A SPEECH TRAINING AID FOR THE DEAF

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Master of Technology

by

NIRANJAN D. KHAMBETE

Roll no. 90314014

Guide: Dr. P.C. Pandey

Co-guide: Dr. S.R. Devasahayam

School of Biomedical Engineering

Indian Institute of Technology, Bombay

July, 1992.

Indian Institute of Technology, Bombay

DISSERTATION APPROVAL SHEET

Dissertation entitled " A SPEECH TRAINING AID FOR THE DEAF," submitted by Niranjan D. Khambete, is approved for the award of the degree of Master of Technology in Biomedical Engineering.

Guide: .....

Co-guide: .....

Internal Examiner: .....

External Examiner: .....

Chairman: .....

Date: 7-8-92

Niranjan D. Khambete: A Speech Training Aid for the Deaf,
M. Tech. dissertation, School of Biomedical Engineering,
Indian Institute of Technology, Bombay, July 1992.

ABSTRACT

Prelingual, profoundly deaf children have great difficulty in achieving intelligible speech. The obvious reason is absence of auditory feedback of their own speech. Even after intensive instructions, their speech may remain deficient in prosodic and phonetic characteristics.

This project is aimed at developing a PC based speech training aid for the deaf. The speech training aid displays vocal tract shape and energy variations for the speech of the speaker (the deaf student under speech training, or the teacher). The system has been designed for operation in two modes: real-time display mode and slow motion review mode. An add-on DSP board using signal processor chip TMS320C25, with on-board memory shareable between the processor and PC, was found suitable for implementation of the aid. Area function is extracted using LPC algorithms and image is generated on the DSP board, and display image and data for storage are transferred to PC frame-by-frame. The system can be further improved by displaying a more realistic vocal tract shape and inclusion of pitch.

ACKNOWLEDGEMENT

I take this opportunity to thank my guides Dr. P.C.Pandey and Dr. S.R.Devasahayam for their valuable guidance and constant encouragement throughout this work. Help from Prof. M.C.Srisailam in providing the DSP board for use in this project is gratefully acknowledged. I also thank all my friends who helped me in completing this work. I am thankful to Prof. T.G.Thomas for patiently correcting my report. I specially mention the help and support extended by Mr. Anil Vartak and Mr. A.D. Apte from Standards Lab.

Niranjan D. Khambete

CONTENTS

CHAPTER 1

INTRODUCTION

Profoundly deaf children have difficulty in achieving intelligible speech, due to lack of feedback of their own speech through the auditory mechanism. The speech problems of these hearing impaired persons cover a wide range, as a result of different extents and types of loss and also the age of onset of hearing impairment. The most difficult problems are faced by those who have severe hearing loss from birth, or before the age at which speech is acquired.

Different techniques are employed to improve the intelligibility and quality of speech of the deaf persons by providing feedback of their speech through an alternate sense modality. These techniques range from the simple one like lipreading to advanced ones like display of vocal tract shape on a screen. The two main objectives of such a training scheme should be to indicate how to produce a particular speech segment and then to evaluate it by indicating the features lacking in it and how to correct for the same.

1.1. SCOPE OF THE PROJECT

Scope of this project is the design and implementation of a speech training aid based on display of vocal tract shape and energy variations in real time on a PC screen by using a suitable digital signal processor as a peripheral, and suitable analysis algorithms. Display of vocal tract shape is expected to help in the acquisition of correct place and manner of articulation, while the display of energy variations will help in improving the voicing and prosodic characteristics of speech.

This work is a part of an ongoing project at IIT Bombay for developing speech training aids for the deaf (Gupte, 1990; Takalikar, 1991; Gracias, 1991). Algorithms for analysis of speech signal have been earlier developed and tested for off-line processing, as well as in assembly language of digital signal processor TMS32010. Work in this project involves implementing the speech training aid in two different modes: real-time display mode and slow motion review mode, for the display of speech parameters on the screen. After studying the different requirements for implementation of the aid, an appropriate choice for the hardware set-up and software algorithms is to be made to satisfy the same. Finally, the system has to be tested for different sounds, and made suitable for clinical testing and further research work.

1.2. OUTLINE OF THE REPORT

Second chapter presents a brief review of problems faced by the deaf in achieving speech along with description of some conventional speech training aids. Third chapter discusses the requirements of the analysis and display setup for implementing the aid. Fourth chapter describes different programs developed in programming language C and in assembly language of signal processor TMS320C25 along with the results. Last chapter summarizes the work done in this project work and suggestions for further work. Appendix A describes algorithms used by different programs for the analysis of speech signal. Results indicating display for different speech segments are available in Apendix B while program listing are available in Appendix C.

CHAPTER 2

SPEECH TRAINING FOR THE DEAF

The normally hearing world has developec language based on speech, with writing as a derivative form developed after spoken language competency is acquired. For profoundly deaf children, poor development of the spoken language impedes access to the written language. This produces a disability which results typically in reading attainment of 5 to 7 years behind that of normally hearing children at the age of 16 (King et al, 1982). This language deficit limits the entire educational process of these children.

Natural methods used to understand and acquire speech by the deaf are lipreading (or visual speech reading), tactile speech reading, cued speech, finger spelling, and signing. Lipreading involves viewing the speakers face to recognize speech from the articulatory cues available. In tactile speech reading, speech is received by placing hand on the speaker's face for monitoring the movements associated with speech production. Hand signals are used to supplement the facial signals in a cued speech, while these hand signals are used to represent letters in finger spelling. Signing is another method to corvey meaning of certain words by appropriate gestures of hand and is found effective when used along with finger spelling. Following

sections in this chapter present a review of speech problems of the deaf and different speech training aids for the deaf.

2.1. SPEECH PROBLEMS OF THE DEAF

Errors or deficiencies in the production of speech by deaf children can be divided into two major categories: errors affecting intelligibility and errors affecting voice quality. Each of these groups can be further subdivided into two categories: errors of articulation and errors involving supra-segmental characteristics.

Improper timing and rhythm in speech affect its intelligibility. Deaf persons often tend to speak slowly, emission rate of syllable or word in speech of deaf is two to three times less than that of the hearing persons (Nickerson, 1975). They fail to make difference between stressed and unstressed syllables, both being prolonged in duration. Pauses are often inserted in inappropriate places such as in between the phrases. Both the problems of pauses and poor rhythm are related to poor breath control and expelling more air during speech production.

Pitch variations, used to indicate stressed and unstressed vowels, to add emphasis, and to carry information

about structure and meaning of the sentence, are absent in speech of deaf. Major difficulties are inappropriate average pitch, unusually high pitch, and improper intonation resulting in monotonous speech or speech with erratic variations in pitch. Pitch variations from vowel to vowel may be the consequence of improper use of muscles or muscle groups for controlling vowel articulation.

Most common articulatory errors are omissions, substitutions, and insertions of adventitious syllables while pronouncing consonants. Initial consonants are particularly prone to omissions. Another common error is confusion between voiced and unvoiced consonants (e.g., pat for bat) and between nasals and plosives (e.g., mat for bat). Use of neutral vowel /æ/ (as in about), as a general purpose vowel also leads to a loss in intelligibility. It has been also observed that labial phonemes, e.g. /p,b,m/ are more distinct than non-labial or lingual phonemes, e.g. /d,k,t/ (Nickerson, 1975). Here it may be noted that the exact place and manner of articulation for lingual phonemes is not visible and consequently the information not available through lipreading. Phonemes that involve executing smooth transitions in position of articulators are prone to errors than those produced by articulators in fixed position.

Quality of voice of a deaf person may get affected by several reasons, one of them being improper velum control resulting in hypernasality or hyponasality. It gives entire speech a characteristic sound. However, improper velum control may also affect intelligibility by causing confusion between sounds /m,n,ng/ and /b,d,g/. Learning of velum control is difficult for deaf because raising or lowering of velum cannot be detected by mere lipreading and activity of velum produces very little proprioceptive feedback.

Breathiness in speech of deaf can be considered as another main factor affecting the voice quality. This is attributed to a relatively large glottal opening produced by failure to close the vocal cords properly. Large variations in volume are also responsible to affect the quality of speech by deaf.

Thus it is clear that the deficiencies in the speech of the deaf, which require correction, are varied and complex. An ideally designed speech training aid should be able to correct all these defects. Thus the quantity of information to be conveyed to the subject is very large and it becomes important to decide upon an optimum method for presenting it.

2.2. SPEECH TRAINING AIDS

An attempt to impart speech training to the deaf must try to establish the feedback between the output speech and the speech production process. Pitch of our speech is controlled by adjusting the tension over vocal cords while the formants are controlled by varying the vocal tract shape by moving the tongue and the lower jaw. Proper co-ordination between these different organs can be established only if the feedback to brain is sufficiently rich in all required features of speech and these features can be easily interpreted to achieve precise control over speech producing organs.

A wide range of speech training aids is available, each one aiming at correcting a particular set of features in speech of a deaf person by modifying or encoding the speech signal so that it is suitably accepted by the chosen sense modality. The following part of this section describes some of these aids along with comments on possible reasons for their success or failure.

2.2.1. Hearing Aids and Cochlear Prosthesis

Hearing impairments may be classified into four major categories, conductive loss associated with pathology of middle ear, ear drum, or ear canal, sensory-neural loss due to defects

in cochlea or auditory nerve, central loss or lack of interpretation of speech, and psychological loss. Conventional hearing aids are useful in cases of conductive loss as they enhance the required parameters of speech signal to overcome the deficiencies in middle ear, ear drum, and ear canal. Cochlear prosthesis encode the information present in speech signal into electrical pulses for stimulating the auditory nerve endings in the inner ear through implanted electrode(s) and are therefore useful in cases of sensory-neural loss. If the hearing loss belongs to remaining two categories, one has to make use of an alternative sense modality to provide feedback of the speech. Vision and touch are the two alternative sense modalities that can be used to provide the feedback by converting the speech signal in suitable form.

2.2.2. Lipreading and Supporting Aids

Lipreading provides information about place and manner of articulation for various speech segments, although to a very limited extent. Details about tongue position are not visible in lipreading. In addition, it is not possible to discriminate between voiced and unvoiced sounds or nasal and non-nasal sounds, by mere lipreading. Information about the prosodic characteristics of speech like pitch variation, syllable stress, etc., is completely absent in lipreading. However, in spite of

all these limitations, lipreading has been a popular technique for speech training among the deaf because of the ease with which it can be learnt and used for speech reception during face to face communication.

To provide the information missing in lipreading, sense organs of touch and vision can be utilized as an additional input channel. Added information improves the communication in two ways: firstly it conveys features such as pitch and intensity variations that are not conveyed in lipreading, and secondly even if the additional information is sometimes redundant, it improves the speech reception against interfering distractions.

Upton (1968) designed a visual aid having miniature lamps mounted on the eye glasses to be worn by the deaf. Fig. 2.1 shows the arrangement of these miniature lamps on eyeglasses. An electronic analyzer is used to extract information about voicing, frication and stops from speech. The miniature lamps are caused to flash in synchronism with speech to form dynamic speech patterns representative of presence of these speech qualities. Studies conducted by Pickett (1974) to evaluate the effectiveness of this aid indicate improvement in lipreading for poor lipreaders but no significant improvement for good lipreaders. Intensive training is necessary to make deaf persons to recognize the different visible patterns. However, details of

Upton's analyzer circuit, and tests evaluating the extraction of various speech features used are not available.

Vibrotactile and electrotactile aids encode the features of speech into spatial patterns of mechanical or electrical stimulation along the skin. Discrimination of speech sounds through a tactile aid combined with lipreading is found to be better than that achieved by lipreading alone. A study done by Stark et al (1976) indicates that prolonged training was required for children to recognize different words and no student was able to produce intelligible speech. The reason for this failure may be mainly due to the slow response of the skin to a stimulus and adaptation to a repeated stimulus thus limiting its frequency response to a level which is just not sufficient to receive all the information present in speech signal.

Limited success of Upton's eye glasses and the tactile aids as speech training aids could be due to three factors. Firstly, assessments for effectiveness of these devices in actually extracting the speech features are not available. Secondly, the deaf person undergoing speech training by using these aids may not be able to effectively receive information presented to them for correcting their speech. Thirdly, the information in the form of abstract visual or tactile patterns is not physically related to the actual efforts required in

producing a particular speech segment correctly.

2.2.3. Visual Speech Training Aids

Speech training aids described in this section have a common feature that they use a screen for presenting the information extracted from the speech signal in suitable form. It is possible to generate varied patterns on the screen. However, it may be better to concentrate on one speech feature at a time and the form of display should be such that the information presented through it can be received and made use of by the deaf person under training. (Levitt, 1980). In the context of explaining the difficulty in reading spectrograms, Liberman et al (1968) have suggested that a single acoustic cue may carry information simultaneously about several successive phonetic segments. This parallel delivery of information makes it possible to evade the limitations imposed by the temporal resolving power of the ear and to understand speech as fast as we do. But, this type of parallel delivery of information will not help to improve speech training by using visual displays because eye lacks the type of decoder the ear has and may respond in different manner to a similar stimulus.

Attempts to make use of display of variations in fundamental frequency of voicing or pitch as a speech training aid were made as early as 1938 (Levitt et al, 1980). Better aids were developed with improvement in the techniques used to find the pitch value more accurately. Displays for these aids evolved from a simple graphical display indicating variation of pitch with respect to time to multicoloured displays to encode range of pitch values on a colour scale. Evaluation of these aids indicate moderate success to establish correct control over pitch. Though the techniques used for extracting pitch give reliable results, the forms of display chosen are not effective (Levitt et al, 1980).

A system developed by Bernstein et al (1986) uses a PC screen for display and signal processor TMS32010 for extracting pitch information from the speech signal in real time. The technique of electroglottography (EGG) is used for monitoring the opening and closing of the vocal cords, and the pneumotachograph (PTG) for monitoring the volume velocity of expiration. These two signals in conjunction with the pitch information serve as a diagnostic tool for detecting abnormalities in voicing. As a speech training aid, system presents certain games, each one intended to correct a particular characteristics of voicing. The different characteristics chosen for this purpose are, sustained vocalization, repeated short vocalization, and intensity changes

with respect to time. Screen display of some of these games are shown in Fig. 2.2. Success of a speech training aid of this type may be doubtful because it really does not convey any information about the actual effort required to produce speech of intended quality. These games may only be helpful to stimulate the deaf student to undergo necessary training for long hours. No results are available indicating the achievement of the aid in improving the intended speech qualities of the deaf student.

Spectrographic display of speech signals have also been experimented with as visual speech training aids. Information available in these displays, include the formant frequencies and their transitions and the fundamental frequency of voicing. Though the formant frequencies are directly related to the vocal tract shape, the information presented is really hard to interpret for achieving correct articulation. Liberman et al (1968) have explained the reason for difficulty in reading spectrograms:

> "Spectrogram presents speech in its uncoded form. The audible signal has been made visible, but is not decoded. As a consequence, it is not highly readable. When speech comes through the ear, it finds a readily available processor that decodes it and removes the strings of phonemes. There is no such decoder available for the eye.

As a consequence, visual representations of speech are hard to read, no matter how good transform or how long the training. . . . . Since much of the encoding, perhaps most of it, occurs at the conversion of muscle contraction to the vocal tract shape, information about articulatory muscle contraction might hold some promise as a useful way of presenting speech signal to deaf."

In consonance with this view, several researchers have been developing visual aids that present information related to vocal tract shape and its variations either in a direct form as a display of cross-section of vocal tract or in some indirect form of game-like displays. Recent advances in digital electronics and signal processing techniques has made it possible to extract vocal tract shape from a speech signal and display it on the screen. A brief review of these type of aids follows with comments on their success and probable reasons of success.

2.2.4. Vocal Tract Shape Display Aids

Bristow et al (source untraceable) have reported a vocal tract shape display aid, using a microprocessor fast enough for real-time processing and a domestic television set for display. Display provides plot of smoothed log area of vocal tract versus

linear distance along the vocal tract from glottis at the left to lips at the right. A similar display , as shown in Fig. 2.3, is obtained by Crichton and Fallside (1974) for their speech training aid by using linear predictive coding for extraction of vocal tract shape. A reference trace for the intended sound is displayed by a dotted line and the display freezes when a definite match is achieved between the reference and the actual shape. In another mode of the display incorporating time consideration, area function is displayed as a 2-D display with time along the x-axis and distance from glottis along the y-axis. The amplitude of the area is coded as intensity of the display. This display, termed as areagram, is somewhat similar to a spectrogram in form. Results of detailed evaluation of the signal processing algorithms are not available.

