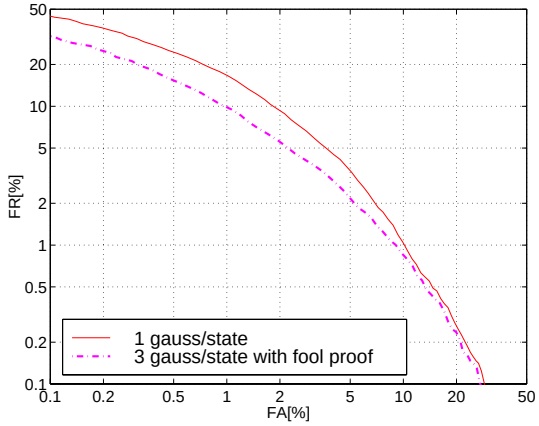

Figure 4b: Comparison of the "fool proof" (3 gauss/state) and the classical training (1 gauss/state) for Mean-0 normalisation.

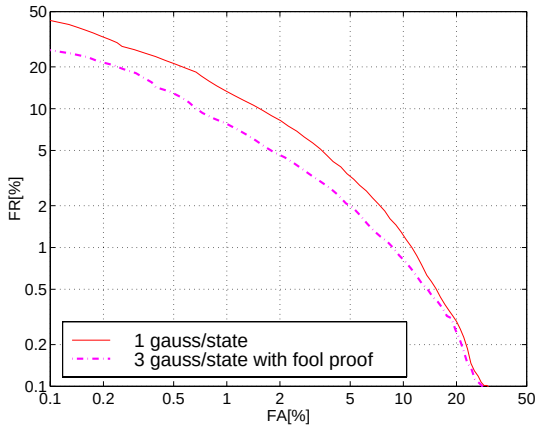

Figure 4c: Comparison of the "fool proof" (3 gauss/state) and the classical training (1 gauss/state) for Z-norm normalisation.

The Figure 4 shows the results when a standard PICASSO model was trained with the help of the "fool proof" method in comparison with the reduced model trained classically. The results are shown for the three normalisations. It can be seen that the "fool proof" approach is very helpful since it permits to obtain rich models even for few training utterances and thus to get very encouraging results compared to the classical training.

3.1.4 Scoring factor

As described in the subsection 0 two different factors should be used when computing the final score with the synchronous alignment approach. The first factor corresponds to the determination of the optimal path; this is the synchronous alignment factor. The second factor permits to compute the final score as a weighted difference between the log-likelihoods of the client and world models; this is the scoring factor. Several experiments have been conducted in order to find the optimal value of the scoring factor. All these experiments have the same conclusion that this factor should be fixed to 0.5, i.e. computing a log-likelihood ratio (LLR) between the client and the world models along the optimal synchronous alignment path. To better understand this result, we plot in the Figure 5 the distributions of the weighted LLR for different values of the scoring factor and that for both clients and impostors data. For a scoring factor of 0.5 the distributions of the weighted LLR are equivalent to the classical LLR distributions with a scaling of 0.5 of their values. For the other scoring values, we observe a shift of the LLR distributions and an increase of the overlapping surface (error surface) between the clients and impostors distributions. Given our experimental results and the justification presented here, only experimental results corresponding to a scoring factor of 0.5 are presented in the following.

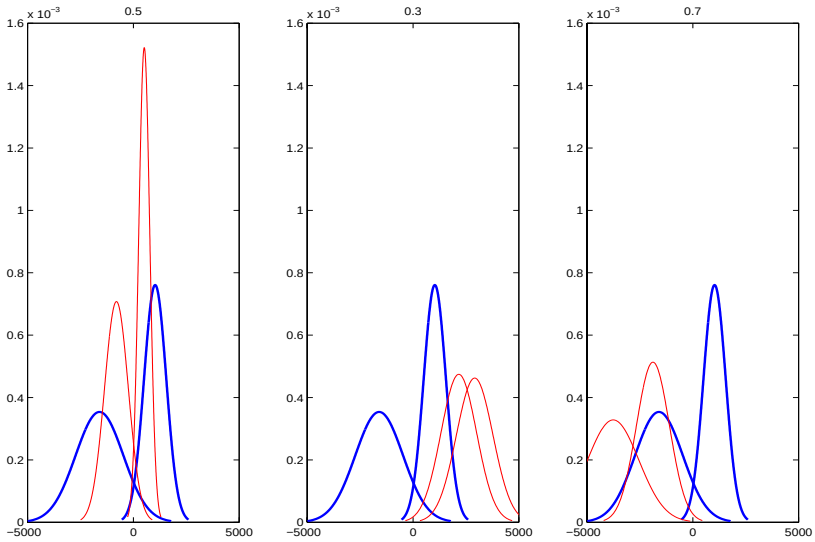

Figure 5: Influence of the scoring factor on the Clients/Impostors scores distributions. Scoring factors are 0.5 (left), 0.3 (middle) and 0.7 (right). Reference distributions relative to the classical LLR are also plotted.

3.1.5 Discriminative synchronous alignment

Several experiments were conducted on the SESP database in order to measure the performance of the synchronous alignment with the discriminative criterion. Only the results for the Z-norm are presented here. For the results with other normalisations please refer to the Appendix A.

The first experiments were conducted on clients models trained classically. The Figure 6 presents the performances for a discriminative synchronous decoding with different synchronous factors. Two sets of results are given following whether a segmental decoding is applied or not. No improvement was obtained with the discriminative synchronous decoding. Generally, segmental decoding provides better results as expected. The best results were obtained with a synchronous factor of 0 ($\alpha = 0$ and $\beta = -1$ in the EQ 1), i.e. optimal path is one corresponding to the world model. Equivalent conclusions can be drawn for the other normalisation methods.

