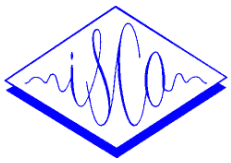
ISCA Archive
http://www.isca-speech.org/archive



3rd European Conference on
Speech Communication and Technology
EUROSPEECH'93
Berlin, Germany, September 19-23, 1993

# THE PHILIPS RESEARCH SYSTEM FOR LARGE-VOCABULARY CONTINUOUS-SPEECH RECOGNITION

*V. Steinbiss, H. Ney, R. Haeb-Umbach, B.-H. Tran, U. Essen, R. Kneser, M. Oerder, H.-G. Meier, X. Aubert, C. Dugast, D. Geller*
Philips GmbH Forschungslaboratorien • 52066 Aachen • Germany • email: steinbiss@pfa.philips.de

*W. Höllerbauer, H. Bartosik*
Philips Dictation Systems • 1102 Wien • Austria

## ABSTRACT

This paper gives a status report of the Philips research system for phoneme-based, large-vocabulary, continuous-speech recognition. Like for many other systems, the recognition architecture is based on an integrated statistical approach. We describe the characteristic features of the system as opposed to other systems: 1. The Viterbi criterion is consistently applied both in training and testing. 2. Continuous mixture densities are used without tying or smoothing. 3. Time-synchronous beam search in connection with a phoneme look-ahead is applied to a tree-organized lexicon.

The system has been successfully applied to the American English DARPA RM task. Here, we report experimental results for a German 13 000-word Philips internal dictation task. In addition to the scientific prototype, a PC version has been set up which is described here for the first time.

**Keywords:** Continuous speech recognition, large vocabulary recognition.

## 1. Introduction

For large-vocabulary, continuous-speech recognition, there are a number of operational prototype systems in research. Like these systems and the IBM system for 20 000-word recognition of isolated-word input, the prototype system described in this paper is based on techniques of statistical pattern recognition and stochastic modelling, where training data are heavily exploited and local decisions are avoided as far as possible. See [12, 16] for references.

The characteristic features of the approach to be presented are:

- A large-sized acoustic vector capturing first and second-order derivatives is used. There is no splitting into separate streams as in most other systems that use tied-mixtures.
- The Viterbi criterion is used both in training and recognition. Continuous mixture densities are used in a way that amounts to what can be called 'statistical template matching'.
- Linear discriminant analysis improves the acoustic analysis.
- For bigram language modelling, a non-linear interpolation has been developed that gives consistently lower perplexities than linear interpolation.
- The concept of time-synchronous beam search has been extended towards a tree organization of the pronunciation lexicon so that the search effort is significantly reduced. A phoneme look-ahead technique results in an additional improvement. A PC based implementation (cf. sect. 8) underlines the efficiency of this search strategy.

The organization of the paper is as follows. We first summarize the statistical approach to speech recognition and then describe the four main entities of our system: acoustic analysis, acoustic-phonetic modelling, language modelling and search. A section with experiments on our internal dictation task follows. The final section describes a PC based implementation of our system.

## 2. System Architecture

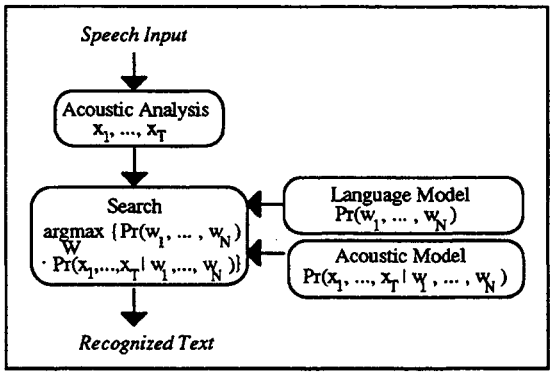



*Fig. 1:* System Architecture

Fig. 1 presents a block diagram of the system architecture. In the preprocessing step of *acoustic analysis*, the speech signal is transformed into a sequence of acoustic vectors $x_1,\ldots,x_T$ (over time t=1,...,T). As the speech signal, and thus this sequence of observations, is not exactly reproducible, a statistical approach is used to model its generation. Statistical decision theory tells that in order to minimize the probability of recognition errors, one should decide for the word sequence $W=w_1,\ldots,w_N$ (of unknown length N) that maximizes [8]

