Qüestió - V. 8, n.º 4 (desembre 1984) pp. 179-196

AN ISOLATED WORD RECOGNITION SYSTEM BASED ON A LOW-COMPLEXITY PARAMETRIZATION PROCEDURE

H. RULOT, E. VIDAL, F. CASACUBERTA
UNIVERSIDAD DE VALENCIA

It is presented in this paper an Isolated Word Recognition System which uses a parametrization scheme based on the two-level clipped signal Autocorrelation Function. The system prototype runs on a 64 kby. partition of a general-purpose minicomputer with quite small specific hardware requeriments and, for moderate sized dictionaries (≲40 words), gives 95-98% recognition rates with response times better than two times real time. The system uses classical Dynamic Programming word-matching, and its main aimed applications are the implementation of low-cost microprocessor-based speech devices and the incorporation of Isolated Word Recognizers among the software utilities of general-purpose (mini-micro) computers. The main system features are discussed and formal evaluation tests are presented.

Keywords: ISOLATED WORD. ON-LINE RECOGNITION. AUTOCORRELATION ANALYSIS.

1. INTRODUCTION.

A parameter-extraction procedure is needed in all Speech Recognition Systems both to reduce the large amount of information contained in speech signals and to adequate this reduced information to the recognition methods to be used.

Classical parametrization techniques require in general, a large computational effort in software-oriented implementations, or specific hardware devides (i.e.channel vocoder) if real-time operation is desired. This is in fact one of the complexity (and cost) factors of the nowdays Isolated Word Recognizers, and the technological solutions usually adopted point towards implementation of VLSI Signal Processors.

Nevertheless, for moderate-requirement applications, the use of alternative methods can lead to simplified Speech Recognition devices. Under this point of view an Isolated Word Recognition System is presented in this paper, which uses a parametrization scheme based on the two-level clipped signal Autocorrelation Function. The system prototype, designed for evaluation and demonstration purposes, runs on a 64 Kby. partition of a general-purpose minicomputer without specific hardware, and for moderate sized dictionnaries (≲40 words), gives 95-98% recognition rates with response times better than two times real times.

The system uses classical Dynamic Programming word-matching, and its main aimed applications are the implementation of low-cost microprocessor-based speech devices and the incorporation of Isolated Word Recognizers among the software utilities of general-purpose (mini-micro) computers.

The remainder of this paper is arranged as follows: Section-2 is devoted to the discussion of the proposed parametrization method. Section-3 describes the actual real-time acquisition-and-parametrization procedure. In Section-4 some details about the word-boundary refinement are given. Section 5 and 6 are concerned with the Dynamic-Programming based word-matching and Dictionary-building procedures. Section-7 presents some of the obtained performance results, and in Section-8 these results are discussed which lead to

- H. Rulot, E. Vidal, F. Casacuberta - Centro de Informática de la Universidad de Valencia - Dr. Moliner, s%n. Burjasot - Valencia.

- Article rebut el Octubre de 1984.

the concluding remarks.

2. THE PARAMETER SET

The Autocorrelation Function (A.F.) of a discrete signal $s(\ell)$ observed through a finite-length window $w(\ell)$ (Short-time Autocorrelation Function) can be defined as /3/:

$$(2.1)\quad R(m) = \sum_{n=-\infty}^{\infty} s'(n)\, s'(n+k);\quad s'(1)=s(1)w(1)$$

The properties of this function, along with its direct existing relations with other classical parameters /1/, /2/, /3/, allow to propose it as a valid parameter set for speech recognition.

The large amount of computation needed for evaluating products in (2.1) can be dramatically reduced if a two-level quantization is applied to the windowed signal (fig. 1), leading to a modified function $\tilde{R}(m)$ which will be called the "Two-level Autocorrelation Function" (TLAF). In this case the products can be converted to exclusive-or functions or alternativelly to single comparations, allowing real-time operation on general purpose mini or microcomputers.

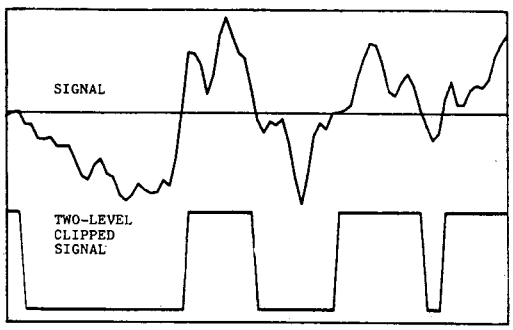

Figure 1: Clipping process.

Some of the original properties of the Autocorrelation Function disappear with this strong clipping; but anothers are conserved or assume modified interpretations /1/. So the phase independance and the enhancement of the signal main periodicities, are bassically conserved; while the interpretation of R(0) as the windowed signal energy is converted to the identity of $\tilde{R}(0)$ with the constant window length, and the relations of the successives diferences of R(m) at the origin with the moments of the squared magnitude spectrum /4/ are reduced to the identity of $\tilde{R}(0)-\tilde{R}(1)$ with the zero crossing of the windowed signal.

The error introduced by the clipping can be theoretically predicted for random or periodic signals /5/, /6/. For actual speech signals, experiments have been made /1/ which show that the effects of this error can be allowed if a large enough time window (20 mseg) is used. Fig. 2 shows the comparation of the regular and clipped A.F. for some of the worst-case speech segments found, all them sampled at 12800 hz., and windowed by a 20 mseg rectangular window.

The signal level for quantization clipping is theoretically assumed to be zero. However, in practice this level can be dinamically determined on the basis of a previously measured environnemental noise, in order to minimize the effects of this noise onto the computed TLAF.

The actual validity of the two-level-clipped A.f. for parametrization of speech signals have been experimentally confirmed /1/. Fig.3 summarizes the experimental results. The graphics show 64 values of the TLAF for different stationary segments, sampled at 12800 hz., of the most common spanish phonemes. Two consecutive 20 mseg. frames of each segment are displayed showing the great stability of results with respect to analysis window time shifts. The different segments are rather well discriminated, even with less than half of the TLAF displayed points, and the points actually needed for adequate discrimination may be further reduced by merely decreasing the sampling frequency. With a frequency of 6400 hz., which still retain most of the psychoacoustic speech information, less than 16 TLAF values are needed, which is comparable with the most commonly used prameter sets.

