ON THE DEVELOPMENT OF A DICTATION MACHINE FOR SPANISH: DIVO*

Javier Macías-Guarasa, Manuel A. Leandro, José Colás, Álvaro Villegas, Santiago Aguilera and José M. Pardo
Grupo de Tecnología del Habla. Departamento de Ingeniería Electrónica. E.T.S.I. de Telecomunicación
Ciudad Universitaria s/n | 28040 - MADRID | SPAIN
macias@die.upm.es

ABSTRACT

The first prototype of a low cost dictation machine for Spanish is described (DIVO). The main characteristics of our recognition approach are: bottom-up, hypothesis-verification strategy; large vocabulary, speaker dependent, isolated word recognition. Its modular structure is the cue for quick development and testing of different implementation alternatives. Two of them are presented: one is based in Static phoneme Modeling (SM) and the other uses Discrete Hidden Markov Modeling (DHMM). The system runs on a standard PC compatible (286 or higher) equipped with a DSP board and is fully voice controlled. This first version of the system can address multiple vocabulary sets of up to 2000 words each, with immediate response and reasonable performance. Modules for increasing vocabulary and performance are being developed.

I. INTRODUCTION

The basic configuration of the isolated word recognition system is shown in Figure 1 below.

The core of the recognition process is based in a hypothesis-verification paradigm. A preselection module with low computational cost selects a list of candidate words. It must ensure that the word pronounced will be within this list with very high probability. Coming into detail, our approach uses a bottom-up strategy, i.e., there are intermediate data structures prior to obtaining the final preselection word list, like the systems described in [7] [8]. This approach opposes to the top-down strategy in which the hypotheses are generated in a direct and integrated way [9] [10].

The main advantage of the proposed structure is the lower computational cost required and the extended flexibility in testing different technological alternatives for each module. The major disadvantage is the lower recognition rate that can be achieved, as the process is not tightly integrated and therefore not guided nor restricted.

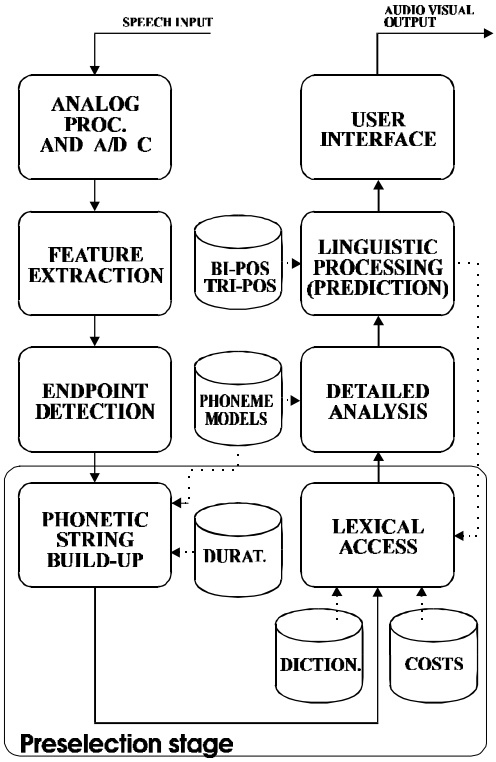

Figure 1.- System Architecture.

Different alternatives can be implemented using the same scheme. Two approaches are being discussed here (related to the phonetic string build up and detailed analysis stages): On one hand *a static pattern matching algorithm* (SM) that exploits the characteristics of the Spanish language. On the other, a traditional *discrete Hidden Markov Model based system* (DHMM). The rest of the modules are common for both alternatives.

The organization of the paper is as follows:

Section II describes the software and hardware architecture of DIVO, a dictation machine for Spanish. Under the same architecture, two different underlying recognition technologies are compared. Performance measurements for the already real-time implemented stages are presented in section III for both approaches. Section IV presents our conclusions and section V includes the planning for further developments.

* This work has been partially supported by INSERSO (Ministerio de Asuntos Sociales) and CYCIT grant #TIC94-0119.

II.- SYSTEM OVERVIEW

II.1.- Software Architecture

II.1.a. Preprocessing and feature extraction (FE): A close talking dynamic microphone is used. The analog process low-pass filters the speech signal and A/D conversion is performed with 14 bits-resolution and 16 KHz of sampling frequency. 10 mel frequency ceptrum coefficients plus log-energy are computed, with a 16 ms. hamming window in 6.25 milliseconds steps for the HMM-based system (10 milliseconds in the SM approach) after high frequency emphasis. An automatic endpoint detector based in energy thresholds, event durations and a set of heuristics is used. In the DHMM system, vector quantization using 128 centroids and the Mahalanobis distance with diagonal covariance matrix is applied.

