Creating a Grammar-Based Speech Recognition Parser for Mexican Spanish Using HTK, Compatible with CMU Sphinx-III System

Carlos D. Hernández-Mena and Abel Herrera-Camacho
National Autonomous University of Mexico (UNAM)/Signal Processing Department, Mexico City
Email: ca_hernandez@uxmcc2.iimas.unam.mx, abelherrerac1@gmail.com

***Abstract*—In this paper, we present the creation of a novel language model for automatic speech recognition based on HTK but compatible with the CMU Sphinx-III speech recognition system. This recognizer works with user defined grammars in the HTK format for speaker dependent recognition in Mexican Spanish. Input files are based on the Sphinx format, so you can use them with no modification in both systems. We provide with the recognizer a "live decode" module based on the Japanese speech recognition system called Julius for live recognition. We will also explain the phonetic alphabet chosen for the system and we will explore some of the new resources that are now available for the Mexican Spanish language. This software is available for free use and it is compatible with the most common operating systems (Windows, GNU-Linux and MAC OS).**

***Index Terms*—speech recognition, mexican spanish, HTK, CMU Sphinx-III, grammar based system**

I. Motivation

Speech resources for languages different to English are scarce in most of the cases. People are used to follow online tutorials or books in English to learn specific technologies that they need in their own academic fields. Maybe some academic fields do not need to be adapted to a particular language because the terminology is well accepted and understood to be in English. It is not the same when we are in the field of phonetics and language technologies.

There are several acoustical phenomena you can find, depending on the language you are working with. For example, in tonal languages like Chinese, the same word means different things for different intonations. In Spanish the intonation is used to distinguish between questions and affirmations but the meaning of all the words remains the same. One other problem when you work with scarce resource languages is the standardization. People create databases with different phonetic alphabets or transcription rules because they do not have a standard rule for doing that. However, another big problem at Mexico is that most of the databases in Mexican Spanish are not available for free use (see [1] and [2]), or the developer groups have been disappeared (for example the TLATOA Group, see [3]), sometimes it is not clear enough if the database is for free use or not because you can not find a link to download it ([4], [5]), or simply the creators have legal issues with the copyrights so they can not share the database (is the case of [6]), this can only mean that when you read a paper presenting one of these databases you do not count with the physical data to do experiments and understand exactly what are they showing and why.

So, this recognizer belongs to a bigger project called CIEMPIESS-UNAM[1] that was born at UNAM and aims to promote and develop standard and free speech technologies for Mexican Spanish.

II. Introduction

Before the full explanation of the recognizer, it is necessary to explain some terminology for readers from other areas.

Speech Corpus: This is a collection of audio files taken from the real world, provided with their text transcriptions at different levels including the phonetic level. A Speech Corpus (or Speech Corpora) can be also known as the database.

Acoustic Model: This is a statistical representation of the sounds that make up each word existing in the audio recordings of the speech corpus. The recognizer uses then the acoustic model to know how the words are and how to recognize them.

Hidden Markov Models (HMMs): This is a statistical method fairly used to create acoustic models for Automatic Speech Recognition (ASR).

The Hidden Markov Model Toolkit (HTK): Is a toolkit developed at Cambridge University for creating acoustic models for ASR based on hidden Markov models. For more information you can see the HTK Book [7].

CMU Sphinx: This is a group of ASR systems based on HMMs developed at Carnegie Mellon University. The later version is Sphinx-IV coded in java, but the recognizer that we developed is based on Sphinx-III coded in C/C++ [8].

Manuscript received April 24, 2014; revised June 27, 2014.

[1] You can visit the website at: http://odin.fi-b.unam.mx/CIEMPIESS-UNAM/. This website is still under construction but you can download our free 17 hour size database.

doi: 10.12720/ijeee.3.3.220-224

Julius Speech Recognition Engine: This is a real-time speech recognition decoder. Acoustic models for Julius are created using HTK[2].

CIEMPIESS Database: This is a novel 17 hour size open-source speech radio corpus in Mexican Spanish developed by the CIEMPIESS-UNAM Project. The CIEMPIESS corpus was designed to be used in the field of ASR for creating robust acoustic models ("in press" [9]).

Computational Phonetic Alphabet: This is a set of symbols that represents the phonemes of a certain language. These symbols must be coded into a computational character encoding scheme like ASCII, UNICODE, etc.