It must be pointed out here that TLAF parameters are absolutely normalized to the window length N, being $\tilde{R}(0)=N$. This can be thought as an advantage, but also as a possible drawback, because the same importance would be given to the loud speech segments that to the soft noisy segments. To overcome

REGULAR AUTOCORRELATION FUNCTION (A.F.) TWO-LEVEL SIGNAL CLIPPED A.F. (TLAF)

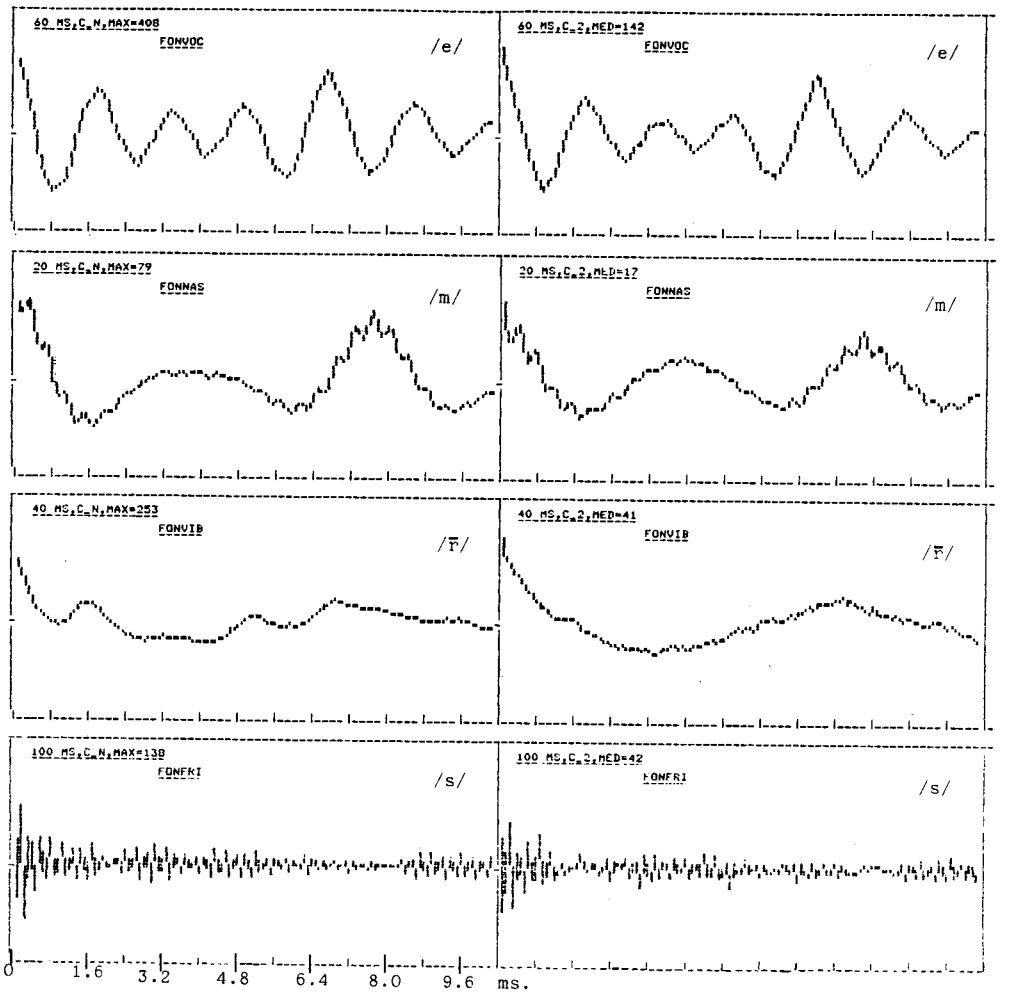

Fig. 2. Comparation of regular and clipped Autocorrelation Functions for some worst-case speech segments.

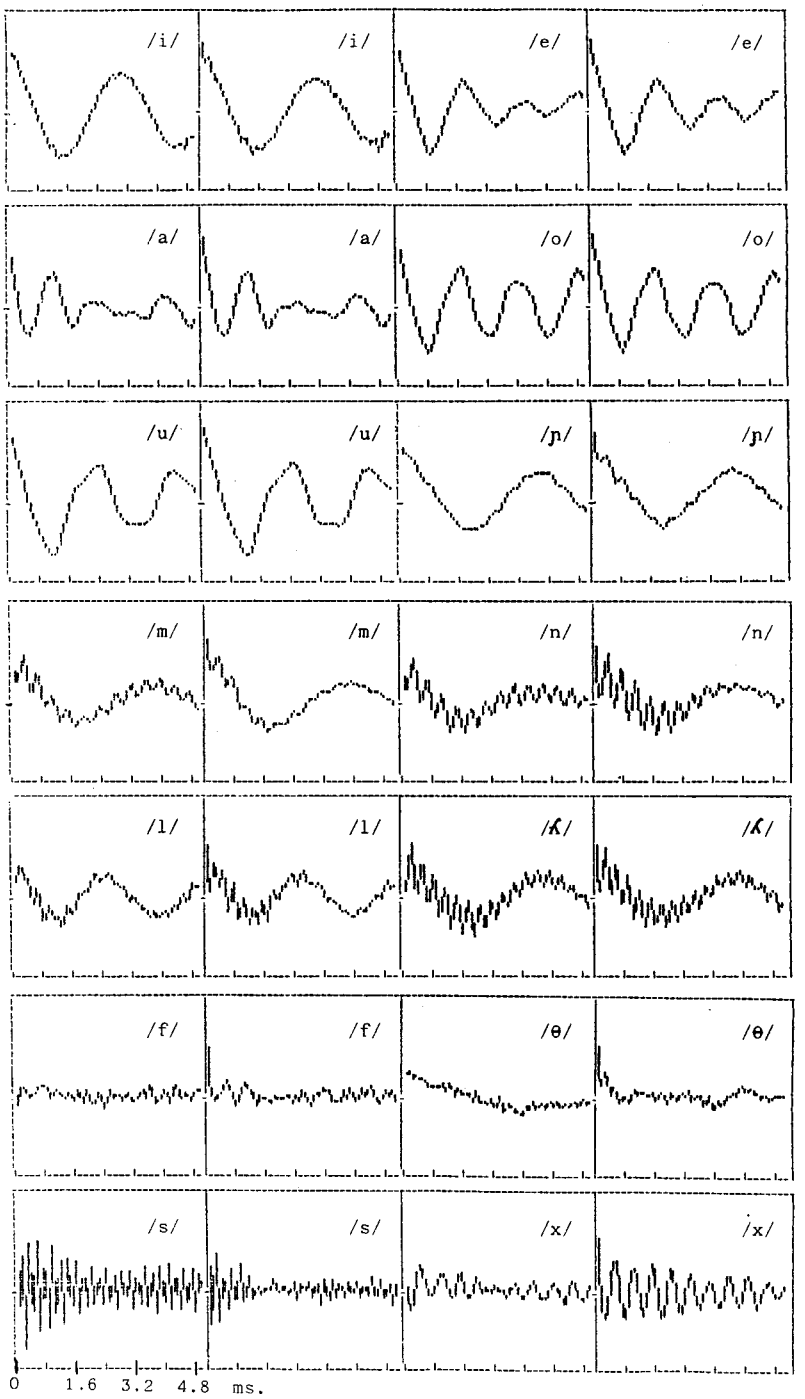

Fig. 3: TLAF of some of the most common spanish stationary phonemes. Two consecutive 20 ms. frames are displayed for each 12800 Hz. sampled phoneme segment.

this problem the meaningless $\tilde{R}(0)$ value has been replaced by the signal short-time Average Magnitude (amplitude) A:

$$(2.2) \quad A=\sum_{n=-\infty}^{\infty} |s'(n)| \; ; \; s'(k)=s(k).w(k)$$

which will be conveniently used in distance computation.