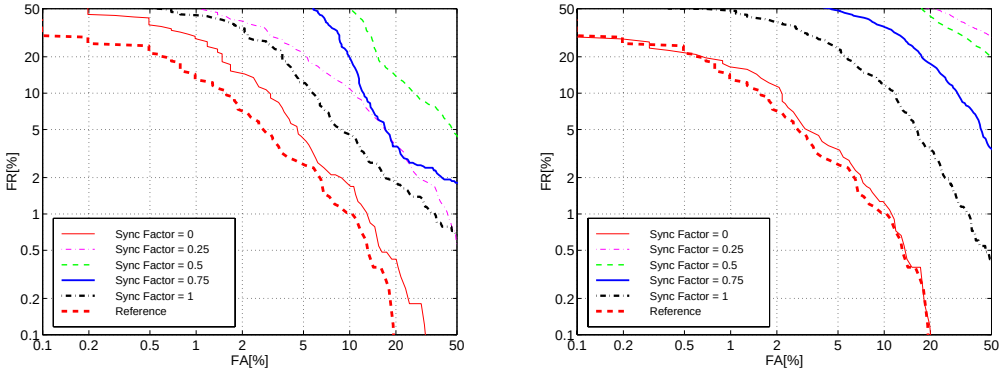

Figure 6: Results for different discriminative synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding.

In the Figure 7 results are provided in the case were the clients' models were trained with the synchronous alignment algorithm as described in the subsection 2.1.3. These results are better then those corresponding to classical training (Figure 6) even if they remain worse than those of the reference system. For world model alignment (synchronous factor = 0) the performances are equivalent to the reference performances. In the case of world model alignment, no improvement can be observed with segmental decoding.

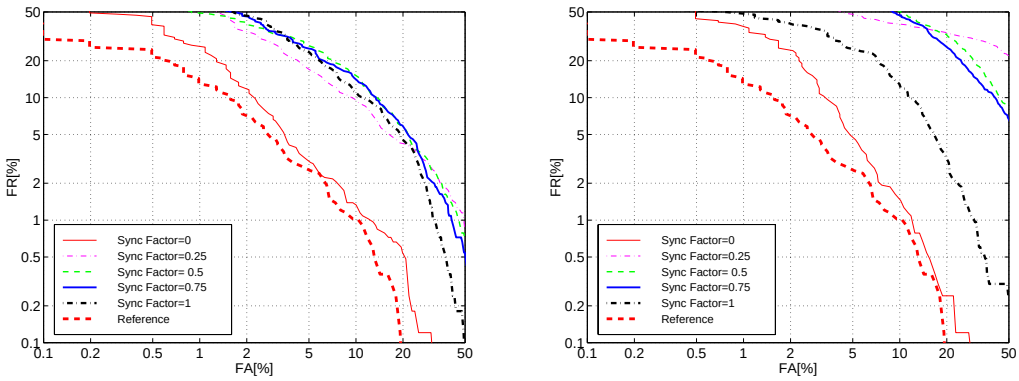

Figure 7: Results for different discriminative synchronous alignment factors. Synchronous alignment training is done. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding.

3.1.6 Joint likelihood synchronous alignment

Experiments were also conducted on the SESP database in order to measure the performance of the synchronous alignment with the joint likelihood criterion. Only the results for the Z-norm are presented here. For the results with other normalisation methods please refer to the .

The first experiments were conducted on clients models trained classically. The Figure 8 presents the performances for joint likelihood synchronous decoding with different synchronous factors. Two sets of results are given following whether a segmental decoding is applied or not. No improvement was obtained with the synchronous decoding. However, the results are much better than those obtained with the discriminative criterion. This tends to prove that it is not easy to directly discriminate between the speakers on the basis of the acoustic observations. It seems that the underlying text is the predominant information in the speech signal. Since the influence of silence models is negligible for the joint likelihood criterion, no significant improvement was observed with the segmental decoding. The best results were obtained with a synchronous factor of 0.25 (α = 0.25 in the EQ 2) showing that the synchronous alignment approach is promising. Equivalent conclusions can be drawn for the other normalisation methods.

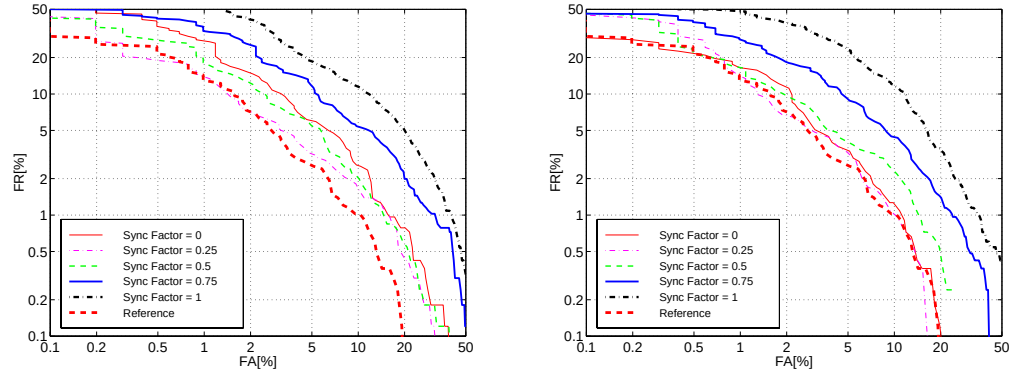

Figure 8: Results for different joint likelihood synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding.