Pronouncing Dictionary: This is nothing but a list of words with their phonetic transcriptions. An ASR system uses it to determinate what are the exact symbols that represents the sound of a certain word in a certain language. The phonetic transcriptions must be done with a computational phonetic alphabet.

As we will see, our recognition system depends on HTK to create the acoustic models with data extracted from the CIEMPIESS corpus.

This work is in part inspired in a previous one [10] that explained how to create acoustic models for Mexican Spanish but using the CMU Sphinx-III recognition system.

III. System Overview

An ASR system works always in two basic steps, the first one is known as the training stage and we will call the second one, the test stage.

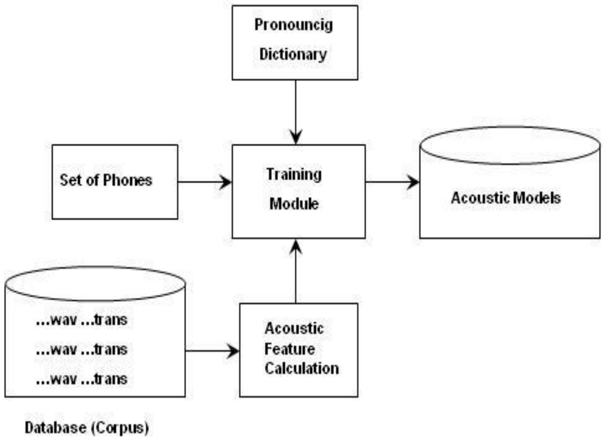


Figure 1. The Training Stage Graphically.

In the training stage the system takes the recordings of the database and converts them into a set of feature vectors. These feature vectors are parametric representations of the audio recordings. The system processes the feature vectors, trying to find repetitive patterns based on the idea that similar words must have similar feature representations. The system then takes the text transcriptions and transforms every word into a sequence of phonemes helped by the pronouncing dictionary. Knowing the phonetic representation of a certain sequence of feature vectors, the system can search patterns in order to create acoustic models for each symbol in the phonetic alphabet. This means that if your phonetic alphabet counts with 30 symbols, the system creates 30 HMMs and uses the information of the next feature vector to update and improve every HMM until the feature vectors are terminated. This is how the system "learns" the acoustic representation of every symbol in the phonetic alphabet and now we can say that the system is "trained". Fig. 1 shows graphically how the training stage is performed.

In the test stage, the system receives input audios in order to recognize (or decode) what are they saying, but this time it does not count with the text transcriptions and the only thing to do, is to convert the input audios into feature vectors. After that, the system searches patterns again on those vectors and compares them with the acoustic models developed at the training stage. If the system finds a match, it will know a specific sequence of phonemes and with the help of the pronouncing dictionary, the system now will know, which word is it. Fig. 2 shows the test stage graphically.

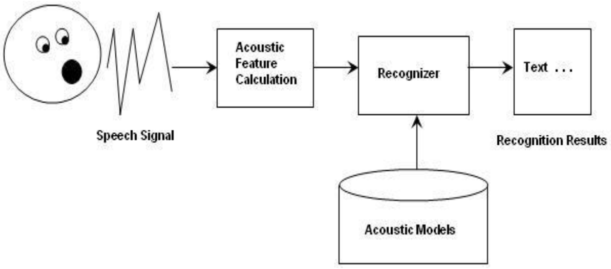


Figure 2. The Test Stage Graphically.

When the system recognizes utterances in real time, we say that is a "live decode", but when the utterances are recorded in several audio files, we say that is a "batch decode". Our system can perform both kinds of decoding. Live decoding is performed with the Julius recognition system and batch decoding is performed with HTK.

The kind of utterances that our system can recognize depends on the input grammar defined by the user. In this case the grammar is nothing but a set of rules that tells the system what are the valid sequences of words that the system expects to receive. The utterances for the training stage do not need to follow any grammar rule, but it must contain all the phonemes of the phonetic alphabet at least once. When the train utterances contain several repetitions of all the phonemes of the phonetic alphabet it is said that the training database is "phonetically balanced".

Finally, we say that our system is "speaker dependent" because the users that are going to use it, need to train it with their own voices. For a different user, the system will lose performance.