In a speech training system designed by Pardo (1982), area function was extracted using linear prediction of speech. Normalizing of area function was done assuming a constant volume normalization as the criterion. This indicates an attempt to give some physical significance to the available area function. A special interpolation algorithm is designed to speed up the performance of the aid. The display of the aid, shows the graph of normalized area function on an inverted y-axis against distance on x-axis, as shown in Fig. 2.4. Articulatory positions are not indicated in the display.

In a modified form of this aid (Pardo et al, 1986), a realistic vocal tract shape is displayed on the screen and is manipulated according to the area function values available after processing the speech signal frame-by-frame. A typical display is illustrated in Fig. 2.5. Signal processor micro P7720 from NEC is used to perform speech analysis with an additional processor for sampling of the analog speech signal, the PC being used only for the display. The system is designed to work in two modes. In the first mode, an image moves on the screen continuously with the input speech, in the second mode, facility is provided to view the shape for an isolated pronunciation for the speaker to match his own shape with it. No results are reported to decide upon the mode in which system has maximum success. They have also reported another aid for indicating variations in intensity, pitch, and voicing on the screen for a spoken sound. This aid aims at training the deaf speakers in improving the prosodic characteristics of speech.

A speech training aid developed by Shigenaga and Kubo (1986) also intends to train the deaf in articulation of vowels and some consonants. For an uttered vowel, the system indicates, how close it is to the intended vowel and then to each of the other vowels, by making use of animation. Then it shows the place and manner of articulation required to produce the intended vowel correctly by displaying the reference vocal tract shape and that

of the actually uttered sound superimposed on each other. To extract vocal tract shape for stop consonants in consonant-vowel context, appropriate pre-emphasis coefficient is selected automatically for the frames of the stop consonant and the following vowel. Since it is not possible to estimate the vocal tract shape while the vocal tract is closed for stop consonants, it is checked as to whether the place of minimum area of estimated vocal tract area function is near to the place of articulation for the intended stop consonant. Fig. 2.6 shows the display of this aid. The reference shape indicates the place of articulation necessary to produce intended consonant. The display has more physical understandability in terms of identifying different articulators such as tongue, lips, etc. and therefore should prove effective for training of articulation. However, it is reported in the paper that the system is not extensively used by deaf students and thus its effectiveness is yet to be proven.

An entirely different approach to impart speech training is the development of an aid known as "palatograph" by Shibata and Hiki (1975), and Fletcher (1978) (reviewed by Levitt et al, 1980). Position of the tongue in contact with upper palate is monitored by means of tiny sensors mounted on an artificial palate placed in the deaf student's mouth. The artificial palate is custom made and is very thin and light so as not to interfere with the normal speech production. No further details are

available about the type of sensors and the related hardware used in the aid. The real advantage of this aid is that it indicates the exact point of contact between tongue and the upper palate for stop consonants which cannot be derived by signal processing techniques used to extract the vocal tract shape from speech signal.

One may conclude that a speech training aid should instruct the deaf student to produce a particular sound, analyze the sound with sufficient accuracy, and present a comparison between the features of the intended sound and the actually produced sound with instructions indicating the ways to improve it. Display of the vocal tract shape can be considered suitable for training articulation of vowels and some stop consonants. A proper training strategy has to be adopted to make the aid really successful in its purpose.

2.2.5. Display System of Gupte and Takalikar:

In earlier efforts at IIT Bombay, a hardware set up and programs were developed for implementing a speech training aid (Gupte, 1990; Takalikar, 1991; and, Gracias, 1991). Fig. 2.7 shows the block diagram of the system. The speech processor is built around DSP-TMS32010 Evaluation Module (EVM) from Texas Instruments. The extension card has four different circuits to

carry out different functions. Interface to EVM module provides data and address buffers and I/O mapping signals. Analog-to-digital converter (ADC) chip is a 12 bit successive approximation type ADC with conversion time of 25 μs. Rate multiplier circuit provides the sampling pulses for the ADC. Extended data memory is mapped into data memory space of TMS32010. Organization of 12, 6116 memory chips provides 4k * 16 of total memory. Addressing of this memory is achieved by a programmable counter. A host interface circuit is required to interface EVM with PC. Information transfer between PC and EVM is through 8 bit command and acknowledgement registers with a status register for handshaking. The interface card fits into the expansion slot of the PC and provides buffers for PC address and data bus. EVM communicates with the PC through a serial link for downloading of assembled programs. An antialiasing filter having 40 dB attenuation in stop band with a transition region of 4.6 kHz to 4.9 kHz is used to lowpass filter the speech signal.

Speech signal from a microphone is amplified and filtered by an antialiasing filter and then digitized at a rate of 10 k samples/second. Acquisition and processing of data is handled by TMS32010, in segments (typically of 30 ms). After a certain segment of the signal has been processed, the extracted parameters are then transferred to the PC for display purpose.

The assembly language program is first transmitted to the EVM from the PC using a serial link established by KERMIT software and executed. A menu driven program, running on the PC, asks for parameters like number of samples per frame, sampling frequency, frame overlap, etc. These parameters are transmitted to the EVM. The assembly language program starts acquiring data from ADC and analyses it using algorithms for linear predictive coding (described in the following chapter) and estimates the variations in cross-sectional area of vocal tract, energy, and pitch. These parameters are then transferred to the PC on interrupt basis before the following frame of speech signal is analyzed. The program simultaneously running on PC displays the variations in vocal tract shape, energy, and pitch for each frame.

Though required speech analysis programs are satisfactorily working in real time, overall real-time performance has not been achieved as programs running on the PC are written in higher level language (Gupte, 1990). The delay in process of data transfer between the PC and the signal processor board, is another major cause of non-real-time performance (Takalikar, 1991). The hardware set up, spread over three different boards, is prone to failures and increases the overall physical size of the system. Therefore, it is necessary to have a faster display programs and a compact hardware set up.

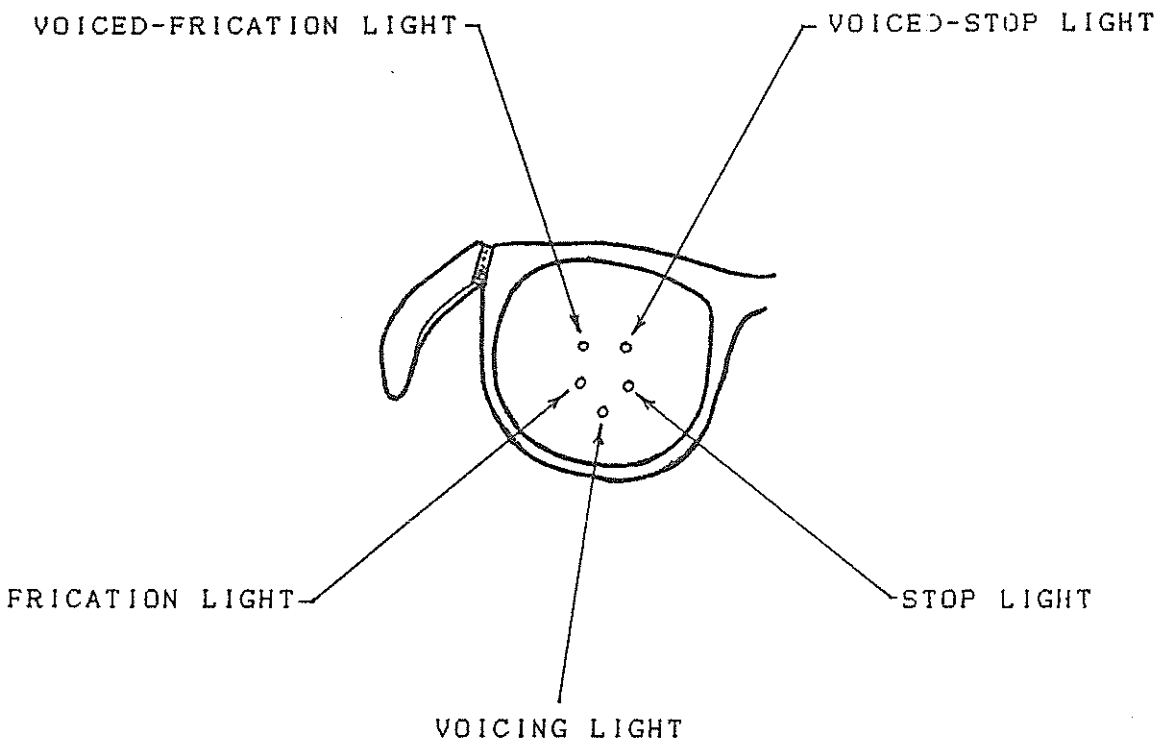
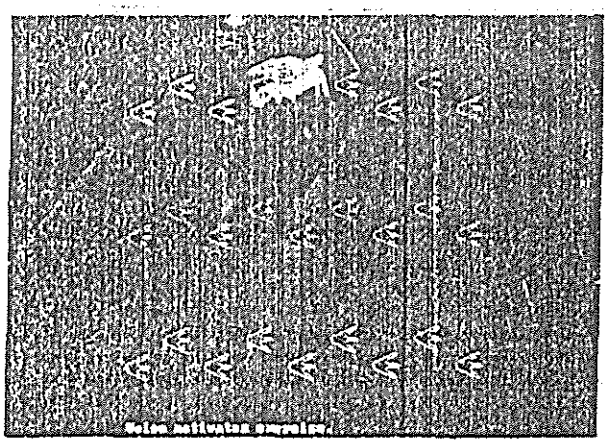
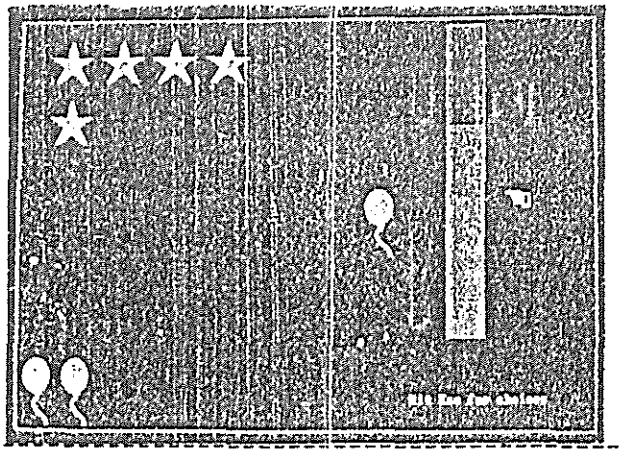


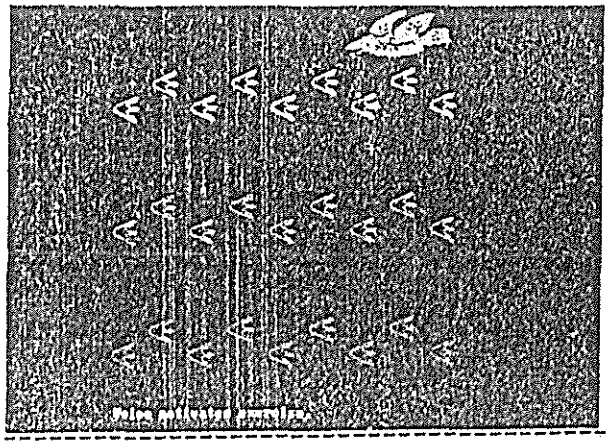
Fig. 2.1. Display arrangement of Upton's Eyeglass Speech Reader (Source: Upton, 1968)



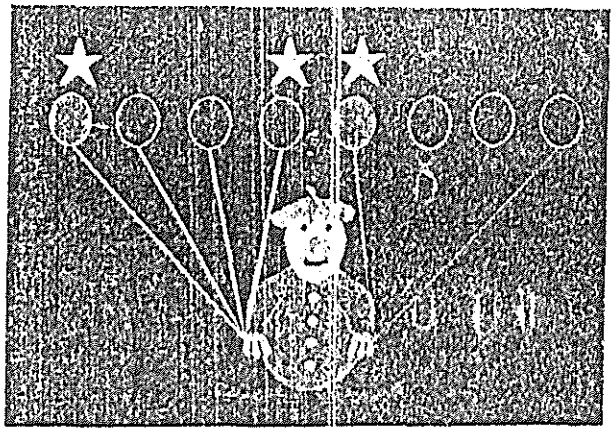
Graphic from repeated short vocalization game. Bird moves one foot print for every syllable vocalized.



Graphic from intensity game with feedback. Child must make balloon rise to the level indicated by the hand on the right of the screen.



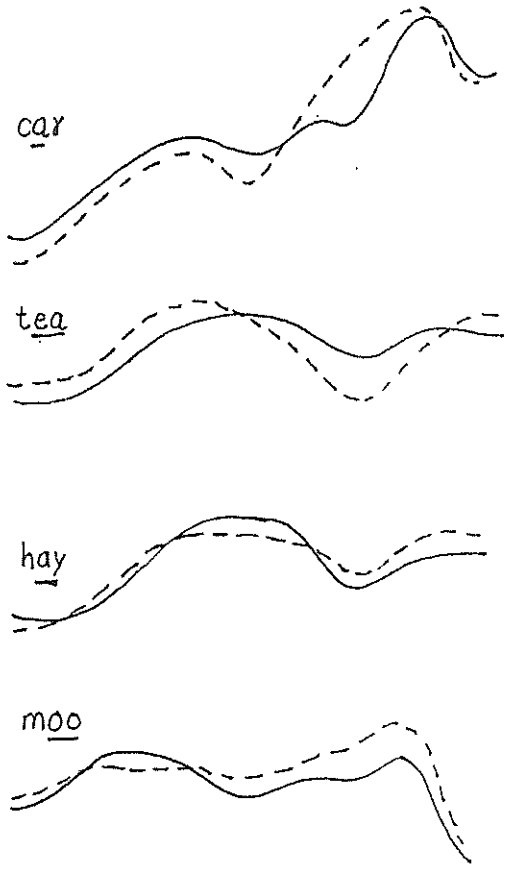
Graphic from repeated short vocalization game. Following successful sequence, bird flies across screen.



Graphic from intensity game with limited feedback. Child must produce intensity levels coded by the color of the balloons.

Fig. 2.2. Display of speech training games.

(Source: Bernstein et al, 1986)



..... Reference trace.

——— Trace for an attempt by deaf student.

Fig. 2.3. Display of smoothed log area function.
(Source: Crichton and Fallside, 1974)

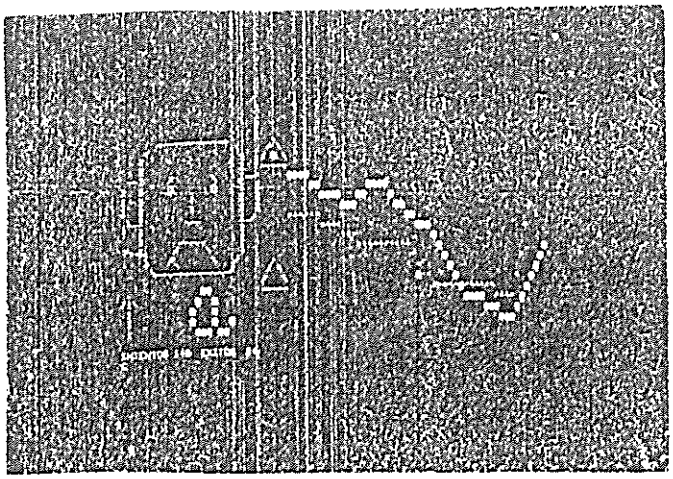


Fig. 2.4. Display for a speech training aid designed by Pardo (1982).