In the Figure 9 results are provided in the case were the clients' models were trained with the synchronous alignment algorithm as described in the subsection 2.1.4. These results are better then those corresponding to classical training (Figure 8) even if they remain worse than those of the reference system. However, results equivalents to the reference system were obtained for a range of synchronous factor values around 0.25. Regarding all the normalisation methods, the best "Equal Error Rate" (EER) was obtained with the joint likelihood synchronous alignment method even if the observed improvement is not significant. In summary, the synchronous alignment approach provides on the SESP database equivalent results as the reference system when the joint likelihood criterion is used. This has at least the advantage of a simpler decoding process. Synchronous decoding consists in finding the best path on a single model instead of a couple of models classically.

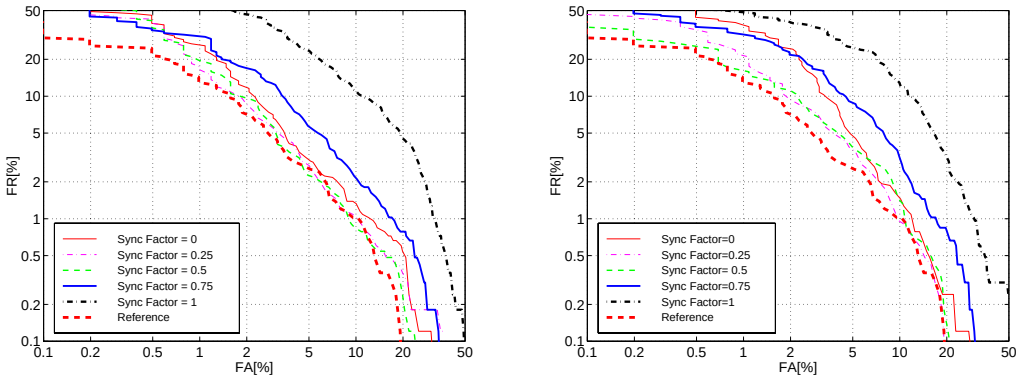

Figure 9: Results for different joint likelihood synchronous alignment factors. Synchronous alignment training is performed. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding.

3.1.7 Speaker dependent synchronous factor

In order to study the dependence on the speaker of the synchronous factor, several experiments have been conducted. We are mainly interested in the case of joint likelihood criterion. For a given client, the speaker dependent synchronous factor was chosen, in a set of 5 values, to give the less EER either on a development set (*a priori*) or on the test set (*a posteriori*). In order to plot the mean performance curves, a score corresponding to a given synchronous factor was normalised by a value corresponding to the threshold of the EER. The results obtained for the Z-norm method are shown in the Figure 10. For the other normalisation methods the results are given in the Appendix A. Looking to those results, it can be shown that a speaker dependent synchronous factor permits to improve the reference system results. However, this study must be further deepened in order to get more reliable conclusions.

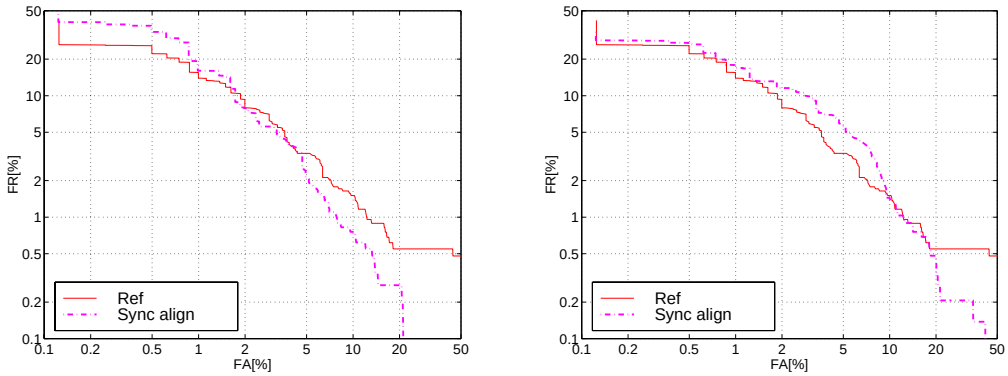

Figure 10: Results for speaker dependent synchronous factor chosen *a posteriori* (left) or *a priori* (right).

4. Conclusions and perspectives

A speech signal carries two main informations; the underlying text and the speaker characteristics. Classical speaker recognition systems have better performance when they explicitly consider this combination in the modelling process. Performance increases when going from text independent speaker recognition towards text dependent speaker recognition. This motivates the development of a new modelling structure called synchronous alignment. This approach is detailed in this document. Classical text dependent speaker recognition systems make use of two stochastic models to describe a password. The first model is speaker dependent and the second one is speaker independent and often called world model. The system decide on the identity of a claimed speaker following in how extent the client model is more likely to produce the observed utterance than the world model. To compute the likelihood of each model, a Viterbi algorithm is generally used providing two separate optimal paths. In this document the proposed modelling structure, synchronous alignment, considers that a unique underlying structure exists which corresponds to a single sequence of states whether it is the client or not that produce the observed password.

Within the synchronous alignment approach, we define two different criteria. Discriminative criterion assumes that the speaker information is predominant in the signal and search for an optimal path that maximises a weighted likelihood ratio between the client and the world hypotheses. In opposite the joint likelihood criterion searches for an optimal path that maximises the joint likelihood of both hypotheses assuming that the underlying text is the predominant information in the signal.