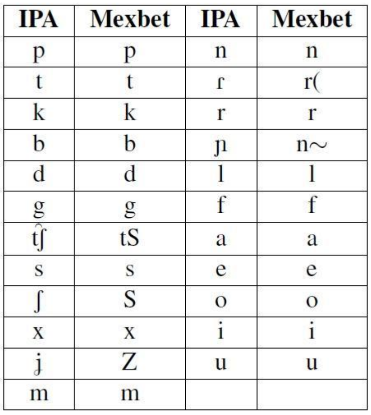
IV. The Lexicon

The lexicon development process begins with the appropriate choice of a phonetic alphabet containing all the phonemes for the language you are going to recognize.

[2] http://julius.sourceforge.jp/en_index.php

For this purpose we can choose between some alphabets created for several variants of the Spanish language like “SAMPA” [11] or “Worldbet” [12], but in this case we decided to use the computational phonetic alphabet called Mexbet [13]. Mexbet was created at UNAM in 2004. It was specially design for the Mexican Spanish of the center of Mexico and it has been demonstrated that it works quite well [14].

TABLE I. IPA-MEXBET EQUIVALENCES



IPA	Mexbet	IPA	Mexbet
p	p	n	n
t	t	ɾ	r(
k	k	r	r
b	b	ɲ	n~
d	d	l	l
g	g	f	f
t͡ʃ	tS	a	a
s	s	e	e
ʃ	S	o	o
x	x	i	i
ɟ	Z	u	u
m	m		

TABLE II. ACOUSTIC CHARACTERISTICS OF MEXBET SYMBOLS

Point	Manner	Mexbet Symbol
	Consonants	
Labial	Occlusive (unvoiced)	p
Labial	Occlusive (voiced)	b
Labial	Nasal	m
Labiodentals	Fricative (unvoiced)	f
Dental	Occlusive (unvoiced)	t
Dental	Occlusive (voiced)	D
Alveolar	Fricative (unvoiced)	S
Alveolar	Nasal	N
Alveolar	Rhotic	r(
Alveolar	Rhotic	R
Alveolar	Lateral	L
Palatal	Affricate (unvoiced)	tS
Palatal	Fricative (unvoiced)	S
Palatal	Fricative (voiced)	Z
Palatal	Nasal	n~
Velar	Occlusive (unvoiced)	K
Velar	Occlusive (voiced)	G
Velar	Fricative (unvoiced)	X
	Vowels / Tonic vowels	
Closed	Front	i / i_7
Closed	Back	u / u_7
Half-Open	Front	e / e_7
Half-Open	Back	o / o_7
Open	Central	a / a_7

Mexbet consists in a set 28 phonemes that represents all the sounds that we use in the Spanish of the center of Mexico. Table I shows the Mexbet phonemes and their equivalent IPA phonemes.

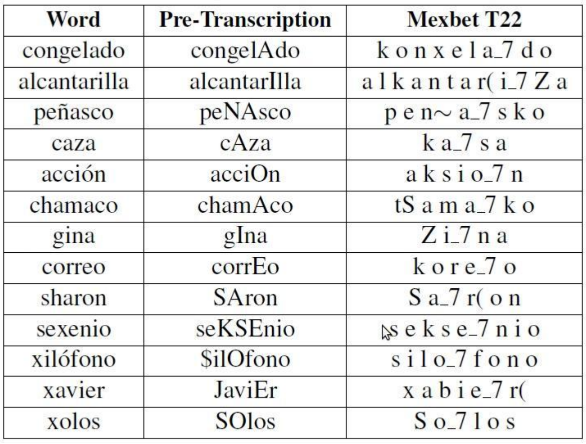
“IPA” is the International Phonetic Alphabet created by the International Phonetic Association. The IPA symbols are standard for every language in the world, but as you can see, the symbols are not ASCII coded. That means that you would have troubles if you try to use this alphabet within Sphinx, Julius or HTK because they do not accept MS Word documents as input files. That is the real reason why we need a computational phonetic alphabet like Mexbet.

The next step is to determine the acoustic characteristics of the phoneme set. This implies some phonetic knowledge that we will not treat here. Instead of that, you can see Table II that shows the acoustic characteristics for the phonemes in Mexbet. This table is really helpful when you want to learn HTK because HTK needs it, but user manuals only show this table for the English language.