II.1.b. Phonetic String Build-Up (PSBU):

***SM approach*:** It exploits the acoustic-phonetic characteristics of the Spanish language, that can be reasonably modeled considering only stable phonemes. The recognition process makes a distance measure between each frame and each prototype, and keeps the closest prototype symbol, generating a symbol sequence (one symbol per frame). The phonetic sequence information is combined with a spectral stability function and acoustic heuristic rules to obtain the final phonetic string. 15 prototypes are used, extracted from 150 words spoken by the user in the training stage. Further details of this scheme can be found in [2].

***DHMM approach*:** It uses a low-cost, frame-synchronous, one pass algorithm [3], based on discrete Hidden Markov Models. It is given the indexes of the quantified vectors from the previous stage. When the end of the word is reached, a backtracking procedure recovers the most-likely sequence of phoneme-like units according to the incoming speech signal (although additional procedures may allow the parallel execution of phonetic decoding and one-pass calculations). In the current implementation, 23 models are generated plus two additional ones for initial and final silence (to improve robustness against errors in the endpoint detection). Traditional Bakis DHMM's are used with three states per model and single and double transitions allowed. The size of the alphabet and the units considered have been chosen to keep computational load low without losing modeling accuracy and training generality. The training database is composed of 500 words and the training procedure is fully automatic.

II.1.c. Lexical Access (LA): It computes similarities between the phonetic-string (with errors due to the non-guided previous phonetic decoding) and each item of the word dictionary. The similarity is computed by a dynamic programming procedure using substitution, insertion and deletion costs [5], previously trained from a 500 word database for the DHMM system and 150 for the SM alternative. The training procedure in this case is done through an iterative algorithm, estimating the occurrences of each possible error and smoothing the final costs. Optimal convergence is reached in three to four iterations. The insertion and substitution costs are context-dependent and the deletion cost is context-independent.

To reduce computational requirements, the dictionary has been implemented using a tree-structure, in which common initial phoneme sequences are shared to reduce the size of the search space by a factor of two to three. Words with the highest scores are selected for the preselection list and are forwarded to the detailed analysis stage.

II.1.d. Detailed analysis (DA):

SM approach: A Dynamic Time Warping procedure computes a score aligning the incoming frame sequence with a word model built by concatenating static phoneme patterns. A duration model helps the alignment process.

DHMM approach: A traditional viterbi algorithm computes a score for each word (concatenating phoneme models) to rearrange the candidates in the preselection list.

II.1.e. Linguistic module (LM): Currently under development, this module works in order to predict the words which are more likely to follow in the sentence (to reduce perplexity), based in a bi or tri-pos-like language model..

II.2.- Hardware Architecture:

The core of the hardware subsystem is the ATT DSP32C, a 32-bit floating point Digital Signal Processor (DSP), running at 40 MHz. It is assembled in a low-cost PC-board developed in our laboratory (called VISHA), to be plugged in a 8 bit bus expansion slot. The board manages up to 4 Mbytes of external SRAM (currently using 512 Kbytes). The analog interface is built around the TLC32044 from Texas Instruments, which include 14 bits linear A/D and D/A converters and the antialiasing and reconstruction switching capacitors filters. Finally, the board includes two amplifiers: the input amplifier adjusts the microphone signal level, and the output one can drive directly an 8 ohm loudspeaker. The VISHA is being used to develop several projects related to speech technology in our department [1] [11] [12].

II.3.- Hardware-Software Integration

II.3.a. Tasks arrangement: All the currently real-time implemented stages of the algorithm (feature extraction, phonetic string build up and lexical access) are performed by the VISHA's DSP. Dictionaries and speaker's models are previously loaded into VISHA's memory by the PC. The only task related to recognition the PC has to perform is the reception of incoming messages from the board. DSP's work is completely independent from PC: it continuously runs the recognition algorithm on the incoming voice samples and, when a word is recognized, it signalizes the event to the PC and keeps on *hearing*. The computer can use these messages as a keyboard signal. In

fact, with an appropriate driver that emulates keyboard or mouse using as input the recognition messages, any existing application can be easily controlled by voice. Such a driver has been developed for both MS-DOS® and WINDOWS™.