We also derive a decoding algorithm and a training algorithm for both criteria. These algorithms are described in this document. We showed that the training algorithm converges to a local optimum for most of the cases.

These algorithms were implemented within the HTK toolkit. They have been experimented on the databases of the PICASSO project. They have been compared to state of the art speaker verification reference system. The results show that equivalent results can be obtained with the joint likelihood criterion. This offers the advantage of a cheaper decoding algorithm since we are only obliged to decode on a single model. Another important results of this work is that the discriminative criterion has limited performance. This shows that the predominant information in the speech signal is indeed the underlying text. This opens the door for an application of this approach in the speech recognition. Actually, one can define an equivalent procedure where, for each speech unit, the world model represents the hypothesis that this unit was not pronounced. The synchronous alignment decoding and training algorithms can be used in this context.

In this work we also report the possibility of defining a speaker dependent synchronous factor. This idea was studied. Some preliminary results showed that this approach might be advantageous.

5. References

[1]Frédéric BIMBOT, Mats BLOMBERG, Louis BOVES, Gérard CHOLLET, Cédric JABOULET, Johan KOOLWAAIJ, Johan LINDBERG, Johnny MARIETHOZ, Chafic MOKBEL, "Robust approaches to speaker verification on the telephone : an overview of the PICASSO project activities. ", EUROSPEECH, 1999.

[2] F. Bimbot, H.P. Hutter, C. Jaboulet, J. Koolwaaij, J. Lindberg & J.B. Pierrot, "Speaker verification in the telephone network: research activities in the CAVE project," EuroSpeech, pp. 971-974, 1997.

[3] G. Gravier and G. Chollet, "Comparison of Normalisation Techniques for Speaker Verification", "RLA2C pp. 97-100, 1998.

[4] Kung-Pu Li and Jack E. Porter, "Normalisations and Selection of Speech Segments for Speaker Recognition Scoring", ICASSP, pp. 595-597, 1988.

Appendix A – Complementary Results with Different Normalisation Methods

In this section complementary results on the SESP database are presented. These results were obtained for two normalisation methods: Mean-0, and Sum.

A-1: Discriminative Synchronous Alignment with Classical Training:

SUM normalisation:

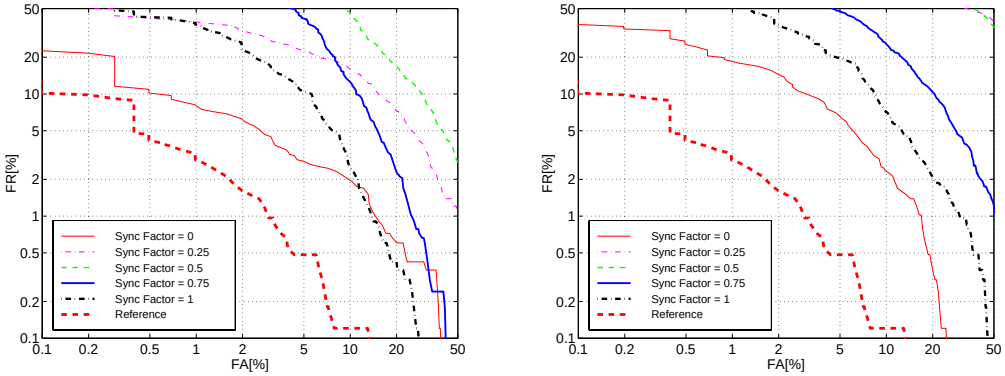

Figure 11: Results for different discriminative synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding.

MEAN-0 normalisation:

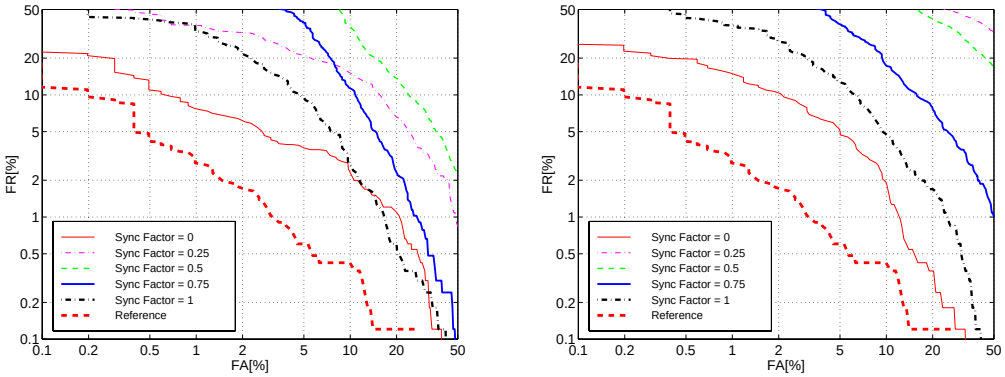

Figure 12: Results for different discriminative synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding.

A-2: Discriminative Synchronous Alignment with Synchronous Alignment Training:

SUM normalisation:

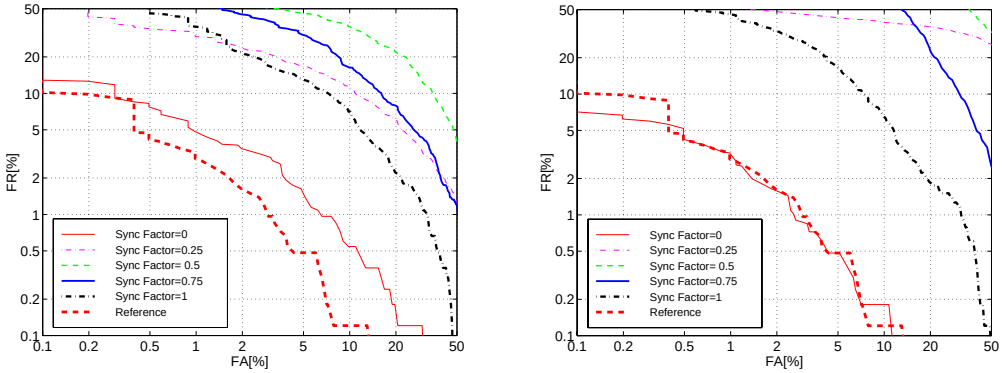

Figure 13: Results for different discriminative synchronous alignment factors. Synchronous alignment training is done. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding.