Finally, application of 6 db/8 preenphasis to the signal prior to quantization, has been incorpored as an optional feature for parameter extraction. This operation supply TLAF patterns which seem rather meaningless through visual inspection, but which show a substantial overall improvement of numerical discrimination-power. This fact confirms once again the importance for speech recognition of the second and third formants /10/ which are often lost by the two level quantization without preenphasis.

With the above dis sed main modifications, the proposed M-dimensional parameter vector for each N-point-sized rectangular window, is defined as follows:

(2.3)

$$\tilde{R}(0)=\sum_{n=0}^{N-1} |s'(n)|$$

$$\tilde{R}(m)=\sum_{n=0}^{N-1} \text{sign}(s'(n)) \; \text{sign}(s'(n+m)) \qquad m=1\ldots M-1$$

Figure 4 shows a graphical representation of one of such vectors corresponding to a 20 ms. segment of the english phoneme /I/.

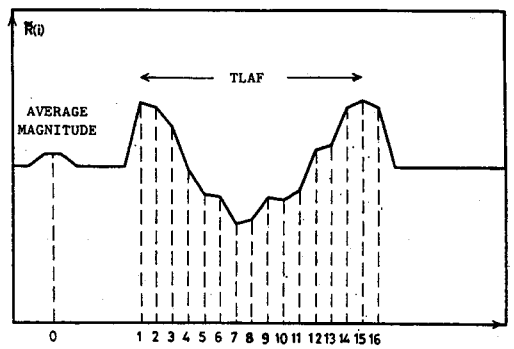

Fig. 4: Graphic display of the proposed TLAF 16-dimension parameter vector for the english phoneme /I/ of /naIt/.

3. ACQUISITION AND TLAF COMPUTATION.

A summarized exposition of the different algorithms and devices for TLAF computation has been presented in /1/.

In the presented system the M<32 (or M<8) first values of the non-overlapping windowed TLAF are computed in real-time by a 16 bits mini (or micro) computer ECLIPSE C-350 (or MP-100) for sampling frequencies up to 12800 Hz (or 6400 Hz).

The algorithm, presented in fig. 5, is based on (2.3) and combines the TLAF computation with the signal acquisition, rough-detection of word-boundaries, and "quality-control" of the acquired signal. The storage of the past M one-bit samples, needed for TLAF computation, is provided for by the processor accumulators working as an algorithm-controlled shift-register. The one-bit quantization is done by assuming a high-value (1) if the sample value exceedes a previously estimated noise-level threshold, and a low-value (0) otherwise. The word-boundary detection and quality-control are based on the values of

$$s'(k)=\begin{cases} s(k): \text{ no preenphasis} \\ s(k)-s(k-1): \text{ 6db/8 preenphasis} \end{cases}$$

the mean signal amplitude, the durations of signal and silence, and the rate of saturated samples. In the hardware configuration utilized, the sampled signal is supplied by a standard 12 bits general-purpose A/D converter; however, for the implementation of a specific device, the A/D requirements can be substantially reduced.

The algorithm works in an "open-microphone" way, storing the real-time computed TLAF vectors in a circular buffer; this process goes on until the signal amplitude has reached an amplitude-threshold, and then the acquisition and parametrization continues into a linear

```
Data: MAX_WAIT_TIME, AMPL_THRESH, SILENCE_TIME_THRESH, TLAF_ORDER,
      WINDOW_SIZE, PREEMPHASYS, NOISE_LEVEL.

begin
  make ciclic_buff_pointer = 0.
  make wait_time = 0.
  repeat
    parametrize(CICLIC_BUFF, ciclic_buff_pointer).
    ciclically increment ciclic_buff_pointer.
    increment wait_time.
  until wait_time > MAX_WAIT_TIME or average_magnitude > AMPL_THRESH.
  if wait_time > MAX_WAIT_TIME then error: 'Waiting too long'.
  make buff_pointer = SILENCE_TIME_THRESH + 1.
  make silence_time = 0.
  repeat
    parametrize(BUFFER, buff_pointer).
    if average_magnitude > AMPL_THRESH then make silence_time = 0
                                       else increment silence_time.
    increment buff_pointer.
  until buff_pointer > buff_size or silence_time > SILENCE_TIME_THRESH.
  if buff_pointer > buff_size then error: 'utterance too long'.
  move (orderly) CICLIC_BUFF to BUFFER[1..SILENCE_TIME_THRESH].
end.


procedure parametrize(BUFF, pointer).
{SHIFT and AUX_SHIFT are TLAF_ORDER bits long shift-registers}
  make average_magnitude = 0.
  make read_samples = 0.
  repeat
    make point = pointer.
    make AUX_SHIFT = SHIFT.
    increment read_samples.
    read_one_sample.
    increment average_magnitude by |sample|.
    if sample > NOISE_LEVEL
    then repeat
           increment point.
           left shift AUX_SHIFT.
           if carry = 0 then increment BUFF[point].
         until point = pointer + TLAF_ORDER.
         make carry = 0.
    else repeat
           increment point.
           left shift AUX_SHIFT.
           if carry = 1 then increment BUFF[point].
         until point = pointer + TLAF_ORDER.
         make carry = 1.
    end if.
    left shift SHIFT.
  until read_samples = WINDOW_SIZE.
  make BUFF[pointer] = average_magnitude.
end parametrize.


procedure read_one_sample.
  wait for sampling_clock_pulse.
  acquire one sample from A/D conversor.
  if PREEMPHASYS then
     decrement sample by last_read_sample.
     make last_read_sample = sample.
  end if.
end read_one_sample.
```

Fig. 5: Signal acquisition and TLAF computation algorithm.