$$Pr(w_1,\ldots,w_N)\ Pr(x_1,\ldots,x_T|w_1,\ldots,w_N) \quad . \qquad (1)$$

The first term, the a-priori probability of word sequences $Pr(w_1,\ldots,w_N)$, is independent of the acoustic observations and is completely specified by the *language model*. It reflects the system's knowledge of how to concatenate words of the vocabulary to form whole sentences and thus captures syntactic and semantic restrictions.

The *acoustic-phonetic modelling* is reflected by the second term. $Pr(x_1,\ldots,x_T|w_1,\ldots,w_N)$ is the conditional probability of observing the acoustic vectors $x_1,\ldots,x_T$ when the words $w_1,\ldots,w_N$ were uttered. These probabilities are estimated during the training phase of the recognition system. A large-vocabulary system typically is based on subword units like phonemes, which are concatenated according to the *pronunciation dictionary* to form word models.

The decision on the spoken words must be taken by an optimization procedure which combines information of the language model and of the acoustic model, the latter being based on the phoneme models and the pronunciation dictionary. The

optimization procedure is usually referred to as *search* in a state space defined by the knowledge sources.

## 3. Acoustic Analysis

### 3.1 Spectral Analysis

The acoustic signal is low-pass-filtered and digitized with a sampling frequency of 16 kHz. The following steps are performed for every frame, i.e. every 10 ms:

- Application of a Hamming window to a 25-ms segment.
- 512-point FFT after padding with zero-valued samples.
- Cepstral smoothing of the logarithmic FFT intensities using a *sin(x)/x* kernel function.
- In the range from 200 Hz to 6400 Hz, sampling at 30 frequency points that roughly correspond to a Mel-frequency scale.
- Normalization of the 30 spectral intensities with respect to their mean value. Together with this "energy" value, they form the 31-dimensional acoustic vector y(t).

To account for varying recording conditions in the dictation task, each acoustic vector is normalized with respect to the long-term spectrum as obtained by averaging over a part of the sentence.

In order to capture the temporal structure of the speech signal, each acoustic vector y(t) is then augmented by slope and curvature information over the time axis. Thus, the original sequence of y(t) of acoustic vectors is replaced by

$$x(t) := \begin{bmatrix} y(t) \\ y'(t) \\ y''(t) \end{bmatrix} = \begin{bmatrix} y(t) \\ y(t) - y(t-\Delta t) \\ y(t+\Delta t) - 2y(t) + y(t-\Delta t) \end{bmatrix} , \quad (2)$$

where the first- and second-order differences were chosen to cover the time intervals [t-Δt, t] and [t-Δt,t+Δt], respectively. The time delay Δt is typically 30 ms. The new sequence of acoustic vectors $x_1,...,x_T$ in a higher-dimensional vector space serves as input to the subsequent processing steps. For the first and second differences of the 30 spectral intensities, pairs of adjacent spectral intensities are averaged so that the final vector consists of 63 components: 30 spectral intensities, 15 first- and 15 second-order differences, and 3 components representing energy and its differences.

### 3.2 Linear Discriminant Analysis

Linear discriminant analysis (LDA) is a well-known technique in statistical pattern classification for improving the discrimination between classes in a high-dimensional vector space ([3] pp. 114 ff.). The basic idea is to find a linear transformation such that a suitable criterion of class separability is maximized. The transformation is obtained as the eigenvector decomposition of the product of two scatter or covariance matrices, the total-scatter matrix and the inverse of the average within-class scatter matrix. Recently, this technique has been successfully applied to speech recognition, for both small [4, 7] and large-vocabulary tasks [6].

When applying LDA to speech recognition, the choice of the proper classes to be discriminated is not obvious - are they whole phonemes, phoneme states or the mixture components of a state? Our experiments indicated that the states are a good choice. The computation of the LDA transform is further complicated by the time alignment problem. Therefore, we use a three-step training. With our standard iterative training we obtain a segmentation of the training data, which provides the class labels for the subsequent estimation of the LDA transform. The third step is a new iterative training using LDA-transformed acoustic vectors.