MEAN-0 normalisation:

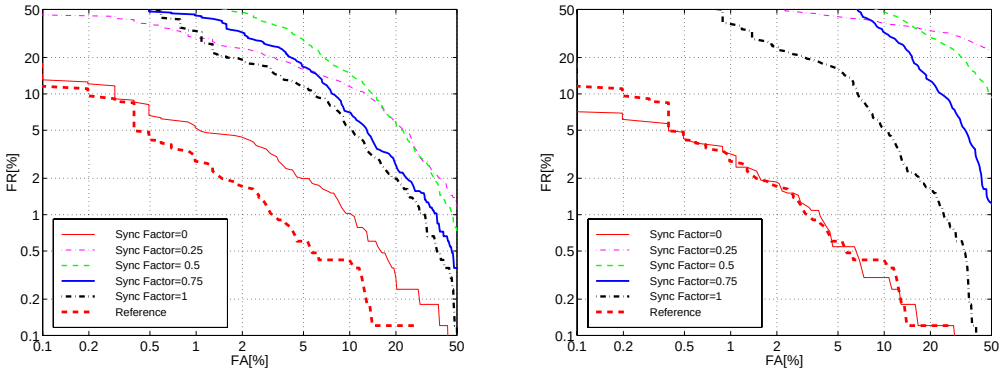

Figure 14: Results for different discriminative synchronous alignment factors. Synchronous alignment training is done. The left figure corresponds to segmental decoding avoiding silence discriminative decoding and the right figure to non-segmental decoding.

A-3: Joint Likelihood Synchronous Alignment with Classical Training:

SUM normalisation:

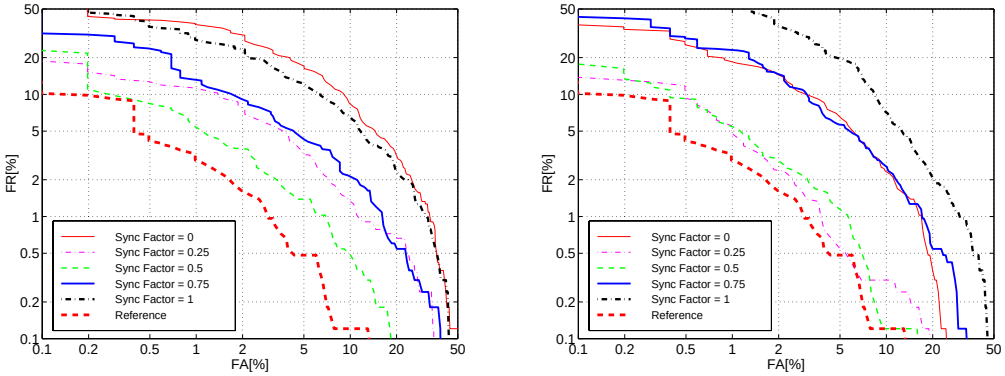

Figure 15: Results for different joint likelihood synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding.

MEAN-0 normalisation:

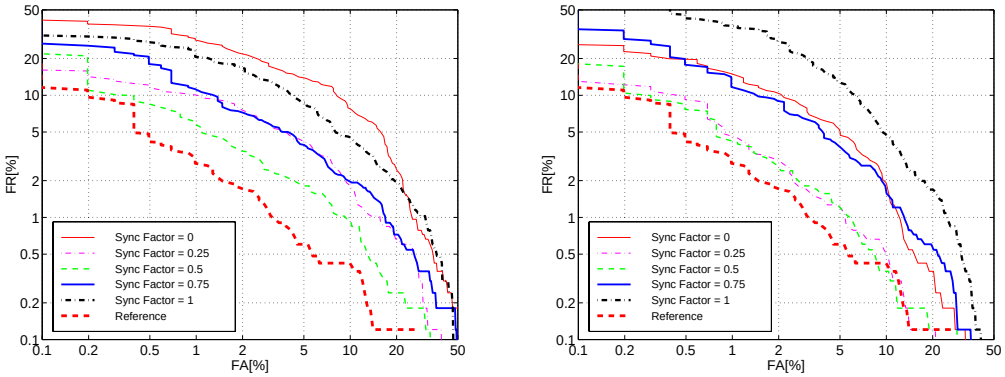

Figure 16: Results for different joint likelihood synchronous alignment factors. The training is done classically. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding.