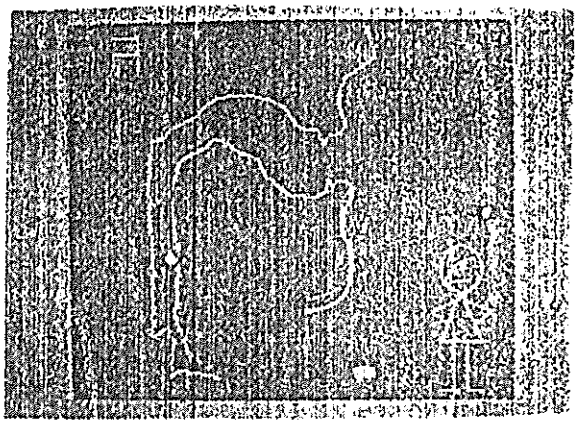
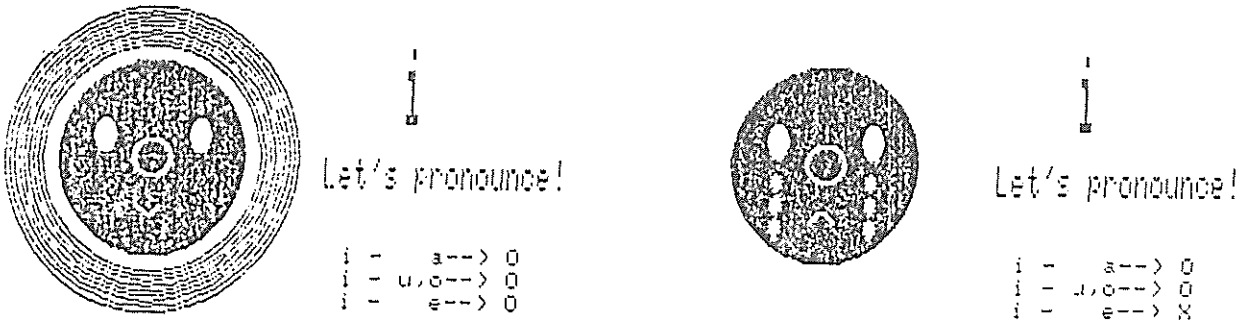


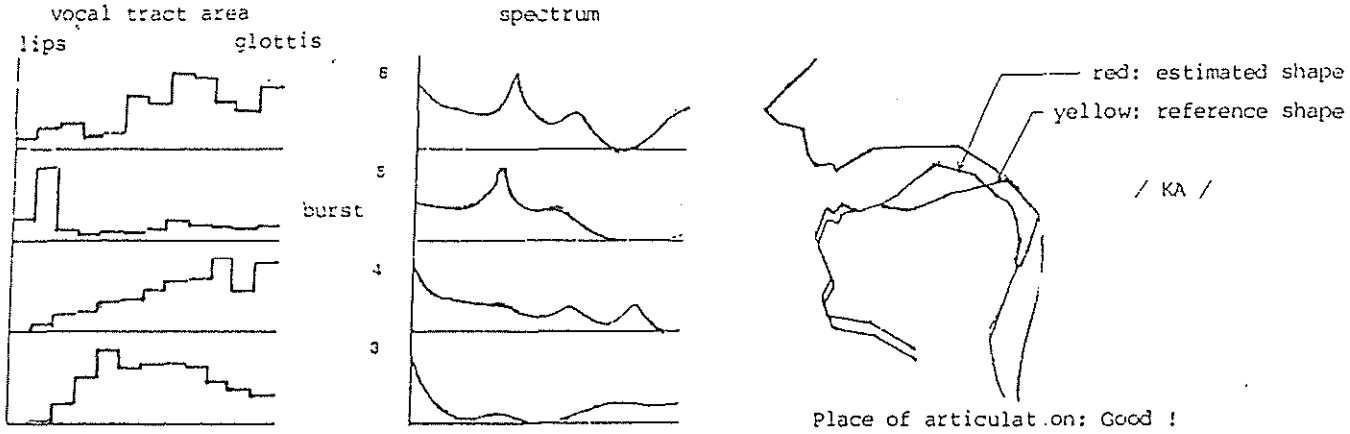
Fig. 2.5. Display for a speech training aid designed by Pardo et al (1986).



(b) Good pronunciation for /i/.

(c) When /i/ was confused with /e/.

Some pictures displayed during practice.



Estimated vocal tract areas and spectra in the vicinity of the onset of burst of /ka/.

Estimated and reference shapes at the onset of burst of /ka/.

Fig. 2.6. Display for a speech training aid designed by Shigenaga and Kubo (1986).

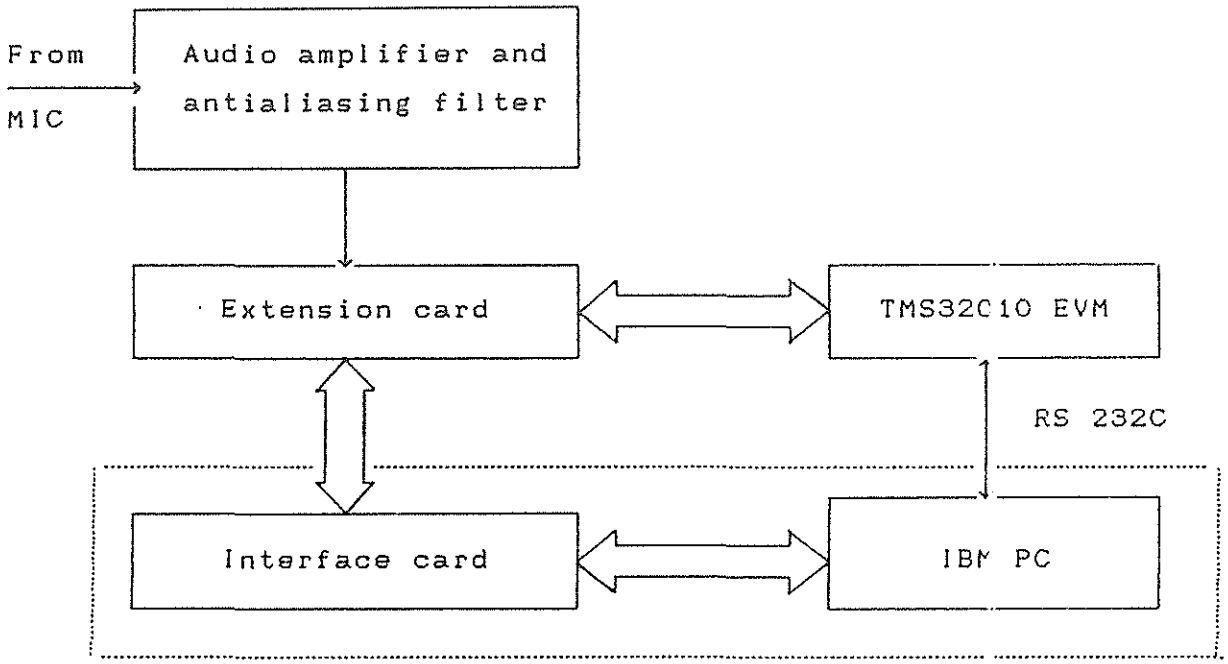


Fig. 2.7 Block diagram of system implemented by Gupte.

(Source: Gupte, 1990; Takalikar, 1991)

CHAPTER 3

IMPLEMENTATION OF THE AID

The development of a vocal tract shape based speech training aid involves selection of appropriate algorithms for analysis of the speech signal and a hardware setup for implementation of the chosen algorithms. The theory underlying these algorithms assumes a mathematical model for the human speech production system and different algorithms are developed to obtain solution to these mathematical equations. Hardware setup chosen for implementation of the aid should achieve the real-time display of vocal tract shape. The following sections describe the mathematical basis of different algorithms meant for estimation of vocal tract shape and the procedures carried out for evaluating the hardware setup for real-time operation of the aid. The final section describes the setup used for the implementation of the aid.

3.1. CHOICE OF ALGORITHMS

Estimation of vocal tract shape can be achieved by processing the digitized speech signal. A model for speech production system, assumed for developing these algorithms, is described in this section, followed by the the algorithm for estimation of the vocal tract shape.

3.1.1. Model for Speech Production System

A general block diagram for the model of speech production system is shown in Fig. 3.1. The excitation generator provides the signal equivalent to the excitation from the glottis and can be modeled by either an impulse train for voiced speech segment or a white noise for unvoiced speech segments (Rabiner and Schafer, 1978). The vocal tract from glottis to lips can be modeled as concatenation of lossless acoustic tubes, as shown in Fig. 3.2. The area function of the vocal tract is defined as the variation in its cross-sectional area with respect to its length from glottis to lips. If a large number of tubes of short length are used, one can expect the resonant frequencies of the concatenated tubes to be close to those of a tube with continuously varying area function. However, this approximation neglects the losses due to friction, heat conduction, and wall vibration, and therefore one may expect the bandwidths of the resonant frequencies to differ from the actual.

Let us consider the k^{th} and $k+1^{st}$ tube sections of cross-sectional area, A_k, A_{k+1} and length l_k, l_{k+1} as shown in Fig. 3.3(a). The pressure $p(x,t)$ and volume velocity $u(x,t)$ in the k^{th} tube are given as

$$p_k(x,t) = \frac{\rho c}{A_k}\left[\, u_k^+(t-x/c) + u_k^-(t+x/c) \,\right] \qquad (3.1a)$$

$$u_k(x,t) = u_k^+(t-x/c) - u_k^-(t+x/c) \qquad (3.1b)$$

where

c = velocity of sound wave propagating in air.

ρ = density of air.

x = distance measured from the left-hand end of the tube.

$(0 \leq x \leq l_k)$

$u_k^+()$ = volume velocity of forward travelling wave.

$u_k^-()$ = volume velocity of backward travelling wave.

The relationship between the travelling waves in lossless adjacent tubes can be obtained by applying physical principle that pressure and volume velocity must be continuous in both time and space everywhere in the system. Thus, applying this continuity condition at the junction between k^{th} and $(k+1)^{st}$ tubes gives

$$p_k(l_k,t) = p_{k+1}(0,t) \qquad (3.2a)$$

$$u_k(l_k,t) = u_{k+1}(0,t) \qquad (3.2b)$$

Substituting Eqs. 3.1 into Eqs. 3.2 gives

$$\frac{A_{k+1}}{A_k}\left[u_k^+(t-\tau_k) + u_k^-(t+\tau_k)\right] = u_{k+1}^+(t) + u_{k+1}^-(t) \qquad (3.3a)$$

$$u_k^+(t-\tau_k) - u_k^-(t+\tau_k) = u_{k+1}^+(t) - u_{k+1}^-(t) \qquad (3.3b)$$

where $\tau_k = l_k/c$ is the time for a wave to travel the length of the k^{th} tube. Solving Eqs. 3.3 gives

$$u_{k+1}^+(t) = (1+r_k)\, u_k^+(t-\tau_k) + r_k u_{k+1}^-(t) \qquad (3.4a)$$

$$u_k^-(t-\tau_k) = -r_k u_k^+(t-\tau_k) + (1-r_k)\, u_{k+1}^-(t) \qquad (3.4b)$$

where r_k is referred to as the reflection coefficient and is the amount of $u_{k+1}^-(t)$ that is reflected at the junction.

$$r_k = \frac{A_{k+1} - A_k}{A_{k+1} + A_k} \qquad (3.5)$$

Eqn. 3.4 can be written as follows in a matrix form in Z domain. If we sample this system at a rate of $F_s = c/2*l_k$, then a delay of τ corresponds to a factor of $z^{-1/2}$. A signal flow graph indicating this relationship is shown in Fig. 3.3(b).

$$\mathbf{U}_k = \mathbf{T}_k\, \mathbf{U}_{k+1} \qquad (3.6a)$$

$$\begin{bmatrix} U_k^+(z) \\ U_k^-(z) \end{bmatrix} = \frac{z^{+1/2}}{1 + r_k} \begin{bmatrix} 1 & -r_k \\ -r_k z^{-1} & z^{-1} \end{bmatrix} \begin{bmatrix} U_{k+1}^+(z) \\ U_{k+1}^-(z) \end{bmatrix} \qquad (3.6b)$$

where $\mathbf{T}_k$ is the inverse transfer function matrix for two tubes.

This model for vocal tract is useful for estimation of area function from discretized speech signal. Following section describes the technique.

3.1.2. Estimation of Vocal Tract Shape

To obtain the display of vocal tract shape for a particular speech segment, it is necessary to know the area function of the vocal tract. Though the area function for a sustained vowel remains constant, in a consonant-vowel (CV) and vowel-consonant (VC) context, variations are observed in the frames of vowel immediately following and preceding the consonant respectively. In case of stop consonants, area function cannot be derived at the instant when there is a complete constriction in the vocal tract at a point. However, the position of minimum area in the early frames of the vowel can indicate place of articulation for the stop consonant.

Linear predictive coding (LPC) may be used for estimating the area function from a speech signal (Wakita, 1973). The transfer function T(z) of the vocal tract modeled as concatenated tubes is equivalent to the filter transfer function (Atal and Hanuer, 1971).

$$T(z) = \frac{G}{1 + \sum_{k=1}^{p} a_k z^{-k}} \qquad (3.7)$$

where, G = Gain factor.

A filter with a transfer function having poles only has been considered, because if we ignore the case of nasals and some fricatives, transfer function of the vocal tract is represented reasonably by an all-pole function. The excitation source in this case is at the glottis (outside the oral cavity) and the nasal cavity is decoupled from the oral cavity by raising the velum. Therefore, there is no anti-resonant cavity that can introduce zeros in the transfer function.

From Eqn. 3.7 and referring to Fig. 3.4(a), we can write

$$S(z) = T(z)\ [G\ U(z)] \qquad (3.8a)$$

or

$$S(z)\ [1 + \sum_{k=1}^{p} a_k\ z^{-k}\ ] = G\ U(z) \qquad (3.8b)$$

or

$$s(n) = -\sum_{k=1}^{p} a_k\ s(n-k) + G\ u(n) \qquad (3.8c)$$

A linear prediction model of the speech production process (Atal and Hanuer, 1971) is shown in Fig. 3.4(b), in which predictor output signal is a linear combination of past p samples of s(n).

$$\hat{s}(n) = -\sum_{k=1}^{p} \alpha_k \, s(n-k) \qquad (3.9)$$

where, α_k = predictor coefficients.

p = predictor order.

The error between the actual value s(n) and the predicted value $\hat{s}(n)$ is given as

$$e(n) = s(n) - \hat{s}(n)$$
$$= s(n) + \sum_{k=1}^{p} \alpha_k s(n-k) \qquad (3.10)$$

If $\alpha_k = a_k$, then e(n) = G u(n). If signal is generated in accordance with Eqn.3.8 with non-time-varying coefficients and excited either by a single impulse or by a stationary white noise input, then it can be shown that the predictor coefficients that result from minimizing the mean squared prediction error (over all time) are identical to coefficients (a_k) of Eqn. 3.8. Further use of least mean square (LMS) error as a basis for estimating the filter coefficients (a_k) leads to a set of linear equations that can be efficiently solved to obtain the predictor coefficients. If we assume signal s(n) as a sample of random process, error e(n) is also a sample of a random process. The expected value of the square of the error is given as

$$E = E\,[e^2(n)]$$
$$= E\left[\, s(n) + \sum_{k=1}^{p} \alpha_k s(n-k)\right]^2 \qquad (3.11)$$

Taking partial derivative of E with respect to α_i and equating it to zero we can calculate the predictor coefficients.

$$\frac{\partial E}{\partial \alpha_i} = 0 \qquad 1 \le i \le p \qquad (3.12)$$

Hence we get the following equation.

$$\sum_{k=1}^{p} \alpha_k E\,[s(n-k), s(n-i)] = -E\,[s(n), s(n-k)]$$

$$\ldots\ 1 \le i \le p \qquad (3.13)$$

The minimum average error is given by

$$E_p = E\,[s^2(n) + \sum_{k=1}^{p} \alpha_k E\,[\, s(n), s(n-k)] \qquad (3.14)$$

Treating the speech signal as a short time stationary signal

$$R(i-k) = E\,[s(n-k), s(n-i)] \qquad (3.15)$$

where, R(i) is the autocorrelation of the stationary process and can be approximated as time average instead of ensemble average for window length N.

$$R(i) = \sum_{n=0}^{N-1} s(n)\, s(n-i) \qquad 1 \le i \le p \qquad (3.16)$$

Hence Eqn. 3.13 can be written as

$$-R(i) = \sum_{k=1}^{p} \alpha_k R(i-k) \qquad 1 \le i \le p \qquad (3.17)$$

The predictor coefficients α_k from $1 \le i \le p$ can be computed by solving a set of p equations in p unknowns.
Eq. 3.17 can be written in a matrix form as follows.

$$\begin{bmatrix} R_0 & R_1 & R_2 & R_3 & \cdots & R_{p-1} \\ R_1 & R_0 & R_1 & R_2 & \cdots & R_{p-2} \\ R_2 & R_1 & R_0 & R_1 & \cdots & R_{p-3} \\ \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ R_{p-1} & R_{p-2} & R_{p-3} & R_{p-4} & \cdots & R_0 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_p \end{bmatrix} = - \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ \vdots \\ R_p \end{bmatrix} \qquad (3.18)$$

It can be observed that all elements along the diagonal of the autocorrelation matrix are equal. Such a matrix is called Toeplitz matrix. In addition, this matrix is a symmetric matrix. These two properties allow a recursive solution to these p

equations in p unknowns. Recursive solution was first developed by Levinson and Durbin (Parsons, 1987). The recursion proceeds in steps. In each step, we have a solution $\alpha_{n-1}(i)$ for a predictor order of (n-1), and we use this solution to compute the coefficients $\alpha_n(i)$ for n^{th} order predictor. The three equations in the recursion are

$$\alpha_n(i) = \alpha_{n-1}(i) + k_n \alpha_{n-1}(n-i) \qquad (3.19)$$

where, $\alpha_n(n) = 0$ for any n.