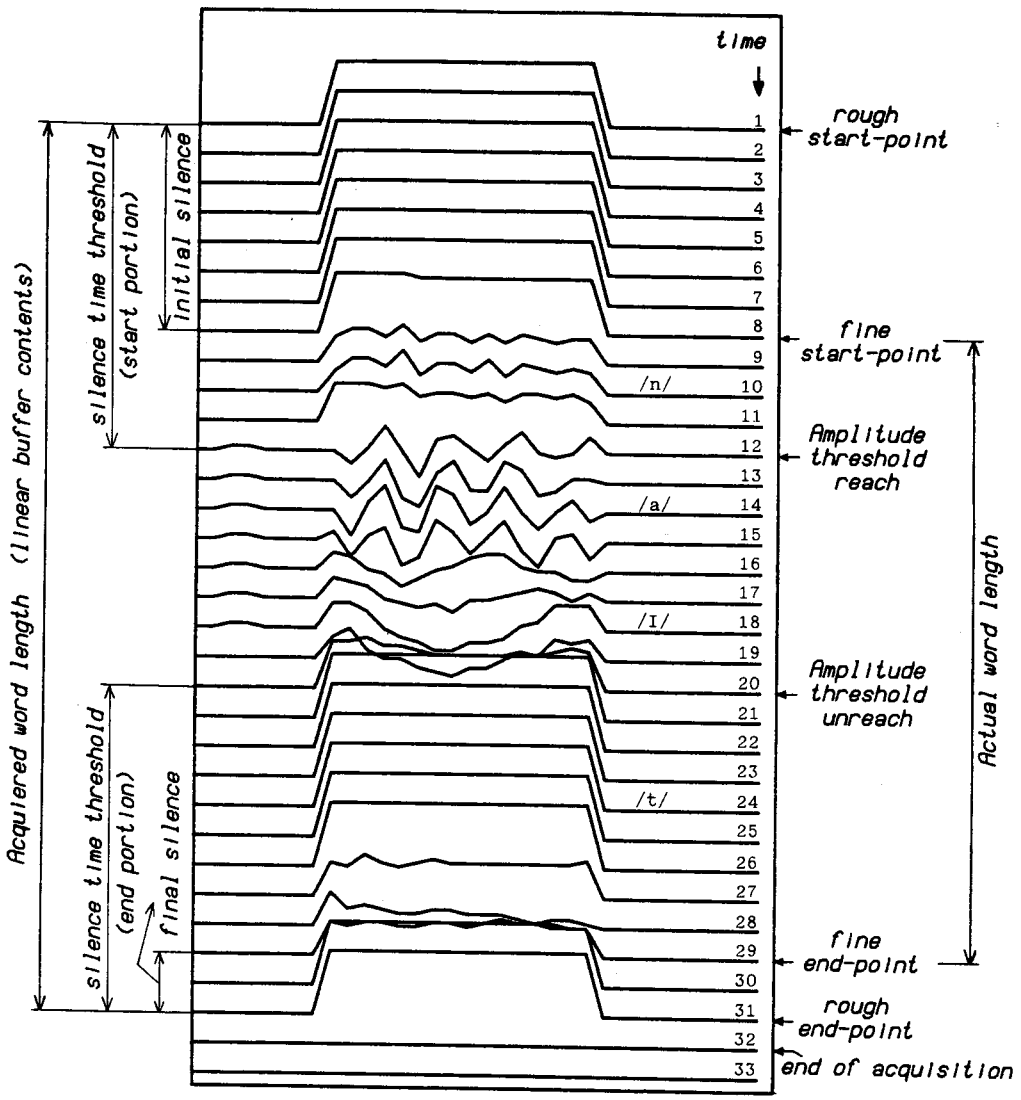

Fig. 6: Word boundaries refinement of the english word /naIt/ ("night"). The TLAF patterns are displayed in the same way as in figure 4.

buffer until the silence duration has reached a silence-time-threshold. The last portion of the circular buffer is then orderly copied to the linear buffer, and from this point on, the control is transfered to the recognition algorithm with all the TLAF parameters stored in the linear buffer and the additional quality-control information in auxiliary registers.

4. END-POINTS REFINEMENT.

Though rough amplitude-based word-boundaries are detected during the acquisition-and-parametrization phase, finer boundaries must be found in order to improve the recognition performance.

In this system, the refinement is performed on the basis of matching a TLAF silence-template with the TLAF patterns of initial and final portions of the acquired word. This matching is made through the use of the same TLAF metric defined for Dynamic Programming word matching, which will be discussed in next section.

The refinement method must prevent the system from loosing the possible initial and final weak speech-segments, and specially those of stop releases. To achieve this goal, the matching sweep is performed on both word-ends in the back to forth direction, until a TLAF pattern close enough (appart enough) to the silence-template is found in the start-portion (end-portion). Figure 6 illustrates this procedure for the English word "night" (/naIt/).

5. TIME-WARPING.

A quite classical recognition method has been adopted in the Isolated Word Recognition System presented in this paper. In this method the spoken words are assumed to be represented by strings of parameter-vectors (TLAF), and the recognition problem is reduced to a simple pattern-matching problem once an adequate time normalization procedure has been introduced. This procedure must take into account the non-linear durational variations of the different segments of speech, and may impose only physical restrictions of continuity, monotonocity, etc. ... to the time evolution of the parameters.

The time- normalization is achieved through the use of an "optimal" time-warping function which map the pairs of matching parameter-vectors corresponding to both words to be compared /7/, /8/. The "inter-word distance" or dissimilitude degree between both compared words is then defined as a ponderate summation of the local distances between the pairs of parameter-vectors related by the warping function.

Both "inter-word distance" computation and warping function finding can be conveniently carried out by Dynamic Programming techniques /7/, /8/. The algorithm used in this work is that of symmetric form with non slope constraints and parallel unit-slope straigth lines as computational window. This algorithm can be formally specified by the recursive formula

$$(5.1)\quad g(i,j)=d(i,j)+\min[g(i,j-1),\ g(i-1,j),\ g(i-1,j-1)]$$

where i,j represent the parameter-vector string indexes of both words being compared, g(i,j) the"minimum-effort" function to fit vectors up to i with vectors up to j, and d(i,j) is the local parameter-vector metric mentioned above. For the TLAF parameter-vectors used in this work it has been used the Hamming distance defined as:

$$(5.2)\quad d(i,j)=\sum_{k=M_1}^{k=M_2} |R_i(k)-R_j(k)| \ ;\ 0<M_1<M_2<32$$

where M_1 and M_2 determine the range of TLAF values involved in the computation. Other more complex distance definitions has been tested, but the performance improvement did not justify the increment of computational load.

6. DICTIONARY BUILDING AND RECOGNITION.

The time-warped "inter-word distance" introduced in last section is used both in Dictionary-building and Recognition tasks.

A dictionary is composed of a directory containing the names of the words and the pointers to the word-templates, which are made-up of parametric representations of the words. In order to obtain a good template for a given word, it is convenient to select the "best" among a set of patterns of the same word. This "optimal" selection involves both getting the convenient patterns under different speaker/environnement conditions, and actually searching for the "best" among them. Some methods have been proposed to find an arquetype template to represent a set of same-word-patterns /8/, /9/. The optimization criterium here adopted consist of selecting the pattern which gives the minimum sum of "inter-word distances" to the other patterns.