A-4: Joint Likelihood Synchronous Alignment with Synchronous Alignment Training:

SUM normalisation:

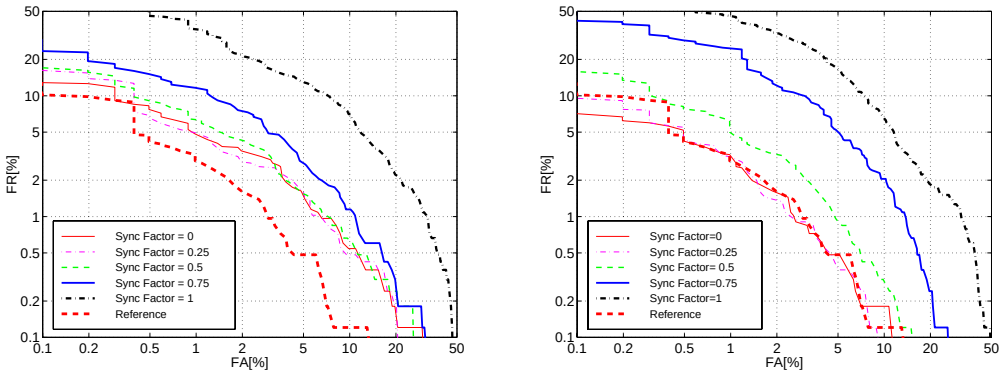

Figure 17: Results for different joint likelihood synchronous alignment factors. Synchronous alignment training is performed. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding.

MEAN-0 normalisation:

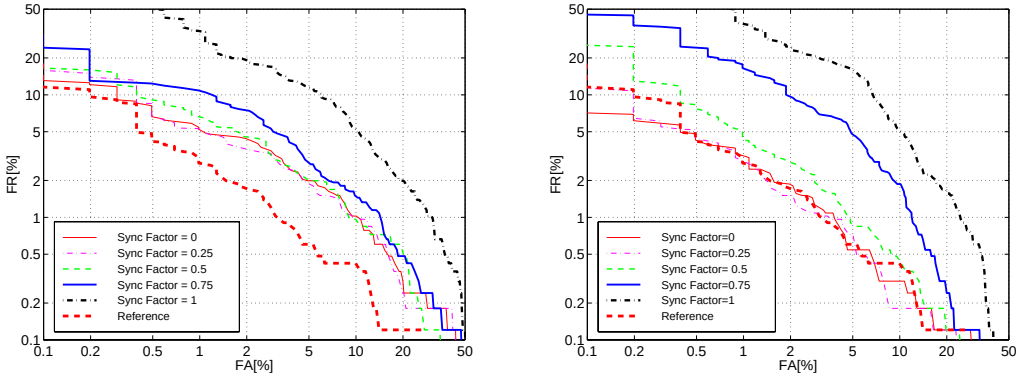

Figure 18: Results for different joint likelihood synchronous alignment factors. Synchronous alignment training is performed. The left figure corresponds to segmental decoding and the right figure to non-segmental decoding.

A-4: Speaker Dependent Synchronous Factor:

SUM normalisation:

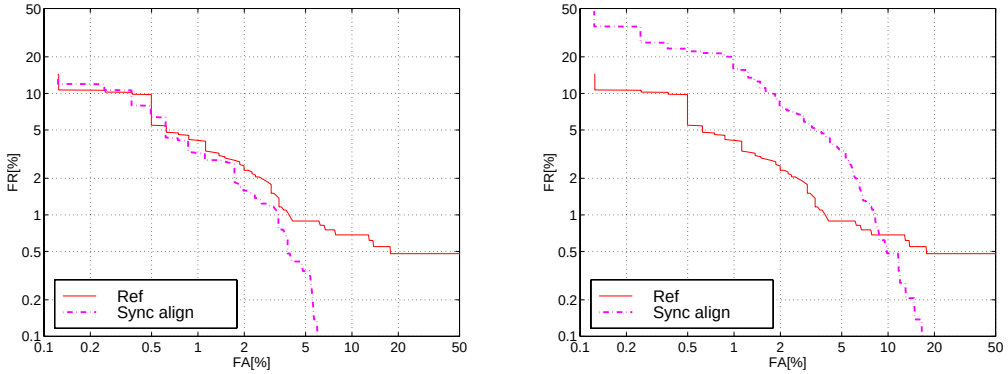

Figure 19: Results for speaker dependent synchronous factor chosen *a posteriori* (left) or *a priori* (right).

MEAN-0 normalisation:

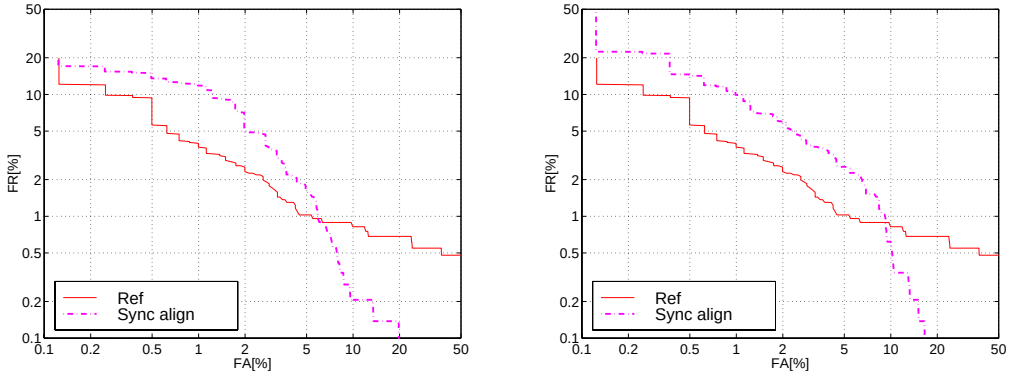

Figure 20: Results for speaker dependent synchronous factor chosen *a posteriori* (left) or *a priori* (right).