$$k_n = \frac{-1}{E_{n-1}} \sum_{i=0}^{n-1} \alpha_{n-1}(i) R_{n-i} \qquad (3.20)$$

$$E_n = E_{n-1} (1-k_n^2) \qquad (3.21)$$

In Eq. 3.19 , the term $\alpha_{n-1}(n-i)$ actually forms an inverse vector of predictor coefficients. A new set of coefficients $\{\beta(i)\}$ can be defined as

$$\beta_p(i) = \alpha_p(p+1-i), \qquad i = 1,2,\ldots,p+1 \qquad (3.22)$$

The β vector forms a "postdictor" which will estimate the value of the preceding sample for a set of p data samples s(n-1) through s(n-p) as shown Fig. 3.5. The forward prediction error $e_p^+(n)$ and the reverse prediction error $e_p^-(n)$ can be defined as

$$e_p^+(n) = \sum_{i=0}^{p} \alpha_p(i)\, s(n-i) \qquad (3.23)$$

where, $e^+(n) = s(n)$ for $p = 0$.

$$e_p^-(n) = \sum_{i=1}^{p+1} \beta_p(i)\, s(n-i)$$

$$= \sum_{i=1}^{p+1} \alpha_p(p+1-i)\, s(n-i) \qquad (3.24)$$

where, $e_p^-(n) = s(n-1)$ for $p = 0$;

The forward and reverse prediction errors of consecutive filter orders can be represented in z domain by following equations.

$$E_p^+(z) = E_{p-1}^+(z) + k_p\, E_{p-1}^-(z) \qquad (3.25a)$$

$$E_p^-(z) = z^{-1} \left[E_{p-1}^-(z) + k_p\, E_{p-1}^+(z) \right] \qquad (3.25b)$$

Eqn. 3.25 can be written in a matrix as follows.

$$\begin{bmatrix} E_p^+(z) \\ E_p^-(z) \end{bmatrix} = \begin{bmatrix} 1 & k_p \\ k_p z^{-1} & z^{-1} \end{bmatrix} \begin{bmatrix} E_{p-1}^+(z) \\ E_{p-1}^-(z) \end{bmatrix} \qquad (3.26)$$

The Eqn. 3.26 is identical to inverse transfer function of vocal tract lossless tube model given by Eqn. 3.6 except for the term $(z^{1/2}/\ 1 + k_p)$. The coefficients k play the same role in Eqn. 3.26 that the reflection coefficients do in Eqn. 3.6 but with negative sign. Thus we can write

$$k_k = \frac{A_k - A_{k+1}}{A_k + A_{k+1}} \qquad (3.27)$$

where, A_k and A_{k+1} are the areas of successive cylindrical tubes. Thus area function of the vocal tract can be computed from coefficients k using following equation

$$A_k = \frac{1 + k_k}{1 - k_k} A_{k+1} \qquad (3.28)$$

where A_k is the area at glottis and is assumed constant.

Appropriate boundary conditions are assumed at lips for proper estimation of the area function. At lips, the vocal tract is assumed open having infinite area so that forward propagating wave is completely reflected. At glottis, the vocal tract is assumed to be terminated into a characteristic impedance and the backward propagating wave flows out to trachea without any reflection causing loss. Linear predictive analysis gives most reasonable results for these boundary conditions (Furui, 1989). In addition, proper estimation of area function requires removal

of effects of source characteristics at glottis and radiation characteristics at lips. The frequency characteristics of the sound source and radiation can be roughly approximated as -12 dB/octave and +6 dB/octave, respectively. Based on this approximation, the sound source and radiation characteristics can be canceled by +6 dB/octave pre-emphasis (Wakita, 1973).

The algotrithm for computing reflection coefficients directly from the autocorrelation coefficients and eliminating the computation of predictor coefficients was developed by Leroux and Gueguen (1977). If the autocorrelation coefficients are normalised by dividing each of it by the zeroth autocorrelation coefficient, then all the intermediate results and final results are guaranteed to lie within +1 and -1. Therefore this algorithm is most suitable for implementation in fixed point arithmetic and is chosen for implementing this speech training aid. Details of the algorithm are available in Appendix A.

3.2. CHOICE OF HARDWARE SETUP

For implementation of the speech training aid in intended modes of real-time display and slow motion review, it may be useful to consider the requirements of the setup as three separate blocks: analysis setup, data transfer setup, and display setup. For the aid to display speech parameters in real time, it

is necessary that all the three blocks work in real time and there is no accumulation of data. If the system works satisfactorily in real-time mode, it can be adapted to work in the other two modes.

3.2.1. Requirements for Real-Time Implementation

Let us consider the input speech signal is to be processed on frame-by-frame basis with a frame length of 30 ms. Acquisition of data for the next frame of 30 ms is done simultaneously when the previous frame is analyzed. These analysis parameters are to be transferred to the PC for display while the newly acquired frame is being analyzed. Further the analyzed parameters are to be processed and displayed on the screen by the PC in the following time frame of 30 ms. Thus, for a particular frame, the display of vocal tract shape for it will be displayed after a delay equal to thrice its time duration, i.e., 90 ms. This value is well within acceptable limits of the delay that is not disruptive for the perception of speech while viewing the speaker's face (Pandey, 1986).

Performance of each block is evaluated for its real-time functioning to make an appropriate choice of the setup for that block. Different procedures carried out to evaluate the performance of each block and the results obtained, and selection

of the setup for final implementation of the aid made on the basis of these results will be described in the following sections.

The signal processor TMS32010 (from Texas irstruments) was used earlier by Gupte (1990) and was reported to complete the signal analysis for estimation of vocal tract area function in real time. Therefore, assuming that the signal processor TMS320C25, the second generation processor from Texas Instruments would be satisfying the speed requirements for real-time signal analysis, we concentrate on to evaluation of the data transfer setup and display setup for achieving real-time performance of the speech training aid.

3.2.2. Evaluation of Data Transfer Setup

Speech parameters derived by the signal processor board are to be transferred to the PC for display in appropriate form. Several modes of data transfer are possible out of which three modes are evaluated for their speed, to select a suitable one for real-time implementation of the aid.

Let us consider that the predictor order used to estimate the vocal tract area function is 12. Therefore, for each frame of 30 ms, we have to transfer 12 values of area function (8 bits each), and energy and pitch values (16 bits each). Thus the

bit rate for data transfer is 128 bits per 30 ms or 4267 bits/second. Therefore, a serial link between DSP board and PC operating at 9600 baud rate should be adequate for data transfer in real time. A kit manufactured by Vinytics Peripherals Ltd. for development of assembly language programs for TMS32010 was considered as the DSP board. It has two serial ports. One port is used for downloading of the assembled programs for the signal processor from the PC to the kit, and the other port can be programmed to transmit the analysis parameters to the PC. After assessment of the programming involved in establishing this serial link, it was found that overhead associated with the serial communication does not permit the data transfer in real-time as expected. Moreover, non-functional ADC circuit on the kit made it difficult to actually evaluate the performance of this mode of data transfer for real-time application.

Gupte (1990) in his setup involving a TMS32010 Evaluation Module (from Texas Instruments) had designed and built a bidirectional port for transferring speech parameters to PC. Handshake between PC and the processor is achieved by interrupts of the processor and a status register. Availability of new data word is continuously checked by PC, by polling a particular bit in the status register. Once the new data word is available and is read by the PC, interrupt is generated to the processor in request of the next data word. Thus parameters corresponding to

each frame are transferred to the PC before processing of the next frame is started. Transfer of data in this mode is not sufficiently fast for the system to function in real time (Gupte, 1990).

The third approach for data transfer is to use a memory that can be shared by the PC and the processor. In this mode, PC should be able to read and write this shared memory at any time independent of the status of the processor i.e., running or on-hold. A PC add-on card PCL-DSP25 manufactured by Dynalog Microsystems Ltd. (DMS), provides this facility of shared memory. External program and data memory, each of size 64k * 16 is mapped into I/O space of the PC. Address to this memory is generated by programmable address counter. The address counter can be programmed appropriately in auto-increment or auto-decrement mode. Once the starting address of the memory block to be transferred is loaded, the counter increments or decrements automatically after each access done by the PC. This access by the PC introduces just a single wait state in the processor's machine cycle causing insignificant reduction in its speed of execution.

A scheme is implemented on this board to verify the performance of this mode of transfer. A program running on the processor continuously acquires an analog signal and stores it in

a circular buffer of 300 words. A program running on PC simultaneously reads the same memory locations and displays the waveform on the screen. There is no noticeable delay between actual analog waveform and that displayed on the PC screen. A change in amplitude of the analog waveform can be instantaneously observed on the PC screen.

3.2.3. Evaluation of Display Setup

Speech parameters analyzed by the signal processor are transferred to the PC for display on the screen. For the system to work in real time, the display should be updated within the time duration equal to that of one frame of speech signal, i.e., 30 ms. Different schemes for updating the image on the PC screen are tried out and evaluated to measure the time required for a single image update cycle. If we display outline of the head on the PC screen, only a part of it, namely the vocal tract, will be actually modified in each update cycle. Therefore, we consider a window of appropriate size, positioned so that the image inside this window will have to be updated for each frame. Size of the window is decided by two factors, one is the acceptable size of vocal tract image on the screen to properly distinguish between its different shapes for different sounds, and the time required to update the image of chosen window size. As the size of the window increases, more and more memory locations will have to be

modified in order to update the image inside this window. Fig. 3.6. shows the vocal tract shape along with the window of size 192 * 32 pixels. The variations in vocal tract shape can be displayed by using this window. Time required for updating the image of this size is calculated by a program described later in this chapter.

First step in evaluation of the display is carried out by writing a program that measures the time required for one image update cycle for a given image size in pixels using two different graphic drivers CGA (Colour Graphics Adapter) and EGA (Enhanced Graphics Adapter) in both high resolution and low resolution mode. Results obtained by this program are noted in Table 3.1. It is observed from the results that for the same image size specified in pixels, memory required for CGA is less than that required for an EGA (both high and low resolution modes), and therefore time for one image update cycle for the image of same size is more on EGA than on CGA. Moreover, display memory configuration for the CGA is simpler than that of EGA. In high resolution mode (640 * 200 pixels), each bit in the memory is directly mapped to each single pixel on the screen, and each block of 80 bytes forms a row. This is illustrated in Fig. 3.7. A single image update cycle consists of two calls for the "putimage()" function (in programming language C), one for erasing the earlier image and the next for displaying new image.

The results indicate that time required for one image update cycle is not within the limit of 30 ms as required for real-time implementation of the aid. Moreover, the results do not include the time required by the PC for generating the image from the received speech parameters.

An alternative to modifying the entire image as one block of memory is to write pixels at the required co-ordinates on the screen. A program is written to measure the time required to plot a specified number of pixels on the screen using "putpixel()" function in C. Results of this program are listed in Table 3.2. In order to update an image, first it is necessary to erase the earlier image by writing the pixels by background color and then writing all the pixels by foreground colour from the newly acquired data. Therefore, the time required for one image update cycle is twice that required for writing 192 pixels.

After measuring the timings for one image update cycle on PC screen in two different schemes, we had to rule out the option of using PC for image updating task for real-time implementation of the aid.

Third scheme of display was tried out by generating the bit image by the signal processor in its external memory and then transferring this block of memory directly to the display memory

of the PC. A program written in TMS320C25 assembly language sets a bit in a particular memory location, address of which is determined by the x and y co-ordinates of the pixel to be displayed on the screen within the window. Following expression is used to find the address of the particular memory location,

base address + (x/2) * (words/row) + integer (y/16)

Then a bit at the location equal to y modulo 16 counted from MSB is set to logic 1 in this particular memory location. There are separate memory blocks for odd and even values of x co-ordinates to take care of interlaced scanning of the display. Both these memory blocks are transferred to the display memory of PC by a program simultaneously executing on PC. This program also stores 12 values of area function and 2 values corresponding to energy (zeroth autocorrelation coefficient) and a dummy value (that can be pitch value) for each frame in an array as well as displays the energy on the screen by a marker on a vertical scale. Time required for plotting 200 points after clearing the entire window by the processor is just 1.6 ms. while that required for the transfer of image and values is 20.38 ms.

3.3. SELECTION OF FINAL SETUP

After evaluating the data transfer and display setup one arrives at the conclusion that the add-on card DSP boar PCL-DSP25 from DMS is suitable for real-time implementation o the speech training aid. This DSP board with TMS320C25 has man more additional features than the TMS32010 EVM board from Texas Instruments used earlier by Gupte (1990).

A schematic of the DSP board PCL-DSP25 is shown in Fig. 3.8. On-board analog-to-digital converter (ADC) has a 35 μs conversion time. The 16 bit programmable timer is programmed to provide start conversion pulses to ADC at required sampling interval. The end of conversion (EOC) pulse from ADC interrupts the processor for reading the digitized sample. A debug monitor is available as one of the development tools for this DSP board. A program for calculating area function and generating the image in the external memory of the signal processor was written as described in detail in the following chapter and was observed to take 5.2 ms for a frame length of 30 ms.

In addition to this add-on card, a microphone amplifier and an antialiasing filter is used in the setup as shown in Fig. 3.9. The preamplifier is a FET-input battery driven amplifier. The unity gain buffer and a non-inverting amplifier are built

using op-amp μA741 to amplify the speech signal to the input range of ADC on the DSP board, i.e., +10V to -10V. The antialiasing filter is a seventh order lowpass elliptic filter with a 40 dB attenuation in the stop band having transition region of 4.6 kHz to 4.9 kHz (Pandey, 1987). The DSP board fits in one of the expansion slots on the PC mother board. The speech signal acquired by the microphone is amplified, filtered, digitized, and analyzed by TMS320C25 to display the vocal tract area function and energy on the PC screen in real-time.

Table 3.1. Time required for one "putimage()' function (in programing language C) for blocksize of 192 * 32 pixels.

Dsiplay driver/ mode	EGA 640 * 200	EGA 750 * 350	CGA 640 * 200
Memory size (bytes)	3306	3306	631
Time (ms)	75.49	80.66	17.80

Table 3.2. Time required for executing "putpixel()" function (in programming language C) 192 times.