II.3.b. Synchronization and time requirements: The synchronization of the three implemented stages of the algorithm (feature extraction, phonetic string build up and lexical access) is different for each method (SM and DHMM): both of them perform feature extraction and phonetic string build up in a frame-synchronous fashion (they are executed as frames come from the front-end), but while the SM model performs *pieces* of lexical access with the already built portion of the phonetic string, the DHMM version has to wait until the phonetic string has been completely built (i.e., word end is reached) to perform the lexical access. This difference, due to the lower computational cost of the phonetic string build up for the SM model and the need of additional partial traceback mechanisms for the DHMM, produces a slightly longer response time for the latter. The delay, however, is in most cases shorter than the pause the speaker has to make between words to let the word end be detected. Moreover, implementation improvements already devised to synchronize both processes will overcome this potential drawback.

In Figure 2, we show the timings in the real-time implementation for SM and DHMM alternatives (units are given in number of samples). While in the SM approach, 126 samples are free to carry out additional processing (lexical access, detailed analysis, etc.), in the DHMM one, the idle time is wasted, as string generation is performed when the word reaches its end.

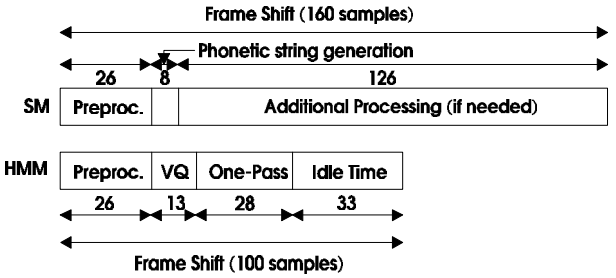

Figure 2.- Comparison of timings for SM and DHMM

II.3.c. Additional real-time features: Real-time implementation of the recognition algorithm has provided the system with some new features: concurrent processing of multiple dictionaries, secondary dictionaries (multiple lexical access using DSP idle time between words), recognition cancellation based on reliability of incoming sound signal, dynamic adaptation to noise level, etc. All of them improve system behavior in real use situations.

II.3.d. Memory requirements: The amount of memory required on the VISHA for the recognition process depends on the vocabulary size needed for the task. Dictionaries are loaded into the board's memory in phonetic form. If the application running on the computer needs graphemes (e.g. a dictation machine), they are kept on PC's memory (expanded memory if available). In the current implementation, more than 10000 words can be loaded into the 512Kb memory of the VISHA board. Anyway, a simple memory expansion (up to 4Mb) can raise this limit, so the effective size of the vocabulary would only be limited by the recognition rate required for the task and achieved by the recognition algorithms.

II.3.e. User interface: The first application we have developed based on the underlying recognizer is DIVO: a window-based dictation machine which provides text editing through voice. The environment is primarily oriented to handicapped people, whose only communication way with a computer is speech. Because of that, DIVO is fully voice controlled, including file management, menu selection, automatic new speaker enrollment and dictionaries customization. Even recognition errors can be corrected by the speaker through voice commands.

II.3.f. Application programming interface: In order to make future developments easier, an application interface has been developed for the recognizer. It consists of a library of functions in C language that performs both control (program and dictionaries loading, multiple dictionaries features, graphemes storing, etc.) and communication (recognized word messages) tasks. Functions for new speaker enrollment and dictionaries customization are also available. The structure of functions is versatile and independent of the recognition technology used (SM or DHMM) or version, allowing separate improvements of both applications and recognizer subsystems.

II.3.g. Implementation status: As mentioned above, the three first stages of the recognition process are already real-time implemented for both SM and DHMM versions. The detailed analysis stages for both SM and DHMM are fully developed and being implemented on the VISHA. Language modeling is currently being developed in simulation, and will be adapted for real-time operation as soon as possible.

III. EXPERIMENTAL RESULTS

The dictation machine is being tested by measuring recognition performance and applying usability tests to feedback the user interface design. We present results for two speakers as a function of the position in which the word was correctly recognized. Rates for the 1st, 5th and 40th candidates are shown in Table 1, to give an idea of the evolution and the inclusion rate achieved for each of the technologies implemented in the preselection stage (phonetic string build up and lexical access). The test database was composed of 500 words different from the ones used in the training script. 2000 words are used as the test vocabulary. No language model is used (i.e., perplexity equals vocabulary size).