Appendix B – Polyvar Protocol

Polyvar database has been recorded by IDIAP. This is a speaker dependent telephone database. The language is the Swiss French. About 143 speakers called from there office or from there home to record the database corpus. Each speaker recorded between 1 and 229 sessions (telephone calls). A subset of the corpus was chosen for the Picasso project. This corresponds to:

- 17 tourist application words (InfoMartigny)
- 1 identification number
- 1 16-digit credit card number
- 1 sequence of 6 single digits including the hash (#) and the star (*) symbols

annulation	casino
cinéma	concert
corso	exposition
galerie du Manoir	Giannada
guide	Louis Moret
manifestation	message
mode d'emploi	musée
précédent	quitter
suivant	

Table 3: list of the command words

Speaker and recording condition
Among the 143 speakers, we have 85 male speakers and 58 female speakers. The number of sessions varies for each speaker. For the recording of the calls, two kinds of platforms were used. An analog recording board was used for almost all the different sessions. A small part of the database was recorded on a digital ISDN platform. For each file, a NIST header contains information about the recording conditions. The files were recorded in A-law format.

Annotation and transcription
Human listeners transcribed the calls. The orthographic transcription of the recorded calls was verified at IDIAP. For further processing, the calls were converted to 16 bit linear PCM. We developed a tool called ANNOTATOR for the purpose of verifying and correcting the annotation of utterances. The ANNOTATOR interface works under SunOS or Solaris on Sun workstations with audio equipment. Moreover, the annotation tool requires an installed version of Xwaves from Entropics and the public domain package TCL-TK.
A small number of natives Swiss French speaking persons were trained on the ANNOTATOR tool to perform transcription of the collected calls throughout the project. This decision helped not only to minimise the learning period, but also to guarantee a high degree of uniformity in the annotation style. The annotation persons worked only half-time on this task in order to avoid mistakes due to fatigue. Finally, the processed calls were stored on the CD-ROMs at hand.

Directory structure
The directory structure uses a shallow directory nesting with contiguous numbers to identify the individual sub-directories and call directories. The following directory structure is defined:

- sfpv\<speaker>\<session>

where:

- <speaker> Defined as: m<nn> or f<nn>
  Where:
  - <nn> is a progressive number for the different speakers
- <session> Defined as: <nn>
  Where:
  - <nn> is a regressive number in the range 00-99 (ending at 00). For speaker who has more than one hundred sessions, alphanumeric numbers are used: a0,a1 ... a9, b0,b1.

File nomenclature

File names follow the ISO 9660 file name conventions (8 plus 3 characters) according to the main CDROM standard. The following template is used:

- S SP NN CCC . alw

For the Swiss French PolyVar, we have the following assignment:

- alw: Nist format, with a-law speech format
- S (f/m): Sex identification Male/Female
- SP (00-99): Speaker identification
- NN (00-z9): Recording session progressive number (00-z9)
- CCC (000-zzz): Item identifier

As it is useful for users to clearly identify the speech file contents by looking at the filename, we have specified an Item identifier.

Group Identifier	5.1.1.1.1 Type of content
c	isolated digits
w	application words

Table 4: description of the group identifier

Group Identifier	**Group counter**	**Item content**
c	01	4 digit id/sheet number
c	02	16 digit credit card number
c	03	1 sequence of 6 digits
w	00-16	17 touristic application words (about Martigny)

Table 5: description of the items

The final specification for the Swiss French PolyVar recordings is as follow, where <nn> represent the session number, <sp> the speaker number and m/f for male/female:

- 3 connected or isolated digits

1 uttered sheet id number (prompted):

- m/f<sp><nn>c01.alw

16-digit credit card number (prompted):

- m/f<sp><nn>c02.alw

1 sequence of 6 single digits:

- m/f<sp><nn>c03.alw

17 tourists application words (prompted: the number of the word **don't** correspond to a specific word):

m/f<sp><nn>w00.alw
m/f<sp><nn>w01.alw
m/f<sp><nn>w02.alw
m/f<sp><nn>w03.alw
m/f<sp><nn>w04.alw
m/f<sp><nn>w05.alw
m/f<sp><nn>w06.alw
m/f<sp><nn>w07.alw
m/f<sp><nn>w08.alw
m/f<sp><nn>w09.alw
m/f<sp><nn>w10.alw
m/f<sp><nn>w11.alw
m/f<sp><nn>w12.alw
m/f<sp><nn>w13.alw
m/f<sp><nn>w14.alw
m/f<sp><nn>w15.alw
m/f<sp><nn>w16.alw

File format specifications
For final storage, the processed and annotated calls were stored in A-LAW format on CD-ROM. Each item on the prompt sheets was stored in a separate file with a NIST header, in which the orthographic transcription of the utterance can be found, as well.
This is an example of a NIST header for one item of a call:

database_id Swiss_French_PolyVar
database_version 1.0
recording_site Idiap
recording_board Analog
recording_date 30/MAR/95
recording_time 12:06:00
sheet_id 5474
utterance_id m0000a01
prompt Chèque
text_transcription Chèque
speaking_mode read
sample_begin 0.131750
sample_end 0.771625
sample_count 7693
sample_n_bytes 1
channel_count 1
sample_coding alaw
sample_rate 8000
sample_byte_format 1
sample_sig_bits 8
sample_checksum 15641

Field one and two represent the database and database_id, which are always "Swiss_French_PolyVar" and version "1.0". Field three shows the recording site, which is always "Idiap" and field four is the recording board, which may be "Analogic" or "ISDN". Recording date and time are also included in the NIST header. The "sheet_id" field contains the number of the sheet, which may the same for different sessions and "utterance_id" represents the name of the item. The "prompt" field and "text_transcription" field represent prompted text and the orthographic transcription of the real utterance.
The "speaking_mode" field explains whether the utterance was "read" or "spontaneous". Other fields are in accordance with NIST specifications.
When possible, the recorded signal was cut 200 ms before and 200 ms after the usable speech segment.