Dsiplay driver/ mode	EGA 640 * 200	EGA 750 * 350	CGA 640 * 200
Time (ms)	54.94	54.94	30.21

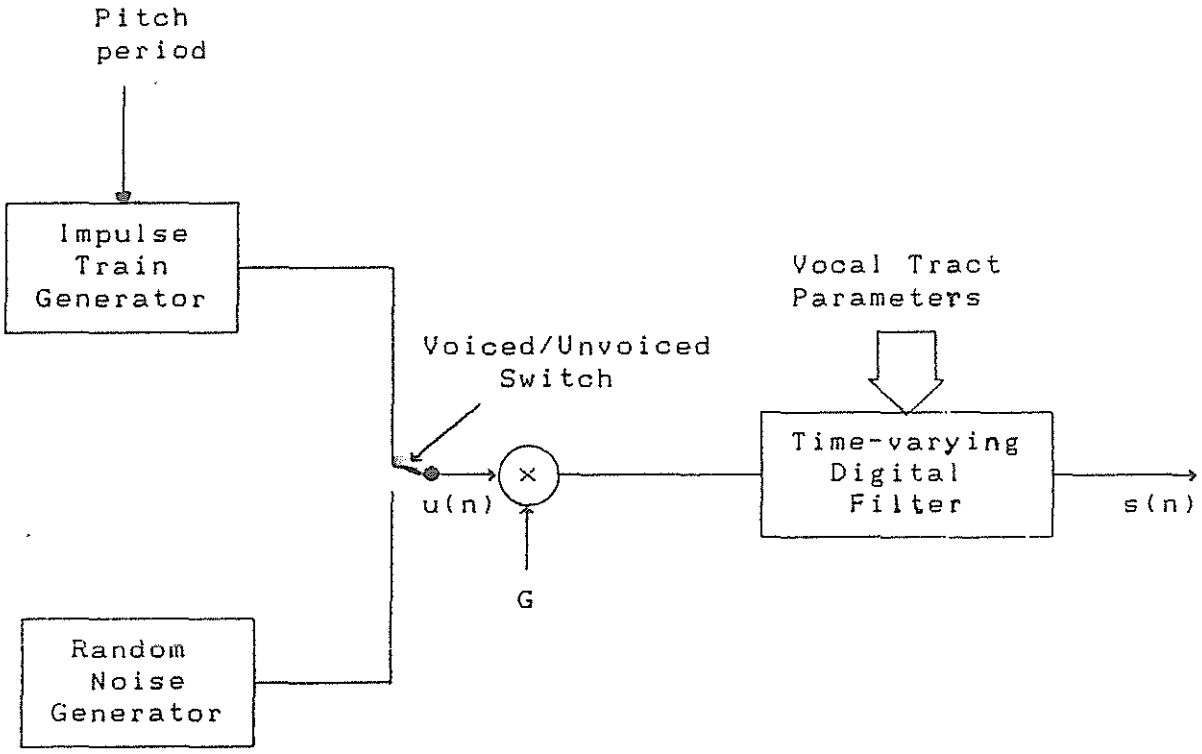


Fig. 3.1 Block diagram for simplified model of speech production. (Source: Rabiner and Schafer, 1978)

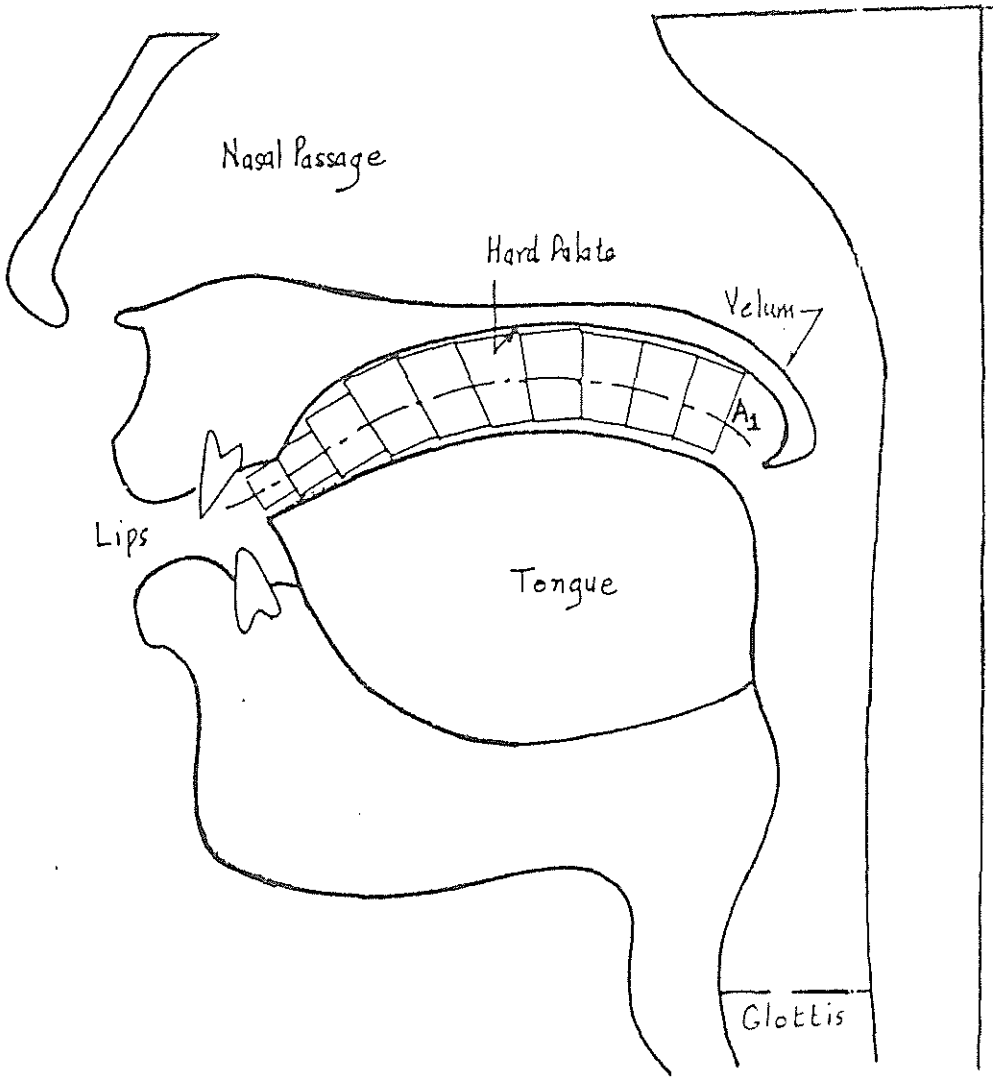


Fig. 3.2. Acoustic tube model for vocal tract.

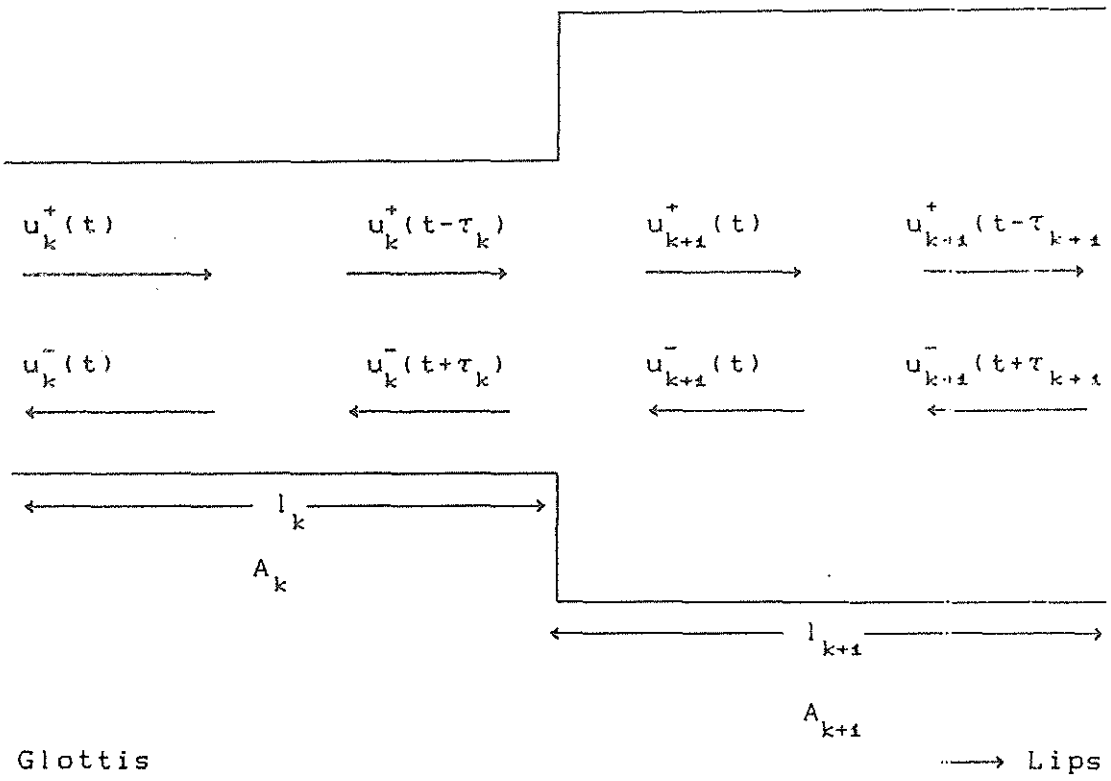


Fig. 3.3.(a) Junction between two lossless tubes

(Source: Rabiner and Schafer, 1978)

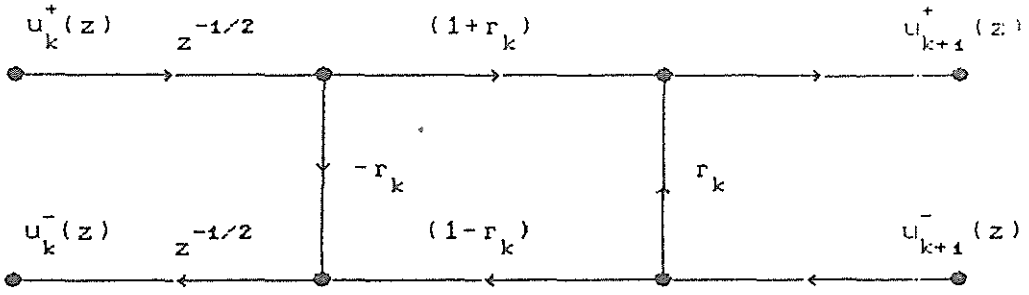


Fig. 3.3.(b) Flow graph representing relationship among z-transform at a junction.

(Source: Rabiner and Schafer, 1978)

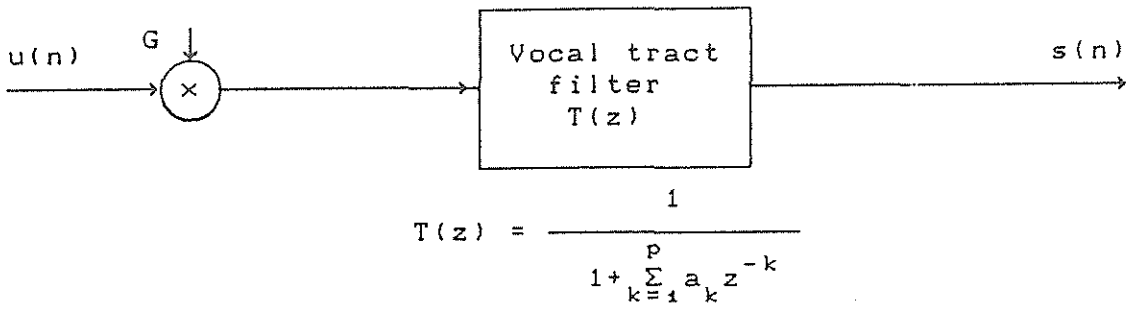


Fig. 3.4.(a) Block representation of vocal tract filter.
(Atal and Hanauer, 1971)

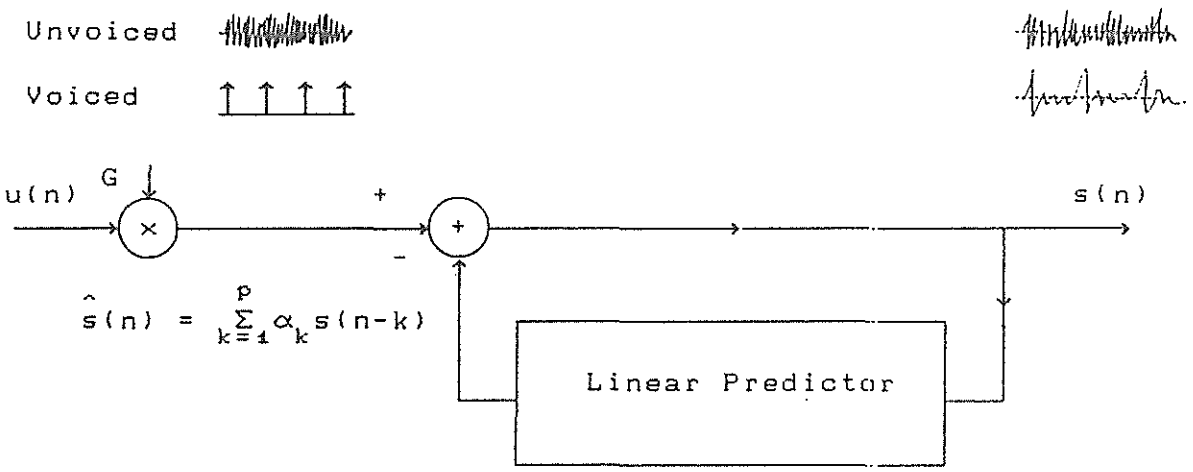


Fig. 3.4.(b) All-pole model for vocal tract.
(Atal and Hanauer, 1971)

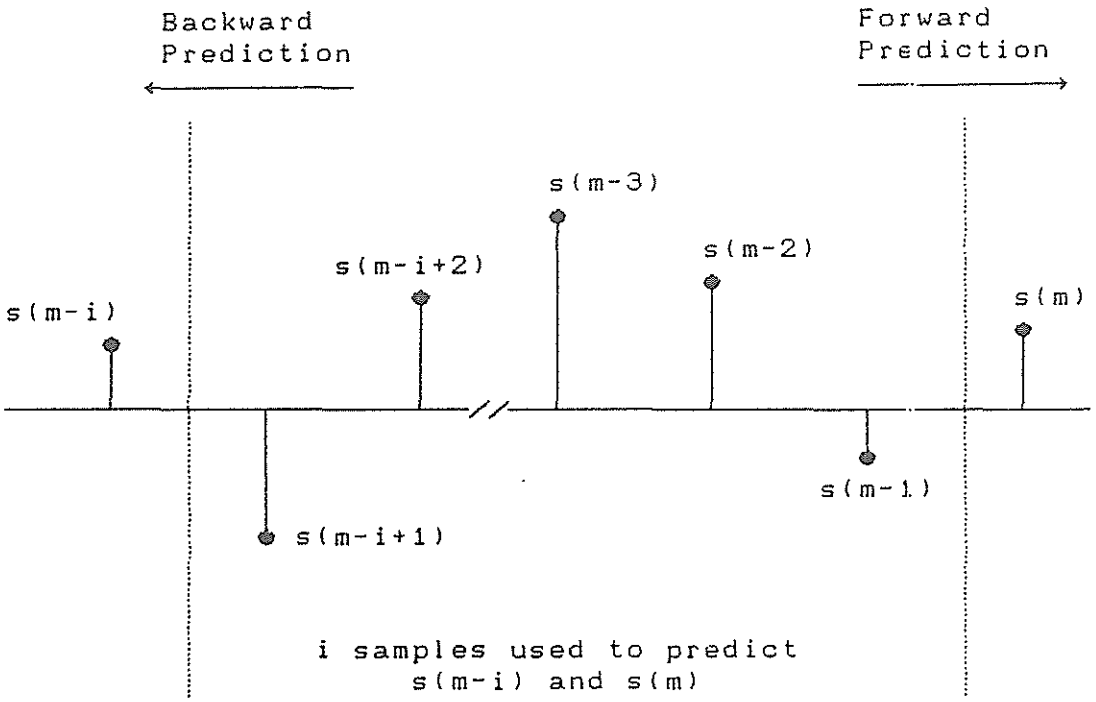


Fig. 3.5 Forward and backward predictor

(Source: Rabiner and Schafer, 1978)

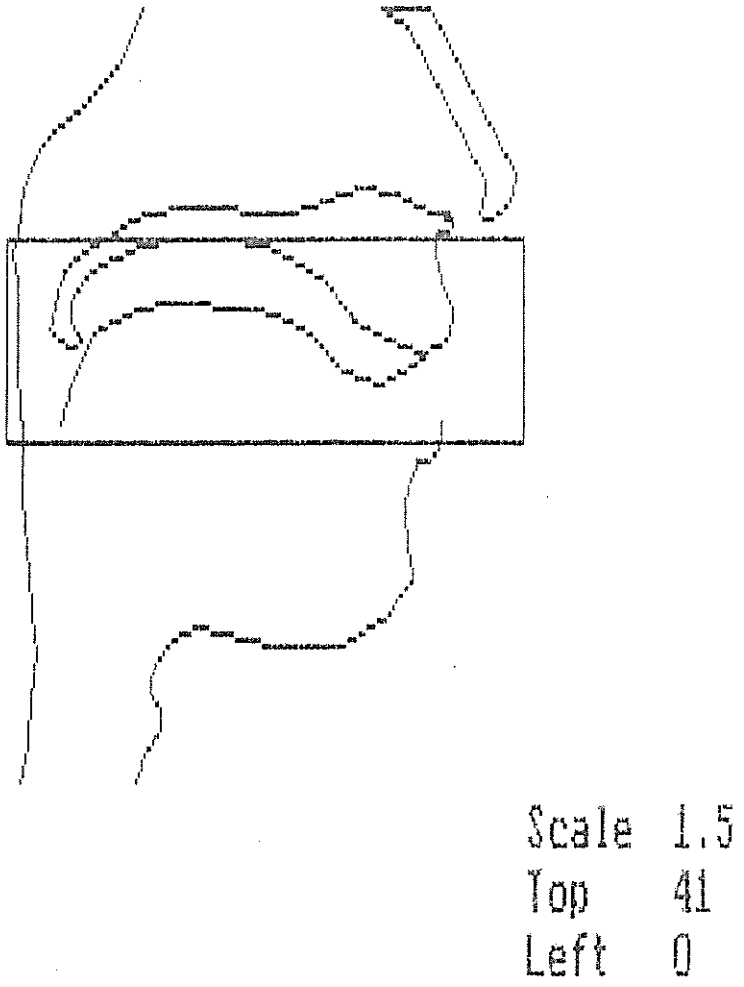


Fig.3.6. Selection of appropriate window size for real-time image updating.