Note that since a *single* class-*in*dependent transformation matrix is employed, the matrix multiplication is done in the acoustic front end once per frame rather than for each log-likelihood calculation.

## 4. Acoustic-Phonetic Modelling

The acoustic conditional probabilities $Pr(x_1,...,x_T|w_1,...,w_N)$ are obtained by concatenating the corresponding word models, which again are obtained by concatenating phoneme models according to the pronunciation lexicon. We use inventories of 40-50 phoneme symbols including symbols for silence and maybe glottal stop. As in many other systems, these subword units are modelled by stochastic finite-state automata, the so-called Hidden Markov Models (HMMs) [2, 8, 11].

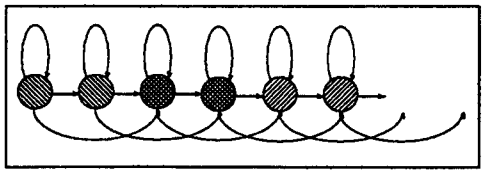


*Fig. 2:* Topology of phoneme HMM

For each state s of the HMM, there is an emission probability density $q(x_t|s)$ of generating the vector $x_t$. The phoneme unit shown in Fig. 2 has a tripartite structure in order to take account of left and right acoustic dependences. Each of the three parts consists of two states with identical emission distributions. The transition probabilities, which allow loop, jump and skip, are tied over all states. Unlike most other HMM structures, this structure has a simple durational model whose most likely duration of 60 ms is close to the average phoneme duration.

No pronunciation variants are used in the pronunciation lexicon, such that the emission distributions have to model deviations from the standard pronunciation as well as coarticulatory effects. The best results were obtained for continuous mixture densities

$$q(x_t|s) = \sum_k c_k(s) b_k(x_t|s) \quad \text{with } 0 \le c_k \le 1 \text{ and } \sum_k c_k(s) = 1 \quad (3)$$

where the so-called component densities $b_k(.|.)$ are unimodal densities such as Gaussians or (as in our system) Laplacians:

$$b_k(x_t|s) = \prod_n \left( \frac{1}{2v_n} \right) \cdot \exp\left( -\sum_n \frac{|x_t(n) - r_{k,s}(n)|}{v_n} \right) . \quad (4)$$

n is the index of the vector components. Each density is completely specified by its location vector $r_{k,s}$. The vector of absolute deviations, $(v_1,...,v_N)^t$, is assumed to be independent of both the component densities and the states and thus serves as an overall scaling for the acoustic vectors.

In contrast to other systems, the Viterbi criterion is used both in training and recognition. This applies even to the level of mixture components, such that the sum over the component densities in eq. (3) is replaced by their maximum [12].

While we typically develop our system on a speaker-dependent German task (cf. sect. 7), we also successfully benchmarked our system on both the speaker-dependent and the speaker-independent part of the well-known American English DARPA (Defense Advanced Projects Agency) RM (resource management) task [1] [12]. The major modifications of our system were the usage of context-dependent phoneme models and a large number of densities. In contrast to other systems, the system does not use across-word models, and there is typically no smoothing of emission probabilities.

## 5. Language Modelling

The language model provides, for each word sequence, an estimate of the probabilities $Pr(w_1,...,w_n)$ or, equivalently, of the conditional probabilities $Pr(w_n|w_1,...,w_{n-1})$. *m*-gram models [9] have established themselves as both a good way to reliably estimate the parameters and to keep them limited so they can be stored and retrieved. In view of the size of corpora available, we typically use a word bigram model $P(w_n \mid w_{n-1})$ or a category-based bigram model (bigram class model) with automatically generated classes [10].

While maximum-likelihood estimation would suggest to take relative frequencies of bigram counts, it is common knowledge that these are particularly bad as estimates and that smoothing is important. The smoothing method that we use is different from those used in other systems. The non-linear interpolation scheme that we use essentially amounts to substracting a constant *d* from the counts and distributing the gained probability mass on less detailed distributions [13]. With this method, we achieve better results than with backing-off or linear interpolation.