Protocol for Picasso on speaker verification

We use only the command words. The database is split into different subsets:

Set 1 and set 2:
Each set contains data from 19 speakers (12M/17F) that can be used as clients. We might need two sets of clients; development set used to compute several global parameters, test set used to validate the experimental algorithm.
The first 5^{th} sessions are used for the client model training. A maximum of 22 accesses is chosen uniformly on the rest of the sessions. The chronology is respected. Each client is an impostor for the other clients: 2 accesses for each impostor. That's mean a maximum of 18734 (17*19*22+17*18*2*19) test access, practically 18106 for the set 1 and 18081 for the set 2.

Pseudo-Impostors set:
This set contains only impostors accesses, usually used for the threshold placement. It is composed of 17 male and 16 female for a total of 9747 accesses.

World model set:
This set contains the data from the speakers used to train the world model. It is composed of 28 male and 28 female for a total of 280 occurrences per word.

Set 1	Set 2	Pseudo		World model		
F45	F44	F00	F01	F02	F03	F04
F47	F46	F07	F10	F05	F06	F08
F49	F48	F11	F12	F09	F17	F18
F51	F50	F13	F14	F19	F21	F22
F53	F52	F15	F16	F26	F27	F30
F55	F54	F20	F23	F31	F32	F33
F57	F56	F24	F25	F34	F35	F36
M01	M00	F28	F29	F37	F38	F39
M03	M02	M38	M42	F40	F41	F42
M05	M04	M43	M45	F43	M24	M25
M07	M06	M46	M49	M26	M27	M28
M09	M08	M52	M57	M29	M30	M31
M11	M10	M60	M63	M32	M33	M34
M13	M12	M64	M66	M35	M36	M37
M15	M14	M68	M77	M65	M67	M69
M17	M16	M78	M80	M70	M71	M72
M19	M18	M81		M73	M74	M75
M21	M20			M76	M79	M82
M23	M22			M83	M84	

Table 6: repartition of the clients on the different sets

The following table shows the statistics of the number of sessions per speaker and the corresponding characteristics.

Male speakers			Female speakers		
Client	**Analog**	**ISDN**	**Client**	**Analog**	**ISDN**
m00	225 sessions		f57	212 sessions	17 sessions
m01	204 sessions	11 sessions	f56	182 sessions	
m02	151 sessions	1 sessions	f55	77 sessions	
m03	161 sessions		f54	162 sessions	
m04	152 sessions		f53	158 sessions	
m05	183 sessions	5 sessions	f52	164 sessions	
m06	146 sessions		f51	73 sessions	
m07	139 sessions		f50	52 sessions	
m08	67 sessions		f49	47 sessions	
m09	65 sessions	2 sessions	f48	40 sessions	2 sessions
m10	63 sessions		f47	41 sessions	
m11	63 sessions		f46	32 sessions	
m12	58 sessions		f45	31 sessions	
m13	56 sessions		f44	30 sessions	
m14	45 sessions		f43	18 sessions	
m15	1 session	45 sessions	f42	14 sessions	
m16	37 sessions	3 sessions	f41	8 sessions	
m17	22 sessions	14 sessions	f40	8 sessions	
m18	26 sessions	8 sessions	f39	7 sessions	
m19	31 sessions		f38	7 sessions	
m20	30 sessions		f37	4 sessions	
m21	28 sessions		f36	3 sessions	
m22	28 sessions		f35	2 sessions	
m23	25 sessions	1 sessions	f34	2 sessions	
m24	13 sessions		f33	2 sessions	
m25	10 sessions		f32	2 sessions	
m26	9 sessions		f31	2 sessions	
m27	9 sessions		f30	2 sessions	
m28	9 sessions		f29	1 session	
m29	8 sessions		f28	1 session	
m30	4 sessions	3 sessions	f27	1 session	
m31	6 sessions		f26	1 session	
m32	5 sessions		f25	1 session	
m33	5 sessions		f24	1 session	
m34	3 sessions		f23	1 session	
m35	3 sessions		f22	1 session	
m36	3 sessions		f21	1 session	

m37	3 sessions
m38	2 sessions
m39	2 sessions
m40	2 sessions
m41	2 sessions
m42	2 sessions
m43	1 session
m44	1 session
m45	1 session
m46	1 session
m47	1 session
m48	1 session
m49	1 session
m50	1 session
m51	1 session
m52	1 session
m53	1 session
m54	1 session
m55	1 session
m56	1 session
m57	1 session
m58	1 session
m59	1 session
m60	1 session
m61	1 session
m62	1 session
m63	1 session
m64	1 session
m65	1 session
m66	1 session
m67	1 session
m69	1 session
m70	1 session
m71	1 session
m72	1 session
m73	1 session
m74	1 session
m75	1 session
m76	1 session
m77	1 session
m78	1 session
m79	1 session
m81	1 session
m82	1 session
m83	1 session
m84	1 session

f20	1 session
f19	1 session
f18	1 session
f17	1 session
f16	1 session
f15	1 session
f14	1 session
f13	1 session
f12	1 session
f11	1 session
f10	1 session
f09	1 session
f08	1 session
f07	1 session
f06	1 session
f05	1 session
f04	1 session
f03	1 session
f02	1 session
f01	1 session
f00	1 session

Table 7: statistic of the number of session per client