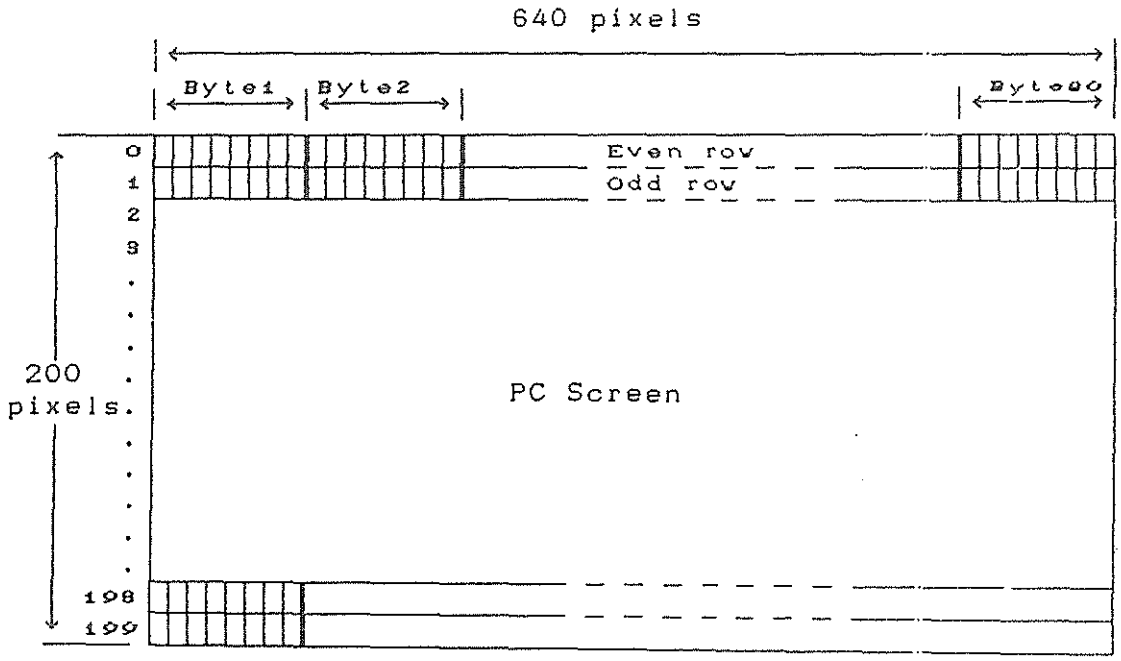


Fig. 3.7. Configuration of display memory in CGA high resolution B/W mode (640 * 200 pixels)
(Source: IBM PC/XT Technical Reference Manual)

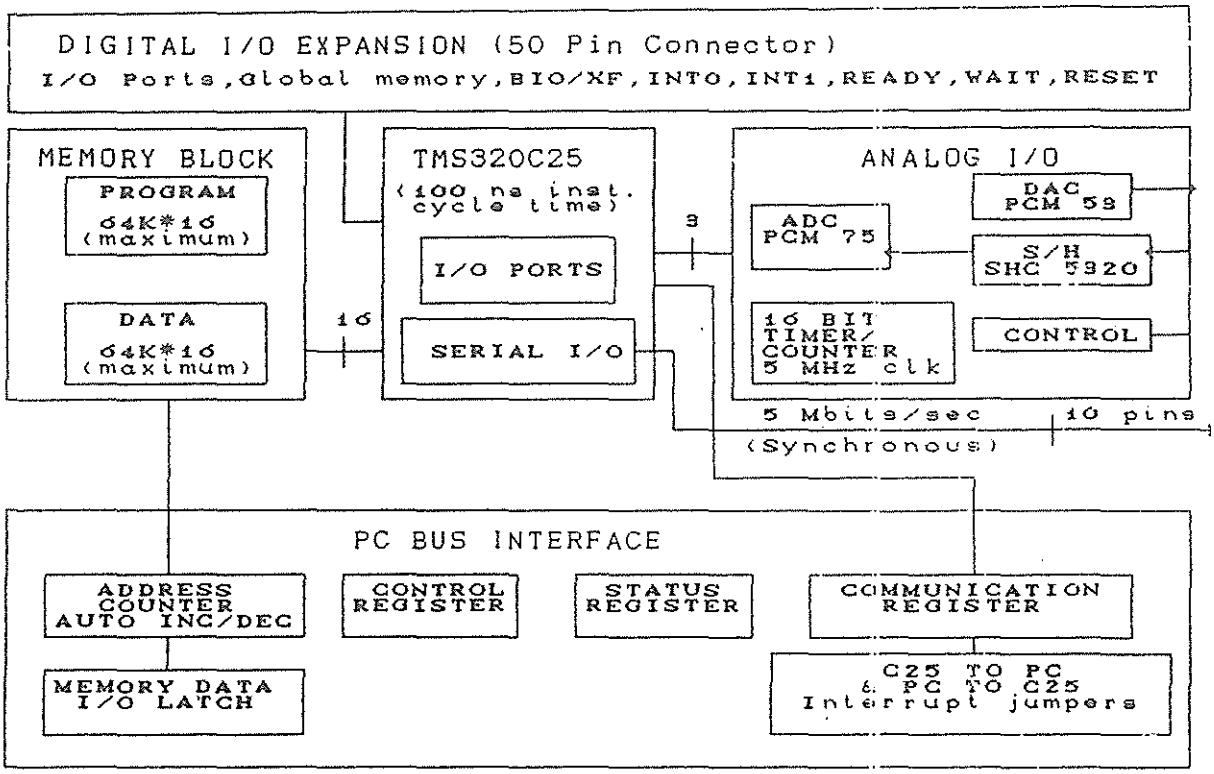


Fig. 3.8. Block diagram of TMS320C25 add-on card PCL-DSP25

(Dynalog Microsystems Ltd., Bombay)

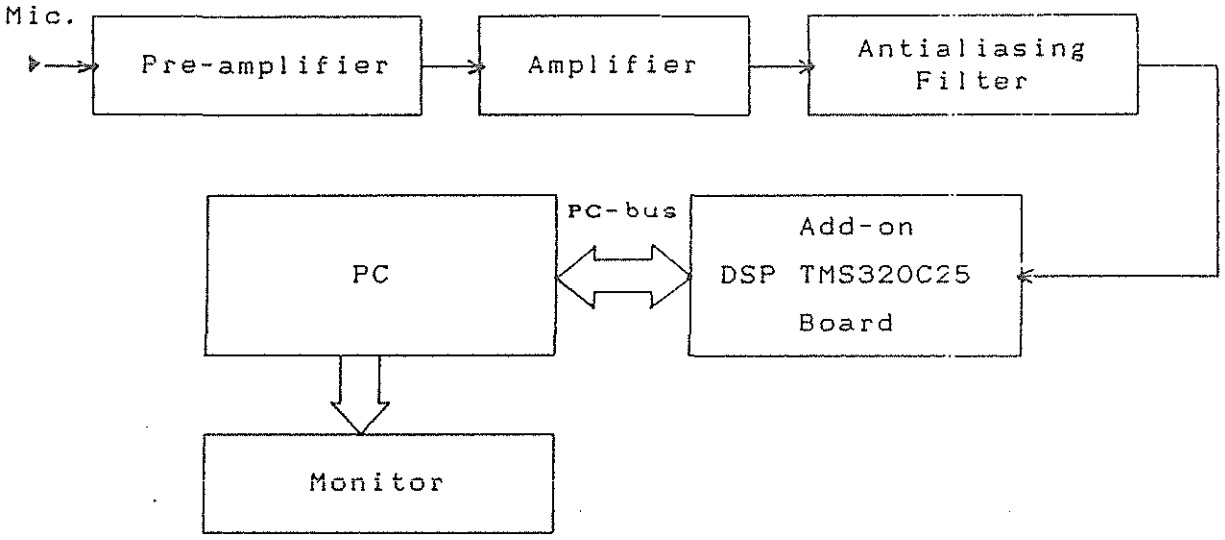


Fig. 3.9. Hardware setup for speech training aid.

CHAPTER 4

SOFTWARE DEVELOPMENT AND TESTING

After the choice of appropriate algorithms for estimation of vocal tract shape, programs implementing these algorithms were developed. These programs were implemented on the hardware setup chosen for real time implementation of the aid as described in previous chapter. The analysis of input speech is carried out first by computing autocorrelation coefficients followed by computing reflection coefficients using Leroux-Gueguen algorithm. Area function is then computed from these reflection coefficients and is displayed on the screen in real time.

The algorithms for extracting vocal tract area function were first developed in programming language C and were tested on PC using discretized speech data files. Same algorithms were then encoded in assembly language of TMS320C25 and were tested by comparing the results with those obtained on PC for same input files. This was followed by writing a program for acquisition and analysis of speech signal in real time. Finally, a program running on a PC was developed to display the area function on the screen in real-time mode and slow motion review mode. The following sections in this chapter describe these programs and lists the test results.

4.1. SPEECH ANALYSIS PROGRAMS ON PC

Programs for analysis of discretized speech files to extract the vocal tract shape were developed in programming language C. The input speech files used by these programs are in the following format.

Line no. 1 → Number of samples stored in the file (N).

Line no. 2 → Data sample 1.

Line no. 3 → Data sample 2.

⋮ ⋮

Line no. N → Data sample N.

All the sample values are 16-bit integers. File can be either in ASCII text or in binary format.

The analysis program first scales the input samples to cover the range available for 16-bit integer representation. Pre-emphasis filter is applied to this input samples with a typical pre-emphasis coefficient value of 0.9. First p autocorrelation coefficients are then computed by the program (where p is the predictor order). Predictor coefficients and reflection coefficients are computed using Durbin's algorithm and Leroux-Gueguen algorithm from these p autocorrelation coefficients. Area function is computed from the reflection coefficients and the results are displayed on the screen. Each algorithm was written as a function in a library SPANLS.C which is compiled and linked with the main program.

4.2.1. Program for LPC Analysis

The program LPC.C computes and displays the predictor coefficients and reflection coefficients using Durbin's and Leroux-Gueguen algorithm. Input speech file is treated as a single frame by the program. Application of pre-emphasis filter and a Hamming window is optional and the predictor order can be set by the user. Program is implemented in following steps.

1. Open the data file, read first sample as total number of samples (N), and allocate memory to store N integers.
2. Read the data values from the file and find maximum and minimum value (maxval and minval respectively).
3. Calculate scaling factor (F) using the following equation

$$F = \frac{32767}{\text{greater of } |\text{maxval}| \text{ and } |\text{minval}|}$$

   Scale the data by multiplying each sample with factor F.
4. Analyze the data through following steps.
   a. Compute first p autocorrelation coefficients.
   b. Compute predictor coefficients by Durbin's algorithm.
   c. Compute predictor coefficients by Leroux-Gueguen algorithm.
5. Display predictor coefficients and reflection coefficients obtained by both the algorithms.

4.1.2. Program for Display of Area Function

Program AREAFN.C displays area function on the screen by analyzing a digitized speech file frame-by-frame. Frame length chosen is 300 samples which is equivalent to 30 ms for a sampling rate of 10 k samples/second. The application of pre-emphasis filter and Hamming window is optional. Analysis of each frame is done by Leroux-Gueguen algorithm to compute reflection coefficients and area function from autocorrelation coefficients for predictor order set by the user. Area function computed for each frame is displayed on the screen.

Program is implemented in following steps.

Steps 1 to 4 same as described for program LPC.C. except for step 4(b).

5. Calculate area function from the reflection coefficients.
6. Initialize the display in graphics mode.
7. Display area function as sections of tubes having same length but width proportional to the area of the tube.
8. Read next 300 samples from the speech file.

11. Repeat steps 3 through 7 until all the frames are analyzed and the area function displayed on the screen.

4.1.3. Testing of Programs

Results of the program LPC.C were first verified by comparing with those obtained from manual calculation for a data file of 5 samples and predictor order 3. Then program TSRS.C was written to generate a time series using equation.

$$y(n) = x(n) - \{ a_1 * y(n-1) + a_2 * y(n-2) + a_3 * y(n-3) + a_4 * y(n-4) + a_5 * y(n-5) + a_6 * y(n-6) + a_7 * y(n-7) + a_8 * y(n-8) + a_9 * y(n-9) + a_{10} * y(n-10) \}$$

$$x(n) = 1 \quad \text{for } n = 0$$

$$= 0 \quad \text{otherwise}$$

This time series is analyzed by functions implementing Durbin's and Leroux-Gueguen algorithm, available in library SPANLS.C. Table 4.1 lists the results as obtained by this program. It can be noted that, the predictor coefficients obtained by both the algorithms are equal to that used actually in the equation.

For testing the program AREAFN.C, it is necessary to analyze a synthesized data file representing sampled acoustic signal as an output from a tube of uniform cross-section. Program WAF.C was written to synthesize and analyze the data . Synthesis of the data is done by taking input as formant frequencies and

corresponding bandwidths stored in a text file. This synthesized data is analyzed by implementing autocorrelation and Leroux-Gueguen algorithm available in library SPANLS.C. The results obtained by this program are as shown in Fig. 4.1. It indicates that the area function estimated by the program is constant as expected.

Second test for this program was carried out by estimating the area function for different vowels, recorded, digitized, and stored in files. Area function obtained for these vowels are shown in Fig. 4.2. It can be seen that area function for back vowels /a/ and /u/ has increasing value towards the lips while that for front vowels /i/ and /e/ has decreasing value towards the lips.

4.2. SPEECH ANALYSIS ON TMS320C25

Algorithms for estimation of vocal tract area function from input speech signal were encoded in assembly language of TMS320C25 and tested using data files, by comparing the results with that obtained on PC for the same data files. Following subsections describe the steps involved in these programs and measures taken to avoid overflow or underflow while computing results in fractional fixed point arithmetic.

4.2.1. Computation of Autocorrelation Coefficients

Program AUTO.ASM calculates first p autocorrelation coefficients where p is the predictor order. The program assumes first data sample at address location 0400h as total number of samples loaded into the memory followed by the actual data samples.

The autocorrelation coefficients are calculated by the program in following steps.

1. Read the first data sample to set the counter for counting number of data samples to be analyzed.
2. Compute zeroth autocorrelation coefficient.
3. Set n = p.
4. Compute n^{th} autocorrelation coefficient, divide it by zeroth autocorrelation coefficient for normalization, and store it in the external data memory location.
5. Decrement n.
6. Repeat step 4 and 5 till $n \geq 1$.
7. Store value of 7FFFh for zeroth autocorrelation.

Autocorrelation coefficients are stored in the external data memory in following format.

Address	Value
1E00-p	R(p)
⋮	⋮
1DFE	R(2)
1DFF	R(1)
1E00	R(0)
1E01	R(1)
1E02	R(2)
⋮	⋮
1E00+p	R(p)

This format for storage of autocorrelation coefficients is specifically chosen according to the requirement of Leroux-Gueguen algorithm for calculation of reflection coefficients.

While computing each autocorrelation coefficient, result of each multiplication operation is scaled by shifting it right by 6 bits. This is required to prevent overflow or underflow that may occur due to accumulation operation. However, overflow or underflow occurring at any intermediate step during accumulation does not affect the final result if the overflow mode of processor TMS320C25 is reset (Chassing and Horning, 1990). Autocorrelation coefficients computed for a data file are listed in Table 4.2 along with that obtained on PC for the same file.

A data file is loaded in the external memory of the DSP board by a program LMDSP.C running on PC. This program scales each data sample to cover the range of 7FFFh to 8000h by calculating a scaling factor from the maximum and minimum values, before loading it into the external memory. Program SMDSP.C reads the results from the external memory and stores it in a file after converting it to its fractional representation.

4.2.2. Computation of Reflection Coefficients

Program KC.ASM computes reflection coefficients from first p (predictor order) normalized autocorrelation coefficients by using Leroux-Gueguen algorithm. Since all the intermediate and final results are guaranteed to be within the range of +1 and -1, no overflow or underflow occurs in these computations (Leroux and Gueguen, 1977). Output values of the reflection coefficients are stored in the external data memory at starting address of 1F00h. Program SMDSP.C is used to store these results in a file. Table 4.3 shows the results obtained for a particular data file by the PC and the signal processor. Less deviation in the results obtained on TMS320C25 indicate that computation of reflection coefficients implemented in fractional fixed point arithmetic is sufficiently accurate. Therefore, all necessary computations are implemented in fractional fixed point arithmetic in the programs written for estimation of area function.