## 6. The Search Procedure

Time-synchronous beam search has successfully been used in the Philips continuous-speech recognizer for several years [15]. We found that it is efficient also for 10 000 or more words [14]. First, all knowledge sources are available at the same level in the integrated search. Second, all hypotheses refer to the same acoustic vector sequence in time-synchronous search. These two key points allow a drastic reduction of the actual search space by pruning less promising hypotheses.

### 6.1 Tree Lexicon

A straight-forward approach of constructing the search space is to synthetically build up word models from concatenating the appropriate phoneme models as given by the pronunciation lexicon. In this space, different copies of the same phoneme occur due to the lexical constraints. For similar reasons, the language model restrictions make it necessary to introduce several copies of the same word, representing contexts that allow for different continuations. This organization, where each state belongs to exactly one word, will be called *linear lexicon*.

When the lexicon grows larger, e.g. from 1000 to 10 000 words, it is more efficient to arrange the pronunciation lexicon as a tree of phonemes (*tree lexicon*). The compression factor for the tree lexicon as compared to the linear lexicon is even surpassed by the reduction in the number of active states, because most of the active states are located in the word beginnings (near the tree's root).

The tree organization of the lexicon also has an undesired consequence for the organization of the search space. In contrast to a linear lexicon, the word identities are unknown at the word beginnings. Particularly for a bigram language model, this means that separate tree copies have to be held, depending on the predecessor word. While the potential search space is blown up by a factor of the vocabulary size, e.g. 10 000, the actual search space grows much more moderately, typically by only a factor of 2. The tree organization is thus very beneficial for large-vocabulary tasks. A detailed discussion with experiments is given in [14].

### 6.2 Phoneme Look-Ahead

The phoneme look-ahead additionally reduces the number of active states by estimating whether a started phoneme will or will not survive the next few time frames (in our system typically 60 ms). In a first step, the likelihood of each phoneme ahead of the current time frame is estimated by carrying out a time-alignment. Then, each time a state hypothesis crosses a phoneme boundary, these figures are utilized for probability estimates for the best path extensions both of this and of any other state, which in turn are used to perform an additional pruning [5].

For the phoneme look-ahead, the original phoneme models are used without any simplification. Note that, in particular for the case of monophones, the number of generic states is much smaller than the number of state hypotheses. The likelihood scores are stored for later use in the detailed match. Like the conventional search, the look-ahead is sped up by beam pruning; in addition, there is no need for book-keeping as in the detailed match. To further reduce computation, the look-ahead is carried out only every other time frame. For the omitted time frame, the look-ahead scores of the previous time frame are used.

## 7. Experimental Tests

We give a very brief look on experiments conducted in connection with our speaker-dependent dictation task. Experiments on other, public available databases that allow a comparison with other systems, are described elsewhere [12, 16].

The data in these experiments are real-life field data from professional text producers. Speakers M-60 and M-61 are lawyers, M-72 and M-73 are radiologists. All speakers are male and work in Vienna, Austria. The speakers were asked to dictate as usual; this includes verbalized punctuation. The dictations were recorded with hand-held microphones on desktop dictation equipment. We processed exactly the same recordings that were also given to the secretaries for transcription. Although the speakers are very experienced with dictation, we found that recognition was harder on this material than on read texts.

| Speaker | Vocabu-lary | Test-set perplexity | Active states / centisecond | Word-error rate in % Del. | Ins. | Total |
|---|---|---|---|---|---|---|
| M-60 | 12 073 | 113 | 8 700 | 3.1 | 1.0 | 10.2 |
| M-61 | 15 188 | 176 | 9 300 | 1.9 | 1.5 | 12.1 |
| M-72 | 13 095 | 267 | 11 600 | 2.4 | 1.9 | 11.2 |
| M-73 | 13 095 | 42 | 14 000 | 0.6 | 1.7 | 5.7 |

*Table 1:* Baseline system on 4 field test speakers. Bigram LM, look-ahead, no LDA, 9 h training, 16 000 mixture components.