	Candidate	SM	DHMM
Speaker 1	1st position	69,7 %	78,2 %
	5th position	89,8 %	92,4 %
	40th position	97,2 %	98,4 %
Speaker 2	1st position	58,5 %	66,5 %
	5th position	80,6 %	86,6 %
	40 th position	96,0 %	97,6 %
Average	1st position	64,1 %	72,4 %
	5th position	85,2 %	89,5 %
	40 th position	96,6 %	98,0 %

Table 1.- Experimental results

The training database used has a low signal to noise ratio due to problems not detected till the verification stage, and resulted to be under 15 dB. So, these results are encouraging if we consider the high inclusion rate achieved and will be validated with further experiments (simulation results are shown in [6] for another two speakers with cleaner test databases).

IV.- CONCLUSIONS

The systems presented have proved to be efficient approaches to the development of a reliable dictation machine for Spanish. The architecture of the underlying recognition system allows quick testing of different technological alternatives, two of which have been presented. The high inclusion rate of the preselection stage (phonetic string build up and lexical access) allows us to predict a very high performance when using the detailed analysis and language modeling stages.

The low computational complexity of the approaches considered, specially in the SM case, will allow further refinements of the algorithms without compromising response time.

Regarding the alternatives for the preselection stage, we have shown that while the SM technology is cheaper in training and computational complexity, it has a worse recognition performance than the DHMM alternative for lower candidate positions. Nevertheless, for the detailed analysis stage, the point to consider is the inclusion rate in the preselection list, where the difference between both systems shortens considerably. The compromise and balance between this two factors will limit the tasks each one is suitable for.

More details on the comparison of both systems in the simulation stage can be found in [6] in this same conference.

V.- FUTURE TASKS

Our main objective is raising the vocabulary size to 10.000 to 20.000 words. Prior to this achievement we have to fully develop the detailed analysis and language modeling algorithms.

In the DHMM approach, refinement in the feature extraction process along with improvements in the implementation to allow parallel execution of the lexical access and detailed analysis are to be made (partial traceback procedures [4]). Speaker adaptation methods to reduce the size of the training script will be also developed.

In the SM approach, dynamic models for non-stable speech segments (diphones) will be studied to increase the performance in the detailed analysis stage.

VI.- BIBLIOGRAPHY

[1] S. Aguilera et al. “Impaired persons facilities based on a multi-modality speech processing system”. Speech and Language Technology for Disabled Persons. ESCA Workshop, pp. 129-132, 1993.

[2] M. Leandro and J.M. Pardo. “Low cost speaker dependent isolated word speech preselection system using static phoneme pattern recognition”. Eurospeech, vol. 1, pp. 117-120, 1993.

[3] H. Ney. “The use of a one-stage dynamic programming algorithm for connected word recognition”. IEEE Transactions on ASSP, vol. 32, n. 2, 1984.

[4] P.F. Brown et al. “Parcial traceback and dynamic programming”. ICASSP, pp. 1629-1632, 1982.

[5] L. Fissore et al. “Lexical access to large vocabularies for speech recognition”. IEEE Transactions on ASSP, vol. 17, n. 8, pp. 1197-1213, 1989.

[6] J. Macías-Guarasa et al. “Comparison of three approaches to phonetic string generation for word recognition”, in this conference. 1994.

[7] H. Ney and R. Billi. “Prototype systems for Large Vocabulary speech recognition: Polyglot and Spicos”. Eurospeech, pp. 193-200, 1991.

[8] P. Buttafava et al. “Architecture and Implementation of the Olivetti PC-based very large vocabulary isolated word speech recognition system”. Eurospeech, pp. 90-93, 1990.

[9] H. Cerf-Dannon et al. “1.0 Tangora - A large vocabulary speech recognition system for five languages”. ICASSP, pp. 183-192, 1992.

[10] J.M. Baker. “Large vocabulary speaker-adaptive continuous speech recognition research overview at Dragon Systems”. IEEE Transactions on ASSP, vol 23, pp. 24-79, 1990.

[11] M.A. Berrojo et al. “A PC graphic tool for speech research based on a DSP board”. ICSLP, pp. 1663-1636, 1992.

[12] J..F. Mateos es al. “A PC card for the rehabilitation of deficient auditive people”. EUSIPCO, pp. 1175-1178, 1990.