4.2.3. Computation of Area Function

Program AR.ASM computes area function from the reflection coefficients stored from address 1F00h. Since there is a possibility of overflow, it must be avoided by appropriately scaling the results at all intermediate steps. The area function is normalized at the end by dividing each value by the maximum value. Since the vocal tract is modeled as cascaded cylindrical tubes, square root of the area function has to be computed for displaying the tubes as sections with their width proportional to their diameter. Therefore, the normalized area function is scaled by shifting right each value by 5 bits to get a maximum value of 1023 so that the square root of area function will lie within 0 and 31 which is the maximum window width chosen for real-time updating of the image.The program calculates the area function in following steps.

1. Set the area function at glottis equal to 7FFFh (0.999969), i.e., to maximum value in fractional fixed point representation.
2. If reflection coefficient is negative (k_n<0), then
   a. Compute $(1/2 + k_n/2)$.
   b. Compute $(1/2 - k_n/2)$.
   c. Compute $\left\{ (1/2 + k_n/2) / (1/2 - k_n/2) \right\}$

d. Compute

$$A_{n-1} = A_n \frac{(1/2 + k_n/2)}{(1/2 - k_n/2)}$$

e. Store the present area value and go to step 4.

3. If reflection coefficient is positive ($k_n > 0$), then

a. Compute $(1/2 + k_n/2)$.

b. Compute $(1/2 - k_n/2)$.

c. If $A_n > (1/2 - k_n/2)$, then go to step g, else

d. Compute $$\frac{A_n}{(1/2 - k_n/2)}$$

e. Compute

$$A_{n-1} = \frac{A_n}{(1/2 - k_n/2)} * (1/2 + k_n/2)$$

f. Store the value and go to step 4.

g. Compute multiplying factor

$$MF = \frac{(1/2 - k_n/2)}{A_n}$$

h. Scale all the previously calculated areas by MF.

i. $A_{n-1} = (1/2 + k_n/2)$.

4. Repeat steps 2 and 3 till all the area values are calculated.

5. Find maximum area value.

6. Divide all area values by the maximum area value and shift each to right by 5 bits.5 bits.

7. Find the square root using look-up table stored at an address 8000h in external data memory.memory.
8. Store the area values at starting address 2000h.

4.2.4. Display of Area Function

This program creates an image of cross-section of the tubes cascaded to each other, in the external data memory of the DSP board. Two separate memory blocks for odd and even scan lines are generated for a single image. Subroutine PUTPIX is used to write a dot at specified x and y co-ordinates within a window of 192 * 32 pixels. Following are the steps involved in this program.

1. Clear both the memory blocks by writing C000h to these memory locations.
2. Set x co-ordinate to zero.
3. Read first area function as y co-ordinate.
4. Call PUTPIX.
5. Increment x co-ordinate.
6. Repeat steps 4 and 5 till x co-ordinate is incremented 14 times.
7. Read next area value as new y co-ordinate and repeat steps 4 through 6 till all area values are displayed.

4.3. REAL TIME ESTIMATION OF AREA FUNCTION

Program SP.ASM estimates and creates an image of vocal tract area function for input speech signal in real time. Speech received by microphone is suitably amplified, filtered by an antialiasing filter, and digitized by ADC at a sampling rate of 10 k samples /second. Program runs in following steps.

1. Initialize the timer for sampling rate of 10 k samples/sec.
2. Initialize interrupt register to unmask INT1 interrupt.
3. Enable interrupt.
4. Store one sample for each interrupt for frame fr.A in buffer buf.A.
5. If less than 300 samples are stored, then go to step 4.
6. Analyze fr.A and simultaneously store one sample for frame fr.B in buffer buf.B. on each interrupt.
7. If analysis of fr.A is not complete, then go to step 6.
8. Create an image of area function for fr.A in memory block img.A , store area values and energy value in memory block blk. A, and simultaneously continue to store samples for fr.B.
9. Write to port 0 to trigger image and data transfer to PC.
10. Store samples for fr.B in buf.B.
11. If 300 samples are not acquired, then go to step 10.

12. Do steps same as 6 through 9, but for samples of fr.B stored in buf.B and the corresponding image generated in memory block img.B and the area values stored in memory block blk. B, while the sample stored on each interrupt is for fr.A in buf.A.
13. Repeat steps 4 through 12 till PC sends a hold signal to the processor.

This program was tested by providing a speech input through a microphone and obtaining the display of area function on a PC screen in real time. Displays obtained for different vowels spoken by the author are illustrated in Fig. 4.3. Upper line of the rectangle is a fixed reference and the distance of each horizontal line segment from this reference, is proportional to the diameter of the tube at that point. The diameter is maximum near the lips for back vowels /a/ and /u/ while it is minimum near the lips for front vowels /e/ and /i/. In addition, the diameter at the lips for /u/ is less than that for /a/ possibly due to the lip rounding in case of /u/.

4.4. PROGRAM FOR SPEECH TRAINING

Program FINAL.C displays the vocal tract area function and energy in speech signal per frame in real-time and slow motion review modes. This program is linked with library file

STI25DEV.C which provides functions for interfacing with the add-on card. The main program is implemented in following steps.

1. Display the main menu and ask for real-time display, review from a file, or exit to DOS.
   If exit to DOS go to step 15.
   If review is from a file, read the area function values from the specified file and go to step 13.
2. Initialize the add-on card and load the object module SP.MPO into to external program memory.
3. Load window coefficients from file H300.BIN. This file is generated by program HAM.C. These coefficients are loaded at an address location 7000h in the data memory.
4. Load look-up table for finding square root of integer numbers 0 to 1024 at address location 8000h.
5. Reset the signal processor and instruct to run the program.
6. Wait for signal processor to complete the analysis and image generation.
7. Transfer 12 values of area function and one value of energy to an array.
8. Transfer the image to the display memory.
9. Locate the energy marker appropriately on the energy scale.

10. Toggle the address of the image memory block.
11. Repeat steps 6 through 10 till a key is hit on the keyboard.
12. Ask for storage of area function values to a file.
    If yes, ask for file name and store area function and energy values from the array.
13. Ask if review is required.
    If yes, ask for slow motion or frame-by-frame display and set the display mode according to the response.
14. Ask for continuing in real time mode.
    If yes, go to step 4.
15. Exit to DOS.

This program was tested for different vowels spoken by three different individuals and consonants in VCV context spoken by the author. Similar area function was observed for a particular vowel spoken by different speakers, as shown in Fig. 4.4. Each rectangle is as an approximation of the vocal tract with the top edge as the upper palate, glottis at the left, and the lips at right. To locate the variations in the area function of the preceding and following vowels in a VCV context, one consonant from each group with a specific place of articulation was chosen and frames preceding and following this consonant were observed in review mode. Sample results obtained are as shown in Fig. 4.5 (Appendix B has all the results). Variation in area

function of the vowel can be noted in immediately preceding or following frame or in both. No area function can be obtained when there is a complete constriction in the vocal tract before a stop consonant is pronounced.

Table. 4.1. Results of program TSRS.C for analysis of a time series.

Equation for time series

```
y[n]=x[n]-(a1*x[u-1]+a2*x[u-2]+a3*x[u-3]
          +a4*x[u-4]+a5*x[u-5]+a6*x[u-6]+a7*x[u-7]
                    +a8*x[u-8]+a9*x[u-9]+a10*x[u-10])
```

Input signal x[n]= 100 for n = 0
= 0 otherwise

i	ai	a-durb	a-lg
1	+2.980000	+2.980039	+2.979992
2	+4.710000	+4.710009	+4.709833
3	+4.600000	+4.599630	+4.599270
4	+2.890000	+2.888696	+2.888204
5	+1.070000	+1.067470	+1.066973
6	+0.200000	+0.196639	+0.196244
7	-0.000000	-0.003215	-0.003472
8	-0.000000	-0.002205	-0.002340
9	-0.000000	-0.001008	-0.001060
10	-0.000000	-0.000247	-0.000258

Table.4.2. Results of autocorrelation algorithm.

Column A - Results obtained on PC.

Column B - Results obtained on TMS320C25.

A	B
+1.000000	+0.999969
+0.471564	+0.471558
+0.033175	+0.033173
-0.035545	-0.035522
-0.123223	-0.123199
-0.220379	-0.220369
-0.080569	-0.080566
+0.182464	+0.182434
+0.234597	+0.234569
+0.056872	+0.056854
-0.068720	-0.068695
-0.011842	-0.011814
+0.068720	+0.068695

Table.4.3. Results of Reflection coefficients.

Column A - Results obtained on PC.

Column B - Results obtained on TMS320C25.

A	B
+0.131274	+0.131348
+0.348746	+0.348969
-0.031574	-0.031036
+0.130501	+0.131012
+0.272727	+0.273976
+0.216923	+0.218872
+0.091870	+0.095154
-0.003672	-0.000061
+0.080706	+0.084442
+0.058257	+0.062561
-0.116276	-0.111023
-0.043392	-0.038940

Formant freq. in Hz.	Bandwidths in Hz.
500	100
1500	100
2500	100
3500	100
4500	100

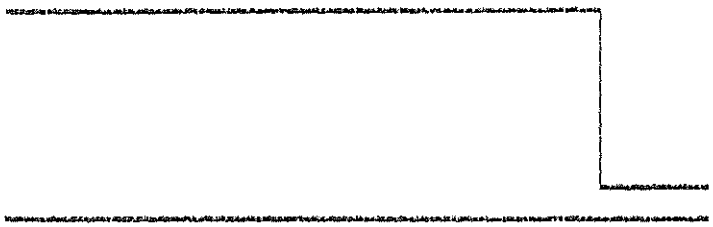


Fig. 4.1. Area function for a tube of uniform cross-section.

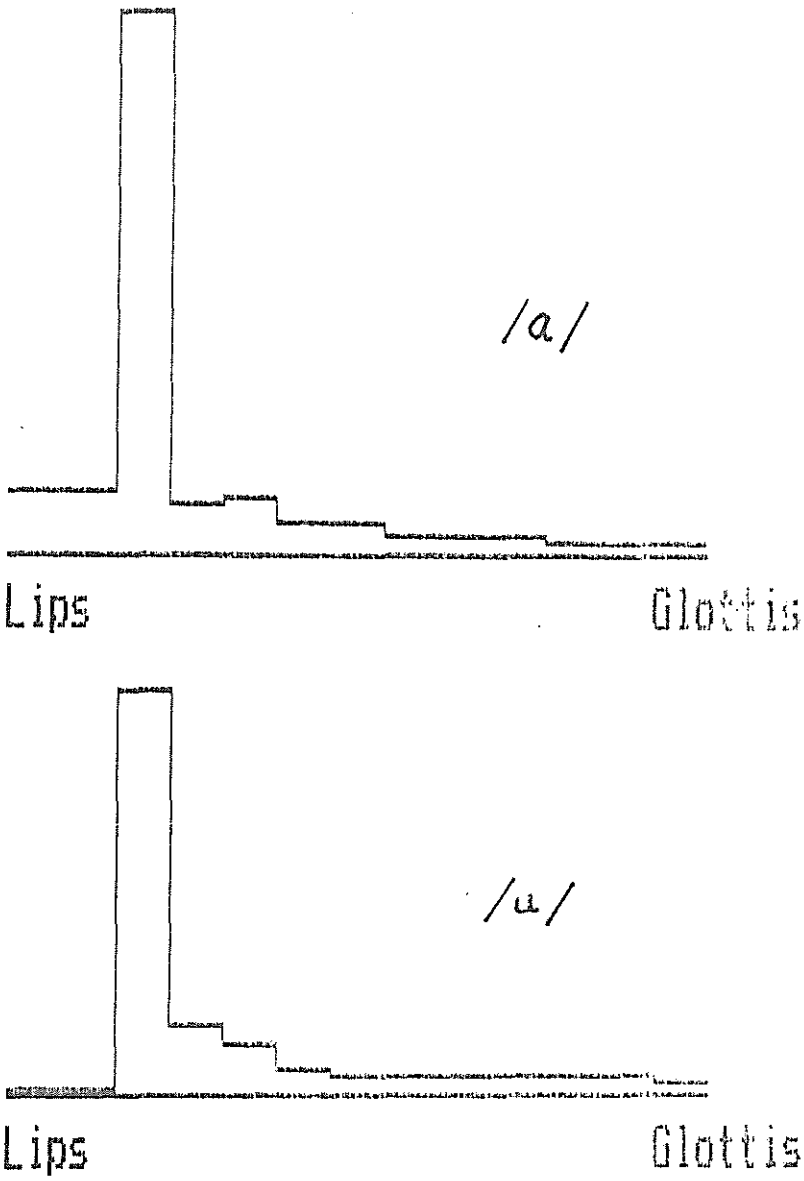


Fig. 4.2 (a). Area function for vowel /a/ and /u/.

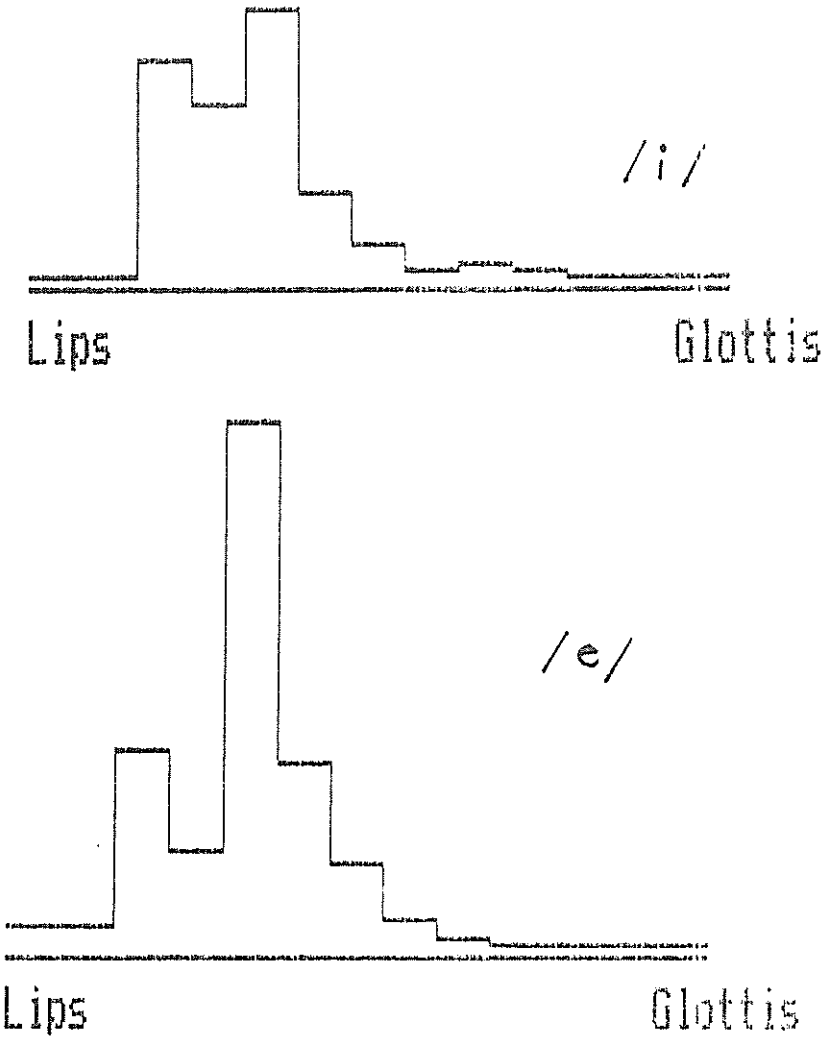


Fig. 4.2 (b). Area function for vowel /i/ and /e/.