| Training material / No. of Densities | 0.7 h | 1.2 h | 2.0 h | 3.2 h | 9.5 h |
|---|---|---|---|---|---|
| 4000 | 16.1% | 14.4% | 13.1% | 12.3% | 11.4% |
| 8000 | - | 13.4% | 12.9% | 11.7% | 10.8% |
| 16000 | - | - | - | 11.6% | 10.1% |

*Table 2:* Error-rate as a function of training set size and number of densities. Speaker M-60, vocabulary size 12 073 words, test-set perplexity 113.

| Speaker | no LDA | LDA |
|---|---|---|
| M-60 | 12.3 | 10.4 |
| M-61 | 15.0 | 12.3 |

*Table 3:* Effect of LDA on the word-error rate (in %). About 3 h training, 4 000 densities.

Table 1 shows the performance of our system with high acoustic resolution (16 000 mixture components) and about 9 hours of training material, but without LDA. The test material comprised 2000 - 3000 spoken words. The number of active states per centisecond before pruning refers to the search effort. Table 2 shows how the error-

rate depends on the training-set size and the acoustic resolution. Monophones were used her; we expect improvements with context-dependent phonemes. Table 3 shows the improvement by LDA.

## 8. A PC Based Continuous-Speech Recognition System for Dictation

Real-world dictation, which is typically connected with large and open vocabulary, is a difficult task that pushes today's technology to its limits. Despite the considerable progress made in recent years, even for co-operative speakers and restricted domains like free-text medical reporting, error-free speech recognition so far cannot be achieved.

Up to now, large-vocabulary systems for dictation have required isolated word input. While this reduces both the word-error rate and computational costs as compared to continuous-speech input, it burdens the user with an unnatural speaking style. In addition, dictating with pauses between words takes more time.

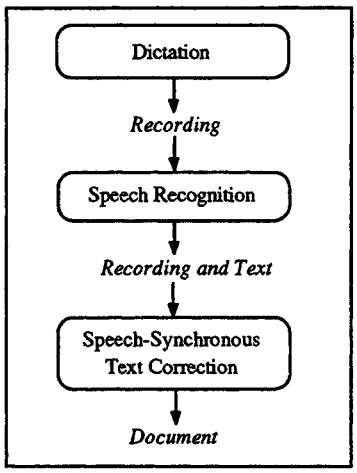



*Fig. 3:* System architecture of the PC based continuous-speech recognition system for dictation.

For people who professionally generate large amounts of texts, e.g. physicians and lawyers, text generation is characterized by a two-step process: The phase of dictation, where speech is recorded either digitally or on tape, and a subsequent separate transcription phase where secretaries transcribe the dictations. (We ignore the proof-reading in this discussion.)

The system developed at Philips Dictation Systems, Vienna, and described here adopts this non-interactive approach and thus allows the person to dictate with a natural speaking style. After the speech is processed by the speech recognizer, the secretary has only to correct the recognition errors, which is both faster and a more interesting job to do.

This three-step approach naturally results in the system architecture as summarized in Fig. 3:

- The dictation is recorded using a microphone; the usual record/replay/fast-forward/rewind functionality is available. So, no change of work methodology is required. Punctuation should be verbalised. The recorded speech is stored on a fileserver in a PC network.
- Speech recognition runs remotely on a PC which is connected to the network. An acoustic front-end performs the acoustic analysis. Recognition is sped up by a dedicated co-processor board containing application-specific ICs. Depending on the speaker and the specific boundary conditions, recognition with a 10 - 20 000-word vocabulary runs in 1 - 3 times real-time.
- In contrast to typing the whole text, the secretary only corrects the errors that occurred in the recognition process. With a special speech-synchronous editor that uses the link between the recording and the text as given by the hypothesized word boundaries, it is possible to listen to parts of the recording while moving through the text.

Three measures have been taken to achieve the lowest possible error-rates without hindering the person who dictates:

- The system has been set-up in speaker-dependent mode such that each of the speakers gets optimal performance.
- Training is done with the dictations that are being produced anyway, together with their proper transcriptions. After several hours of dictations, the system reaches optimal performance.
- A high-quality acoustic analysis together with a large number of mixture components guarantee a high acoustic resolution.