The dictionary building subsystem has been augmented with evaluative and demonstrative capabilities such as graphic display of the TLAF patterns, etc. It is included also a dictionary evaluation feature (again based on the "inter-word distance") which supplies a figure of "dictionary discriminability" along with a matrix of dissimilitudes between all dictionary words.

Once a dictionary is build-up, the recognition procedure consists just of comparing each uttered-word (represented by its TLAF vector string) with all dictionary templates. The recognized word name will be the name of the template with minimum "inter-word distance" to the uttered word. The recognition subsystem has four operation modes, depending on the verbosity and the way of returning the results of recognition and/or signal quality-control. It can supply, for each uttered word, an "evidence figure" which aims to give a measure of the worthiness of the recognition result. This "evidence figure" has been heuristically defined and experimentally validated as:

$$(6.1) \qquad e = \frac{\inf(\{D-\inf(D)\})}{\inf(D)} - 1$$

where D is the set of inter-word distances between the uttered word and the dictionary templates.

7. RECOGNITION EXPERIMENTS.

Several system constants, as well as environnement and/or using conditions may affect the system performance in a way which is not easily predictible. The most importants among them can be classiffied as follow:

* Acquisition constants:
  - -sampling frequency.
  - -window size (or subsampling frequency).
* Parameter set constants:
  - -number of parameter used (TLAF order).
  - -using or not the first parameter (average magnitude).
  - -using or not preemphasis.
* Time-warping constants:
  - -time-warping window size.
* Environnement conditions.
  - -signal-to-noise ratio.
* Dictionary-dependent conditions:
  - -dictionary size.
  - -average word size.
* Speaker conditions:
  - -individual way of speaking
  - -speaker classes (male-female, child-adult).

Some other constants, such as acquisition and signal quality thresholds, boundary refinement constants, etc., was fixed previously, and their optimal values has been utilized throughout all the experiments described b. . Some of these constants, however, has been adequatelly modified in some of the signal-to-noise experiments in order to obtain the optimal results when the hardly poor signal quality were refused by the quality-control algorithm and/or made the boundary refinement algorithm to fail.

In order to estimate the useful ranges of all the above listed system parameters, a lot of recognition experiments has been carried out, in which all the constants and conditions has been systematically varied. The main results of these experiments are summarized in fig.7 through fig. 12 and table 4.

The speech material used for the results of fig. 7 to 10 consists of the 10 spanish digits (table 1), isolatelly uttered by a single male speaker. An ammount of 13 repetitions (130 words) of the dictionary has been utilized; three of them for dictionary building and the

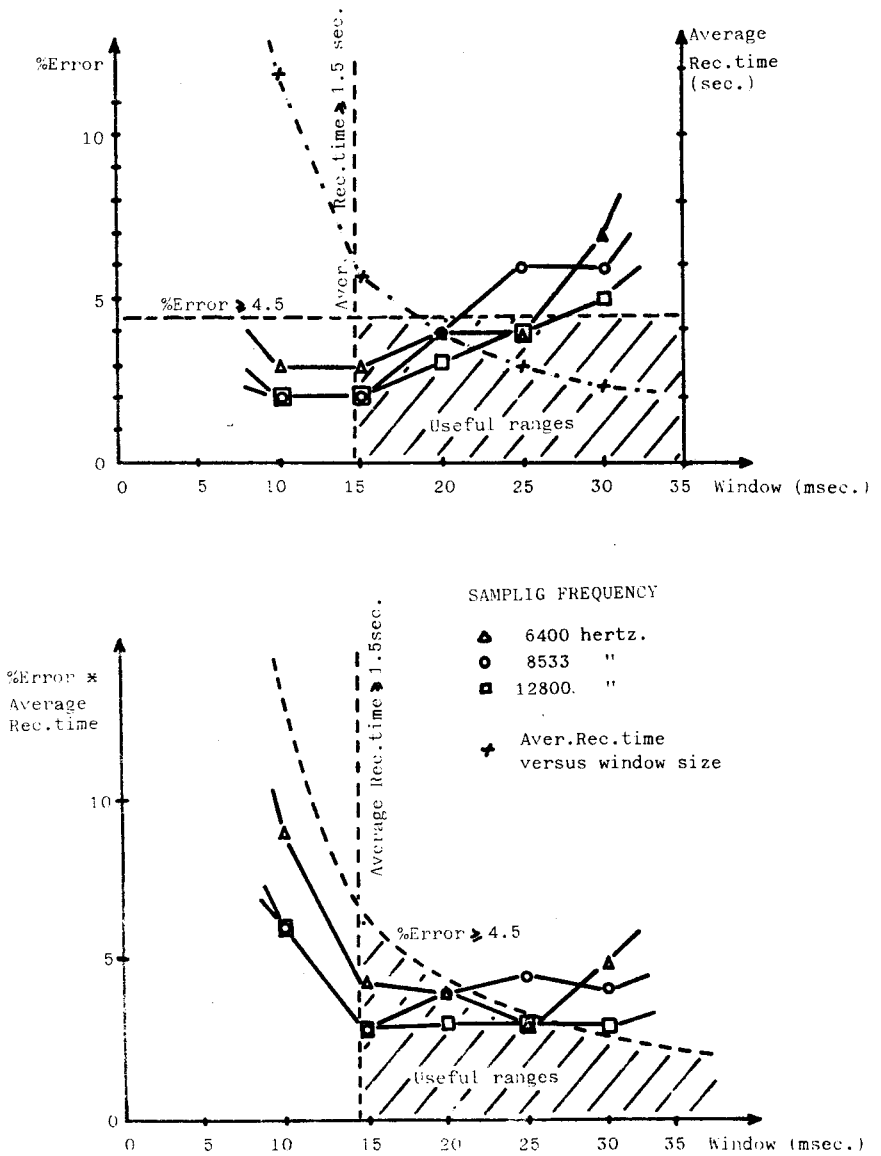

Fig. 7: System performance versus acquisition constants. Time Warping Window=160 msc. Number of parameters =16. Using average magnitude=yes. Using preemphasis=yes.