The column “Point” informs the articulation point of the phoneme. This means how you have to put your mouth and tongue in order to pronounce the phoneme. The column “Manner” informs where the air of your mouth goes, and how. For example, if the air goes through your nose we say that the phoneme is “nasal”, but when there is no air, we say that the phoneme is “occlusive”.

Is interesting the fact that Sphinx creates this table automatically using something called “automatic phone clustering” (You can read a good example of this for the Spanish language at [15]).

Finally, Fig. 3 shows some examples of Spanish words transcribed with the Mexbet (for more information of how to use Mexbet see [9]).



Word	Pre-Transcription	Mexbet T22
congelado	congelAdo	k o n x e l a_7 d o
alcantarilla	alcantarIlla	a l k a n t a r(i_7 Z a
peñasco	peNAsco	p e n~ a_7 s k o
caza	cAza	k a_7 s a
acción	acciOn	a k s i o_7 n
chamaco	chamAco	tS a m a_7 k o
gina	gIna	Z i_7 n a
correo	corrEo	k o r e_7 o
sharon	SAron	S a_7 r(o n
sexenio	seKSEnio	s e k s e_7 n i o
xilófono	$ilOfono	s i l o_7 f o n o
xavier	JaviEr	x a b i e_7 r(
xolos	SOlos	S o_7 l o s

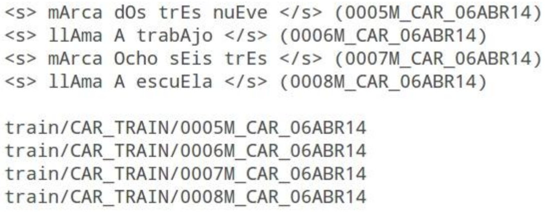
Figure 3. Spanish words transcribed with Mexbet

V. SYSTEM CONFIGURATION

There are various configuration and data files that you have to set up in the “etc” directory, in order to use the system.

Audio Recordings: The system has been design to be compatible with the CIEMPIESS database; nevertheless the current version of CIEMPIESS comes with audio files in the NIST-SPHERE format that is not quite compatible with most of the audio players, so you have to convert them to a MS WAV format. For the test stage, you are obligated to record your own data to play with the system, and you have to be sure that you use the same MS WAV file format at the rate of 16 kHz, 16 bit, mono for full compatibility.

Text Transcriptions: Transcriptions are created in a plain text file with the extension “.transcription” and this file is always associated to another, containing the path for the audio recordings in the current operating system. The latter file comes with the extension ".fileids". You need both files for the training and the test stage. Fig. 4 shows few lines of both files as an example.

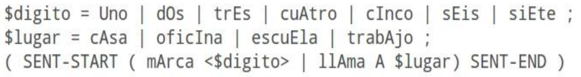


```
<s> mArca dOs trEs nuEve </s> (0005M_CAR_06ABR14)
<s> llAma A trabAjo </s> (0006M_CAR_06ABR14)
<s> mArca Ocho sEis trEs </s> (0007M_CAR_06ABR14)
<s> llAma A escuEla </s> (0008M_CAR_06ABR14)

train/CAR_TRAIN/0005M_CAR_06ABR14
train/CAR_TRAIN/0006M_CAR_06ABR14
train/CAR_TRAIN/0007M_CAR_06ABR14
train/CAR_TRAIN/0008M_CAR_06ABR14
```

Figure 4. Few lines of the ".transcription" and ".fileids" files.

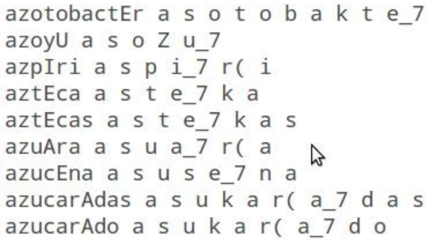
Task Grammar: The task grammar is defined by the user in a file with the extension ".grammar". This grammar is in the HTK format (for more details see the HTK book). Fig. 5 shows an example of how this grammar file is seen.



```
$digito = Uno | dOs | trEs | cuAtro | cInco | sEis | siEte ;
$lugar = cAsa | oficIna | escuEla | trabAjo ;
( SENT-START ( mArca <$digito> | llAma A $lugar) SENT-END )
```