The first release of this system is a German version. Field trials are being carried out in several hospitals in Austria and Germany.

## REFERENCES


Abbreviation: *ICASSP* stands for *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing.*

[1] X. Aubert, R. Haeb-Umbach, H. Ney: "Continuous Mixture Densities and Linear Discriminant Analysis for Improved Context-Dependent Acoustic Models", ICASSP, Minneapolis, MN, pp. II 648-651, April 1993.

[2] J. K. Baker: "Stochastic Modeling for Automatic Speech Understanding", in D. R. Reddy (ed.): 'Speech Recognition', Academic Press, New York, pp. 512-542, 1975.

[3] R.O. Duda, P.E. Hart: 'Pattern Classification and Scene Analysis', Wiley, New York, 1973.

[4] M. J. Hunt, C. Lefebvre: "A Comparison of Several Acoustic Representations for Speech Recognition with Degraded and Undegraded Speech", ICASSP, pp. 262-265, Glasgow, May 1989.

[5] R. Haeb-Umbach, H. Ney: "A Look-Ahead Search Technique for Large-Vocabulary Continuous-Speech Recognition", Proc. Europ. Conf. on Speech Communication and Technology, Genova, pp. 495-498, Sep. 1991.

[6] R. Haeb-Umbach, H. Ney: "Linear Discriminant Analysis for Improved Large Vocabulary Continuous Speech Recognition", ICASSP, San Francisco, CA, pp. I 13-16, March 1992.

[7] R. Haeb-Umbach, D. Geller, H. Ney: "Improvements in Connected Digit Recognition Using Linear Discriminant Analysis and Mixture Densities", ICASSP, Minneapolis, MN, pp. II 239-242, April 1993.

[8] F. Jelinek: "Continuous Speech Recognition by Statistical Methods", Proc. of the IEEE, Vol. 64, No. 10, pp. 532-556, April 1976.

[9] F. Jelinek, R. L. Mercer, S. Roukos: "Principles of Lexical Language Modeling for Speech Recognition", *in* S. Furui, M. M. Sondhi (eds.): 'Advances in Speech Signal Processing', Marcel Dekker, New York, pp. 651-699, 1992.

[10] R. Kneser, H. Ney: "Improved Clustering Techniques for Class-Based Statistical Language Modelling", *elsewhere in these* Proc. Europ. Conf. on Speech Communication and Technology, Berlin, Sep. 1993.

[11] S. E. Levinson, L. R. Rabiner, M. M. Sondhi: "An Introduction to the Application of the Theory of Probabilistic Functions of a Markov Process to Automatic Speech Recognition", The Bell System Technical Journal, Vol. 62, No. 4, pp. 1035- 1074, April 1983.

[12] H. Ney: "Modeling and Search in Continuous-Speech Recognition", *elsewhere in these* Proc. Europ. Conf. on Speech Communication and Technology, Berlin, Sep. 1993.

[13] H. Ney, U. Essen: "On Smoothing Techniques for Bigram-Based Natural Language Modelling", ICASSP, Toronto, pp. 825-828, May 1991.

[14] H. Ney, R. Haeb-Umbach, B.-H. Tran, M. Oerder: "Improvements in Beam Search for 10000-Word Continuous Speech Recognition", ICASSP, San Francisco, CA, pp. I-9 - I-12, March 1992.

[15] H. Ney, D. Mergel, A. Noll, A. Paeseler: "Data Driven Organization of the Dynamic Programming Beam Search for Continuous Speech Recognition", IEEE Trans. on Signal Processing, Vol. SP-40, No. 2, pp. 272-281, Feb. 1992.

[16] H. Ney, V. Steinbiss, R. Haeb-Umbach, B.-H. Tran, U. Essen, "An Overview of the Philips Research System for Large-Vocabulary Continuous-Speech Recognition", *to appear in* Int. Journal of Pattern Recognition and Artificial Intelligence, 1994.