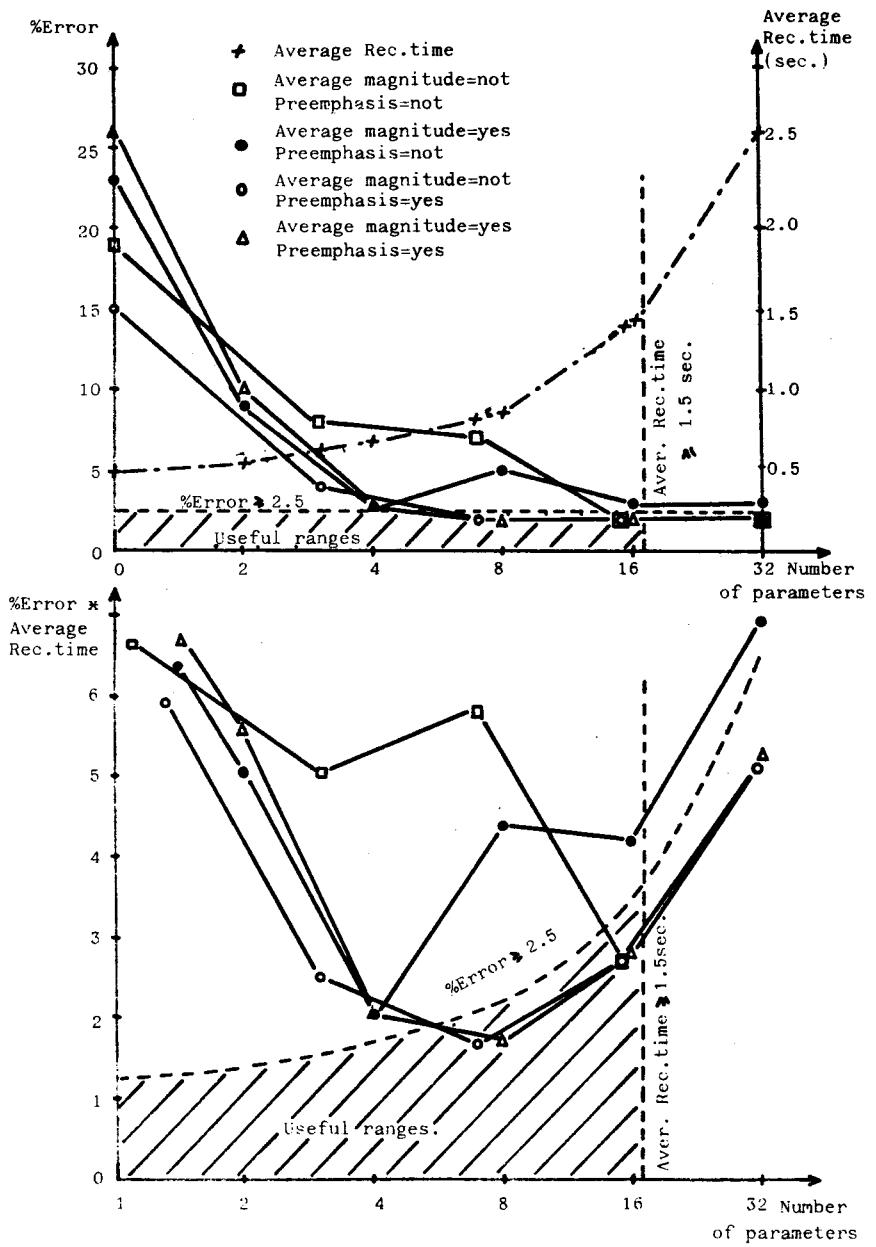

Fig. 8: System performance versus parameter set constants. Sampling frequency=8533 Hz. Window size=15 msec. Time warping window=150 msec.

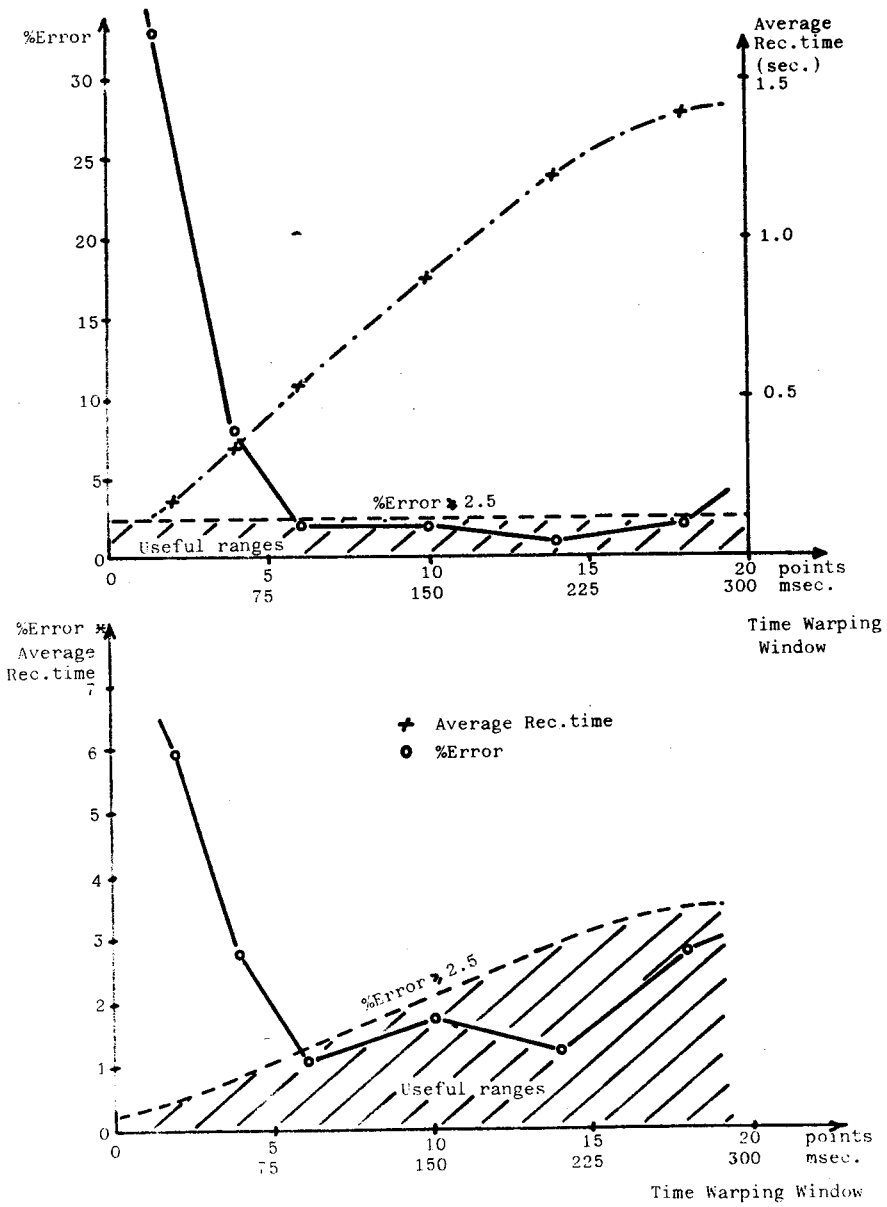

Fig. 9 : System performance versus time warping constants. Sampling frequency=8533 Hz. Window size=15 msec. Number of parameters=8. Using average magnitude=yes. Using preemphasis=yes.