Figure 5. Task Grammar file

CIEMPIESS Pronouncing Dictionary: We provide the CIEMPIESS pronouncing dictionary with the software in the "etc" directory. It is an archive with the ".dic" file extension. This file contains about 53,000 words with their phonetic transcriptions in Mexbet. You can freely change the dictionary but the phonetic transcriptions must match with phonemes listed in the ".phone" file that is also in the "etc" directory. Take into account that you have to choose the words for your task grammar and those words need to be in the dictionary alphabetically sorted. If you don't know how use Mexbet, the best thing you can do is to choose only words contained at the dictionary. Fig. 6 shows some few lines of the CIEMPIESS pronouncing dictionary.



```
azotobactEr a s o t o b a k t e_7 r(
azoyU a s o Z u_7
azpIri a s p i_7 r( i
aztEca a s t e_7 k a
aztEcas a s t e_7 k a s
azuAra a s u a_7 r( a
azucEna a s u s e_7 n a
azucarAdas a s u k a r( a_7 d a s
azucarAdo a s u k a r( a_7 d o
```

Figure 6. Few lines of the CIEMPIESS Pronouncing Dictionary

Control Scripts: The scripts that glue the whole system together are written in Python, and that is why this recognizer is so compatible with several operating systems. If you have correct input files in the "etc" directory you can now perform a recognition experiment. First, you need to convert your audio recordings into feature vector files using the script "make_feats.py". Then, you perform the training stage with the script "RunAll.py" and finally you can recognize your test files with the script "/decode/slave.py". For live decoding you also need to manipulate the task grammar and convert it into the Julius format. You can see the "sample.grammar" and "sample.voca" files in the "scripts_py/live_decode" directory to figure out how to convert the grammar format or see the Julius documentation online.

VI. Experiments

We performed one simple recognition experiment in order to detect possible programming bugs. We took 407 audio files from the CIEMPIESS database for the training stage and we recorded 30 utterances for the test stage. The resulting Table III shows the word error rate (WER) and the sentence error rate (SER) when the system is trained (a) with the CIEMPIESS database and tested with the 30 utterances, and (b) when the system is trained and tested with the same utterances.

TABLE III. Recognition Results

	Train	Test	% WER	% SER
(a)	CIEMPIESS	30 Utt.	19.1	46.67
(b)	30 Utt.	30 Utt.	3.37	16.3

As we can see the results in **(b)** are the expected and that means that the system is working fine. The results in **(a)** are not as good, but they are still good taking into account that the system is speaker dependent, and the utter in the training utterances are different to the voice in the test.

This means that users, who correctly train the system, will have better results.

VII. Conclusions

What we have created is an open-source system that can perform live and batch decoding, that is compatible with most of the current operating systems and people can use it to perform accurate speech recognition experiments easily, and at no cost.

The design of language model is novel for Mexican Spanish language, and it involves particularities of Spanish.

VIII. Further Work

We will improve the system for speaker independent recognition and we have to perform different tools for doing better and more accurate configurations in both the training and testing stages.

We have the challenge to develop good tools for promoting the field of speech recognition in Mexico and we are compromised with the open source software that is free and easy to use.

In the future, we also want to perform experiments with indigenous dialects of Mexico like Nahuatl o Huastec. We can see some pioneer works of this in [16]-[18].

Acknowledgment

The authors wish to thank UNAM PAPIIT/DGAPA project IT102314, CEP-UNAM and CONACYT for financial support.

REFERENCES

[1] A. Moreno, R. Comeyne, *et al.*, "SALA: SpeechDat across Latin America. Results of the First Phase," in *Proc. LREC. European Language Resources Association*, 2000.

[2] J. Llisterri, "Las tecnologías del Habla para el Español," in *Proc. Fundación Española para la Ciencia y la Tecnología*, 2004, pp. 123-141.

[3] I. Kirschning, "Research and development of speech technology and applications for mexican spanish at the tlatoa group," in *Proc. CHI'01 Extended Abstracts on Human Factors in Computing Systems*, 2001, pp. 49-50.

[4] E. Uraga and C. Gamboa, "VOXMEX speech database: Design of a phonetically balanced corpus," in *Proc. LREC. European Language Resources Association*, 2004.

[5] J. M. Olguín-Espinoza, P. Mayorga-Ortiz, H. Hidalgo-Silva, L. Vizcarra-Corral, and M. L. Mendiola-Cárdenas, "VoCMex: A voice corpus in Mexican Spanish for research," *International Journal of Speech Technology*, vol. 16, no. 3, pp. 295-302, 2013.