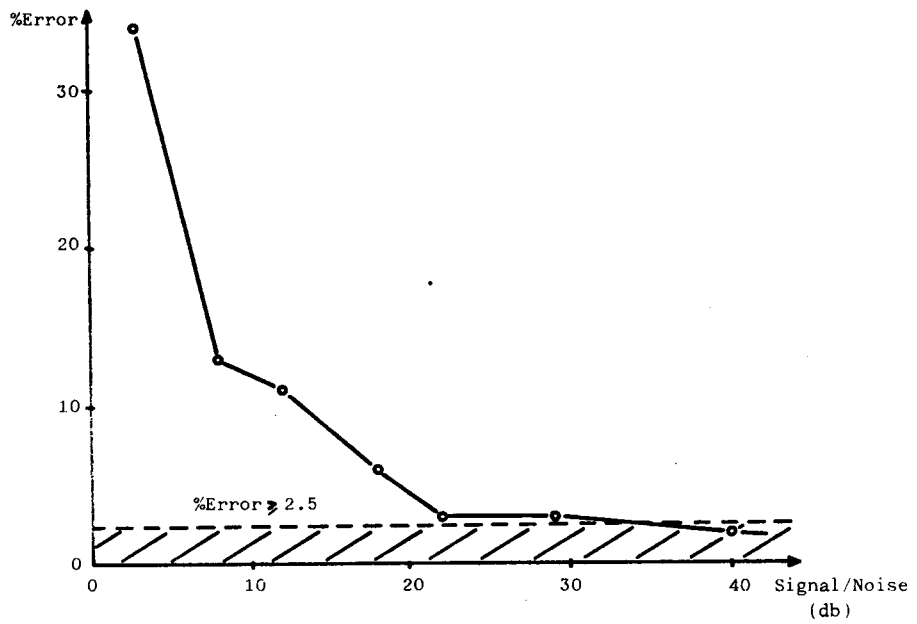

Fig. 10: System performance versus environnement conditions. Sampling frequency=8533 Hz. Window size=15 msec. Number of parameters=8. Using average magnitude=yes. Using preemphasis=yes.

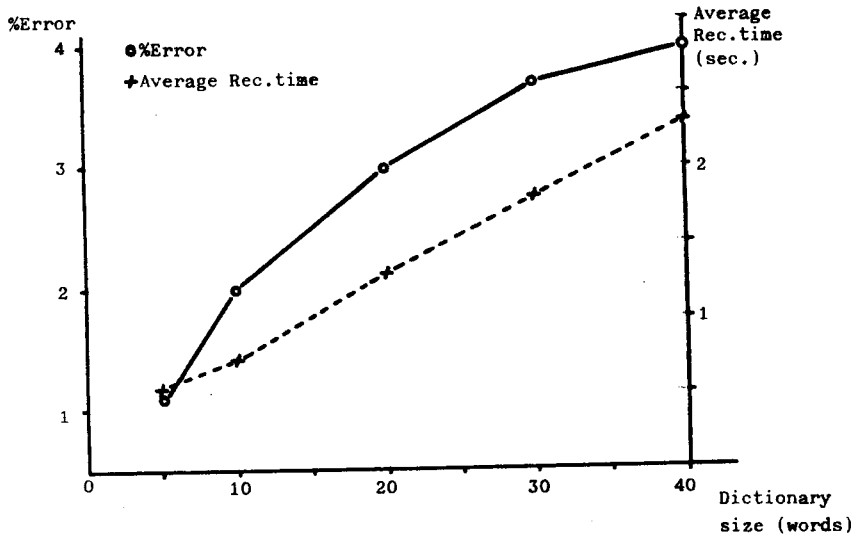

Fig. 11: System performance versus dictionary size.

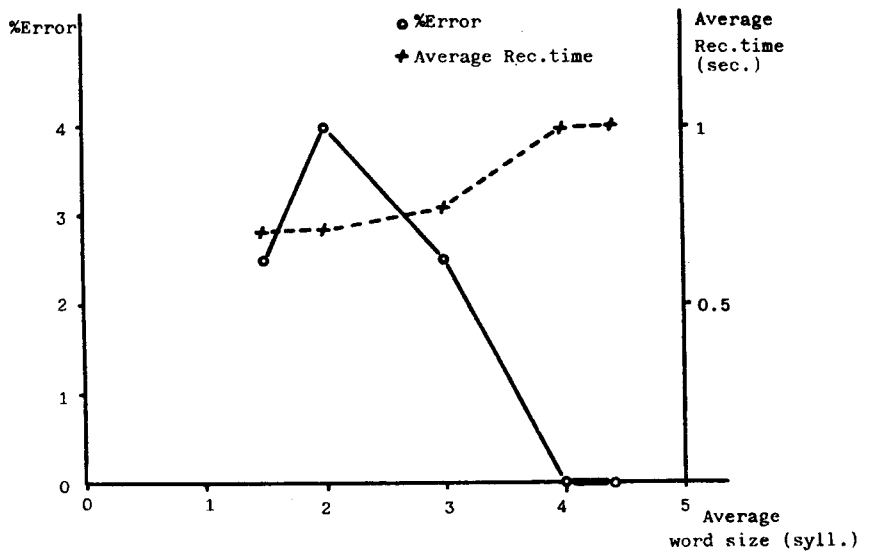

Fig. 12: System performance versus average word size.

Table 1
Digits dictionary

Word	Phonetic trans.	# of syl.
cero	θéro	2
uno	úno	2
dos	dós	1
tres	trés	1
cuatro	kwátro	2
cinco	θíŋko	2
seis	seɟs	1
siete	sjéte	2
ocho	óco	2
nueve	nwéβe	2

Table 2
Computer science dictionary

Word	phonetic trans.	# of syl.
bis	bís	1
fin	fín	1
par	pár	1
red	r̃éd	1
voz	bóθ	1
árbol	árβol	2
bucle	búcle	2
ciclo	θíklo	2
cola	kóla	2
dato	dáto	2
grafo	gráfo	2
lista	lísta	2
matriz	matríθ	2
nodo	nóðo	2
pila	píla	2
serie	sérje	2
tabla	táβla	2
tipo	típo	2
valor	balór	2
cadena	kaðéna	3
condición	koŋdiθjón	3
dígito	díxito	3
índice	íŋdiθe	3
lectura	lektúra	3
memoria	memórja	3
programa	proɣráma	3
puntero	puŋtéro	3
registro	r̃exístro	3
sucesor	suθesór	3
sintaxis	siŋtáɣsis	3
algoritmo	alɣorítmo	4
autómata	autómata	4
catálogo	kâtáloɣo	4
procesador	proθesaðór	4
semántica	seméŋtika	4
subrutina	subr̃utína	4
aleatorio	aleatórjo	5
clasificación	klasifikaθjón	5
organigrama	oryanivráma	5
procedimiento	proθeðimjéŋto	5

Table 3
Spanish names dictionary

Word	Phonetic Trans.	# of syl.
Ana	ána	2
Bruno	brúno	2
Celia	θélja	2
Daniel	danjél	2
Elisa	elísa	3
Francisco	fraɲθísko	3
Gabriel	gaβrjél	2
Isabel	isaβél	3
Juan	xwan	1
Luis	luís	1
María	maría	3
Nieves	nieβes	2
Olga	ólγa	2
Pedro	péðro	2
Quique	kíke	2
Rosa	r̄ósa	2
Sebastián	seβastján	3
Teresa	terésa	3
Úrsula	úrsula	3
Yolanda	ǰolánda	3

Table 4
System performance for
different speakers

speaker	Sex	Performance
BAC	F	95%
EMA	F	97%
LRC	M	95%
EMV	M	95%
JRP	M	97%
FCN	M	96%

rest (100 words) for recognition.

For the results of fig. 11 and 12, the speech data corresponds to the 40-words dictionary given in Table 2. This dictionary has been uttered 8 times by another male speaker with the acquisition sessions carefully scheduled in time throughout 4 weeks; the first three acquisitions was used for dictionary buildings, and the rest (200 words) for recognition.

For the results of Table 4, finally, it has been utilized the 20-words dictionary shown in table 3. For each speaker, all the words has been uttered 3 times for dictionary building, and the recognition results are given on the basis of at least 5 on-line repetitions of the dictionary (100 words by each speaker).

All the acquisition sessions was made in a rather quiet terminal-room (s/n = 40 db.) through a close-talking microphone and an analog preprocessor which adapt the signal to the following A/D conversion. None of the speakers involved in the experiments had previous experience with speech devices, and the only advertisement given to them (besides those automatically given by the signal quality control algorithm) was to speak as natural as possible.

In all the appropriate experiments, the results are plotted either in terms of error-rate (on the basis of a 0.7 MIPS Eclipse C-350 processor), and error-per-time product. The last plot is useful to ilustrate the "overall" system performance. A shaded zone has been drawn in each plot which delimitates the region in which both the error-rate and recognition-time have acceptable values for the corresponding experiment.

The dictionary-size dependent results, has been obtained by progressively partitioning ar random the largest dictionary (Table 2) into smaller subdictionaries, each of which maintaining constant the average number of syllables per word (2.75 syllables/word). The recognition results of all the subdictionaries of the same size are then averaged to give the error-rate figure for the considered size. A similar procedure has been followed-up in order to obtain the results dependent upon the average word size. In this case, all the subdictionaries have the same size (10 words) and have been defined by randomly selecting the appropriate words from the master dictionary, so as to get the desired average number of syllables per word in each partition.

8. CONCLUSIONS.

It has been presented in this paper an Isolated Word Recognition System based on the (real-time computed) Two-Level signal quantized Autocorrelation Function (TLAF) as a parameter-extraction procedure.

Following the results presented in last section, it can be concluded that the system gives acceptable error-rates and recognition-times (0%-5%; 0.5-1.5 sec) at sampling frequencies ranging 6400 to 12800 hz., and (related) non-overlapping window sizes of 15 to 25 ms., with a number of TLAF parameters in the range of 4 to 17. Using both signal-average-magnitude ($\tilde{R}(0)$) and 6 db/octave pre-emphasis has shown to be helpfull to improve the recognition performance, and specially bad results are obtained when none of them are used. The time-warping window size seems to give good results with values from 100 up to 300 ms., and actual values to be used can be "tunned" to the average length of the words to be recognized. The noise tolerated by the system can reach signal-to-noise figures of about 25 db. with still acceptable error-rates (3% for the digits dictionary), and further noise raising degrade the system in a softly progressive way. The recognition time grows less then linearly with the dictionary size (thanks to the inmediate refusing of word candidates with durations exceding that of the uttered word in more then the T.W. window size), and dictionaries up to 40 words are quite well handled by the system with error-rates smaller than 4%. The speaker-dependent results, finally, show performance variations among individuals, and among speaker classes (male-female) which are not different from those obtained with other classical parameter sets.

The system prototype runs on a 64 kbytes partition of a general pourpose minicomputer with quite small special hardware requirements, and

is being used up to date as a demostrative Isolated Word Recognizer. It is also capable to be configurated as an oral interface /11/ for applications running on the same computer, and, under this point of view, can be considered as one more utility of the computer software environnement.

9. REFERENCES.

/1/ H. RULOT, E.VIDAL, F.CASACUBERTA: "La función de autocorrelación en el reconocimiento de la palabra". V Congreso de Informática y Automática, Madrid, p.p.799-803,(May 82.)

/2/ H.RULOT, E.VIDAL, F.CASACUBERTA: "Isolated Word Recognition System based on the Autocorrelation Function". Portugal Workshop on Signal Processing and Applications; Povoa de Varzim, p.p.B1/2/1-8, (Sep. 82.)

/3/ L.R. RABINER, R.W. SCHAFER: "Digital Processing of speech signals". Prentice-Hall,(1978.)

/4/ A. PAPOULIS: "Probability, Random Variables, and stochastic processes". MacGraw-Hill,(1965)

/5/ J. MAX: "Methodes et techniques de traitement du signal et application aux mesures physiques". Masson et Cie., (1977).

/6/ J.H. VAN VLECK, D. MIDDLETON: "The spectrum of clipped noise". IEE Proceedings, Vol. 54, N.1, January 1966.

/7/ H. SAKOE AND S. CHIBA: "Dynamic programming algorithm optimization for spoken word recognition". IEEE Trans. Acoust. Speech. Signal Processing; Vol ASSP-26, p.p. 43-49, (Feb. 1978).

/8/ L. RABINER, S. LEVINSON: "Isolated and conected word recognition theory and selected applications". IEEE Trans. on Communications; Vol COM-29, N.5, p.p. 621-659, (May 1981).

/9/ G.L. BRADSHAW, R. COLE, ZANGGE LI: "A comparison of learning techniques in speech recognition". IEEE-ICASSP, Vol.1 p.p. 554-557, (May 1982).

/10/ J.L. FLANAGAN: "Speech analysis, synthesis and perception", Springer-Verlag, Berlin (1972).

/11/ H. RULOT, E. SANCHIS, F. CASACUBERTA, E. VIDAL: "An oral query for bibliographic data-base retrieval". 4Th. Marocco Workshop on Signal Processing and applications.Marraketch, (Sep. 1984).