[6] L. A. Pineda, L. Villaseñor, J. Cuétara, H. Castellanos, and I. López, "DIMEx100: A new phonetic and speech corpus for mexican spanish," *Lecture Notes in Computer Science*, vol. 3315, pp. 974-984, 2014.

[7] D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland, *The HTK Book (for HTK Version 3.4)*, Mar. 2006.

[8] A. Chan, E. Gouvea, and R. Singh, *The Hieroglyphs: Building Speech Applications Using CMU Sphinx and Related Resources*, Mar. 2007.

[9] C. D. Hernández-Mena and J. A. Herrera-Camacho, "CIEMPIESS: A new open-sourced mexican spanish radio corpus," in *Proc. LREC. European Language Resources Association*, 2014.

[10] A. Varela, H. Cuayáhuitl, and J. A. Nolazco-Flores, "Creating a mexican spanish version of the CMU sphinx-III speech recognition system," *Lecture Notes in Computer Science*, vol. 2905, pp. 251-258, 2003.

[11] J. C. Wells, "SAMPA computer readable phonetic alphabet," in *Handbook of Standards and Resources for Spoken Language Systems*, chapter Part IV, Section B, 1997.

[12] J. L. Hieronymus, "ASCII phonetic symbols for the world's languages: Worldbet," Technical report, 1994.

[13] J. Cuétara-Priede, "Fonética de la ciudad de México. Aportaciones desde las tecnologías del habla," Msc. Thesis in spanish linguistics (in Spanish), Universidad Nacional Autónoma de México, México, 2004.

[14] L. A. Pineda, H. Castellanos, J. Cuétara-Priede, L. Galescu, J. Juarez, *et al.*, "The corpus DIMEx100: Transcription and evaluation," in *Language Resources and Evaluation*, 2009.

[15] R. de Córdoba, J. Macías-Guarasa, J. Ferreiros, J. M. Montero, and J. M. Pardo, "State clustering improvements for continuous HMMs in a Spanish large vocabulary recognition system," in *Interspeech*, John H. L. Hansen and Bryan L. Pellom, Ed. 2002.

[16] J. A. Nolazco-Flores, L. R. Salgado-Garza, and M. Peña-Díaz, "Speaker dependent ASRs for huastec and western-huastec náhuatl languages," *Lecture Notes in Computer Science*, vol. 3523, pp. 595-602, 2005.

[17] A. Hernández and A. Herrera, "Acoustics of Nahuatl at Tzinacapan," in *Proc. 12th Congreso Mexicano de Acústica*, Nov. 2007.

[18] S. Suárez-Guerra, J. L. Oropeza-Rodríguez, J. C. Flores-Paulin, and L. P. Sánchez-Fernández, "Digit recognition in the náhuatl language: An evaluation using various recognition models," *IEEE Computer Society*, pp. 143-147, 2010.

Carlos D. Hernández-Mena was born in Mexico City in 1983. He received his Bachelor's degree in the area of Electronics and Communication Engineering from The National Polytechnic Institute located in Mexico City in the year 2006. He received his M.E degree in the area of speech recogniton from the National Autonomous University of Mexico (UNAM) in 2010. His current phD research includes continuous speech recognition, microcomputers and applied phonetics. He assisted to the "Verano Científico Tec-Profesor" summer school in the Monterrey Institute of Technology and Higher Education (ITESM) located at Monterrey, Mexico in 2013. Nowadyas Carlos teaches microprocessors at UNAM. Prof Hernández-Mena is current member of the Acuctical Society of America (ASA), the Institute of Electrical and Electronics Engineers (IEEE) and the International Speech Communication Association (ISCA).

J. Abel Herrera-Camacho received degrees in Mechanical-Electrical Engineering, M. S. Electronic Engineering, and the Ph.D. Engineering, from Universidad Nacional Autonoma de Mexico (UNAM), Mexico, in 1979, 1985, and 2001, respectively, the PhD. was with support of the University of California in Davis. He did a postdoctoral research in 2001 at Carnegie Mellon University, and a sabbatical research at USC. He is author from more than 50 scientific papers on codification, recognition, and synthesis. He is coauthor of the book *Linear Algebra, theory and exercises*, printed 9 times from 1986 yto 2009. Currently, he is the director of Speech Laboratory in the faculty of Engineering of UNAM.