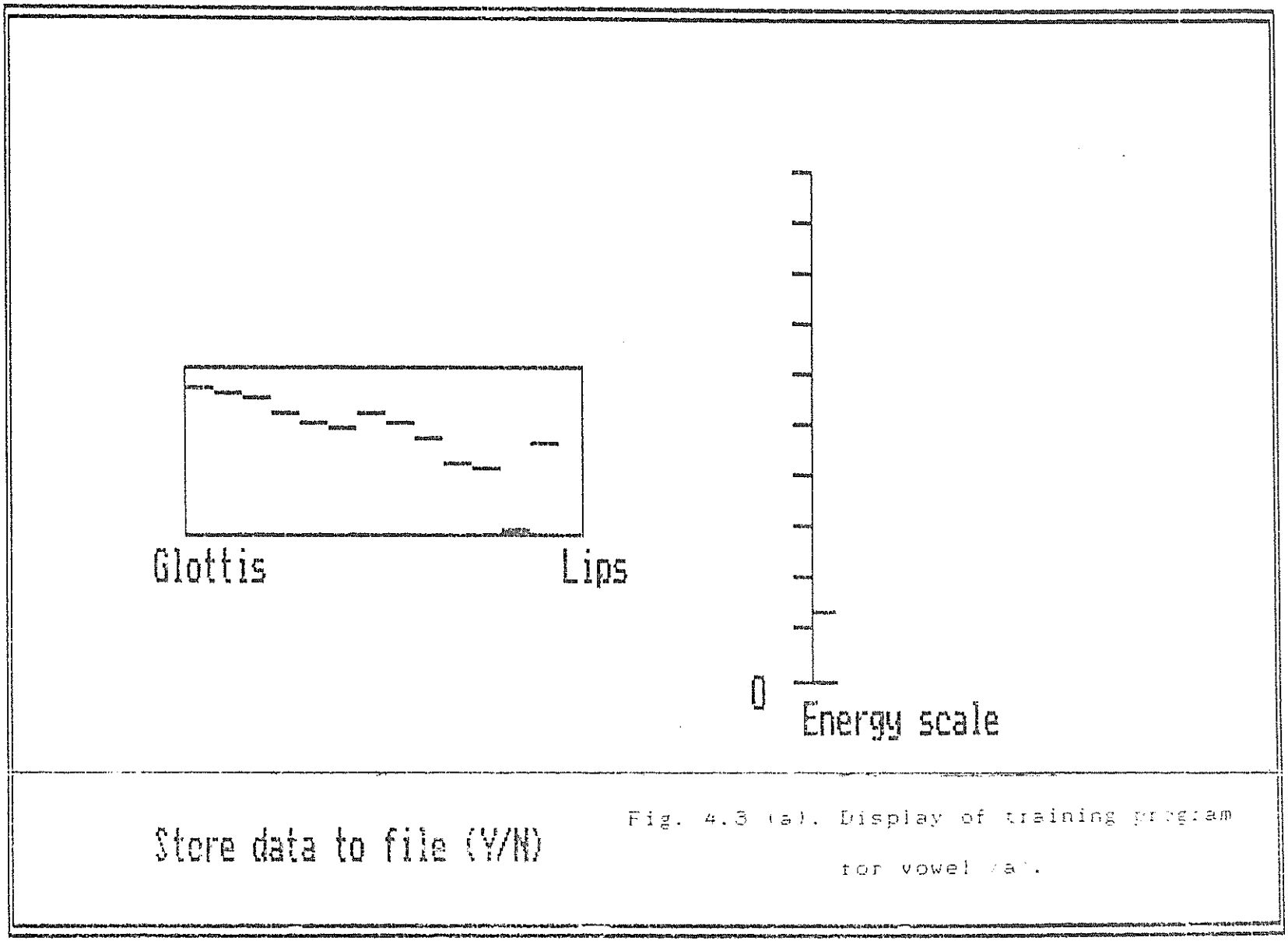


Fig. 4.3 (a). Display of training program for vowel /a/.

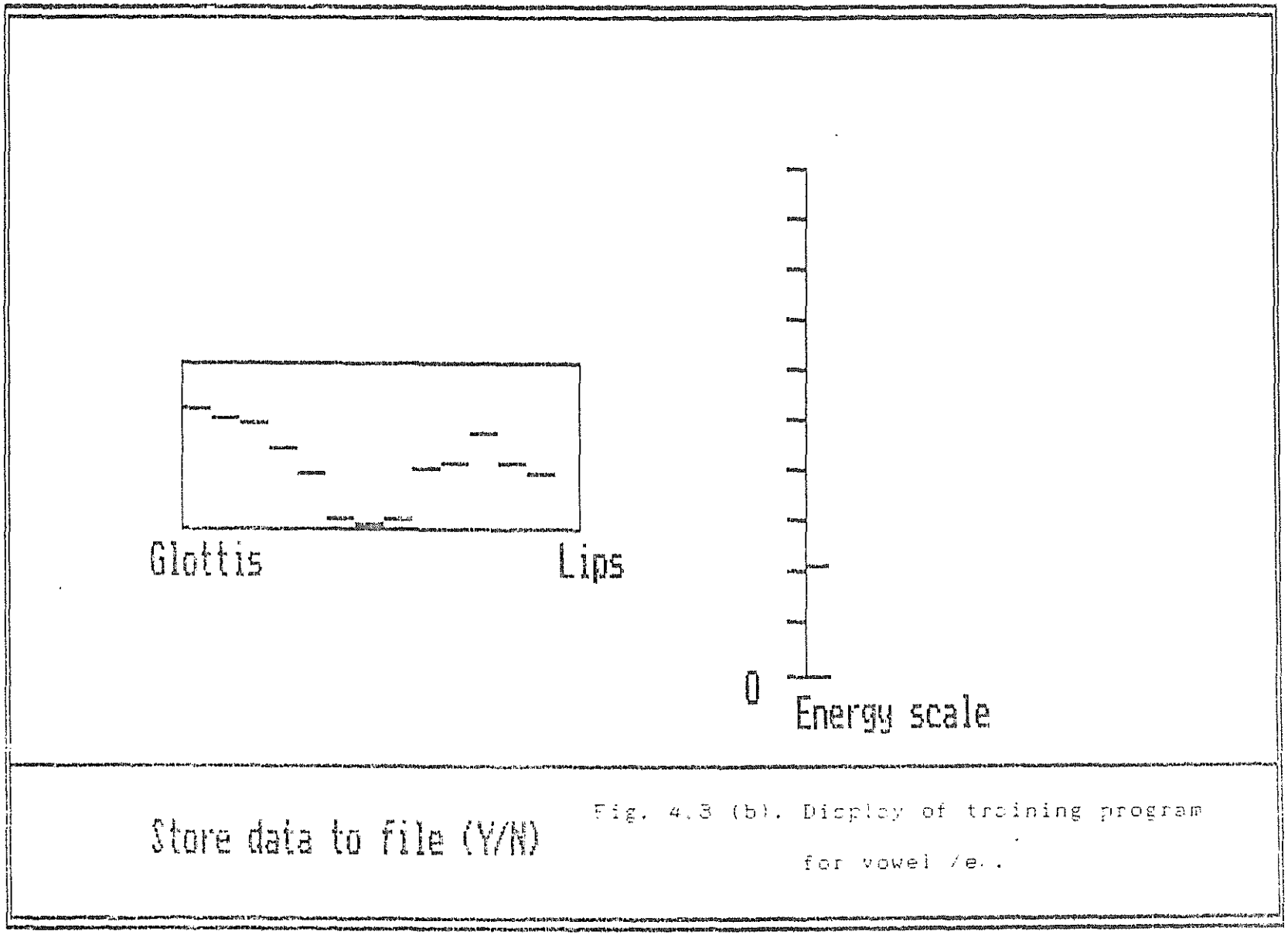


Fig. 4.3 (b). Display of training program for vowel /e..

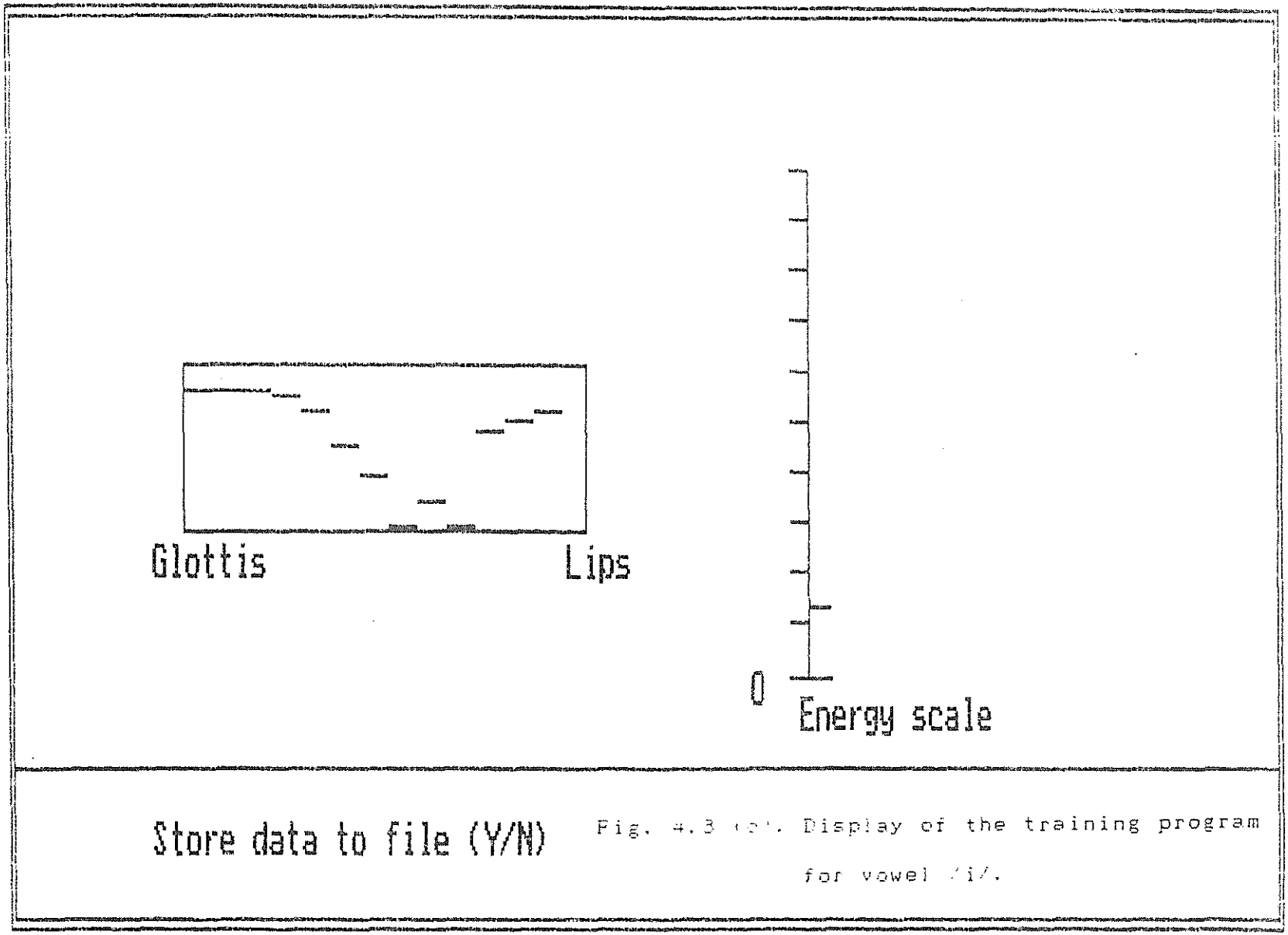


Fig. 4.3 (b). Display of the training program for vowel /i/.

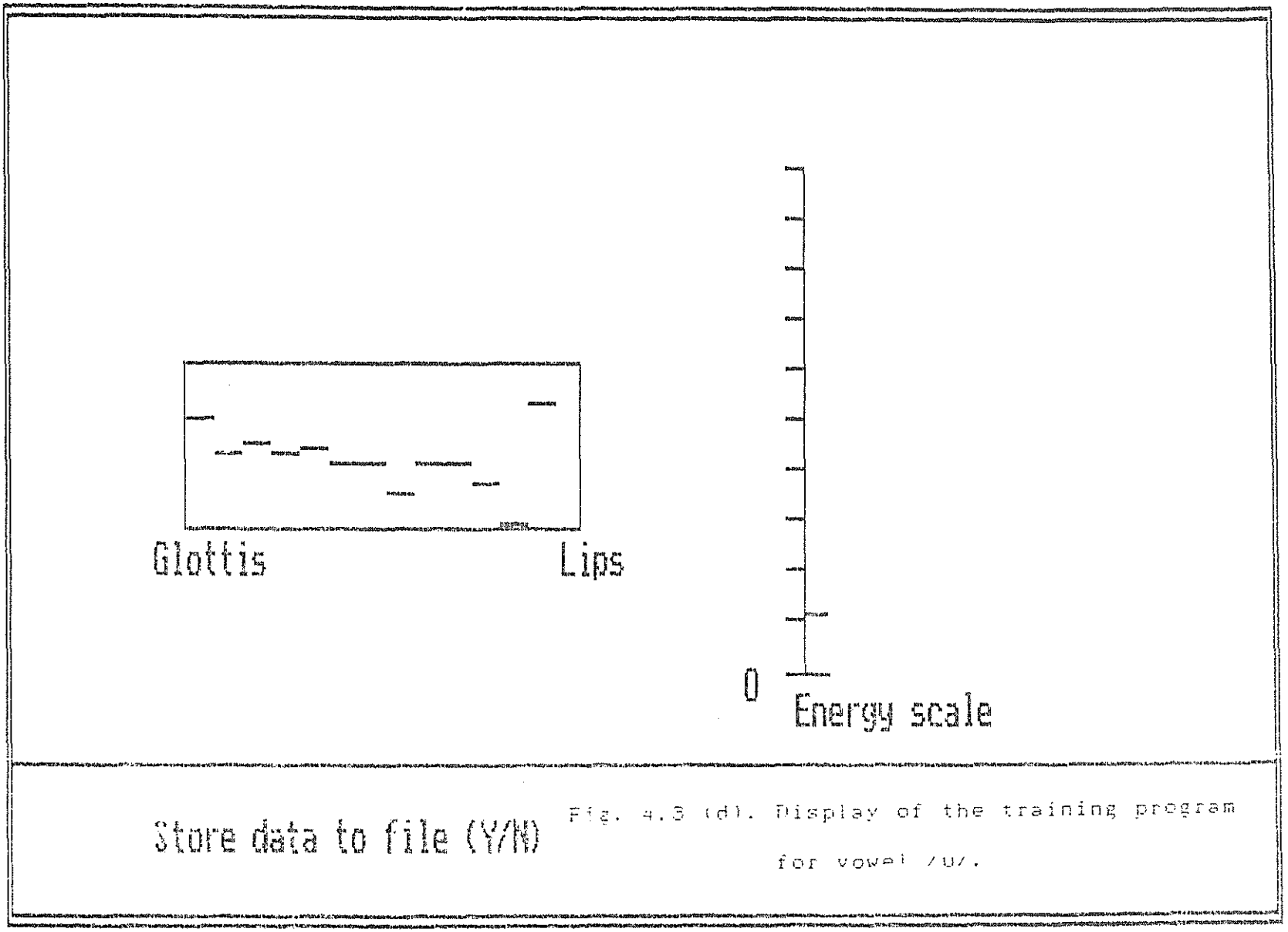


Fig. 4.3 (d). Display of the training program for vowel /U/.

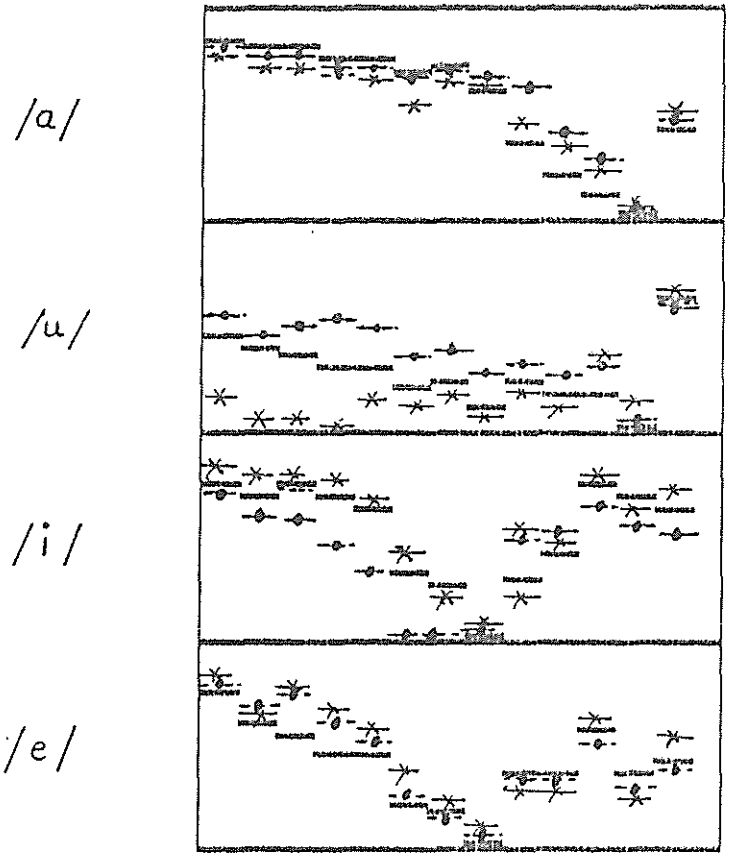


Fig. 4.4. Area function for vowels spoken by three speakers.

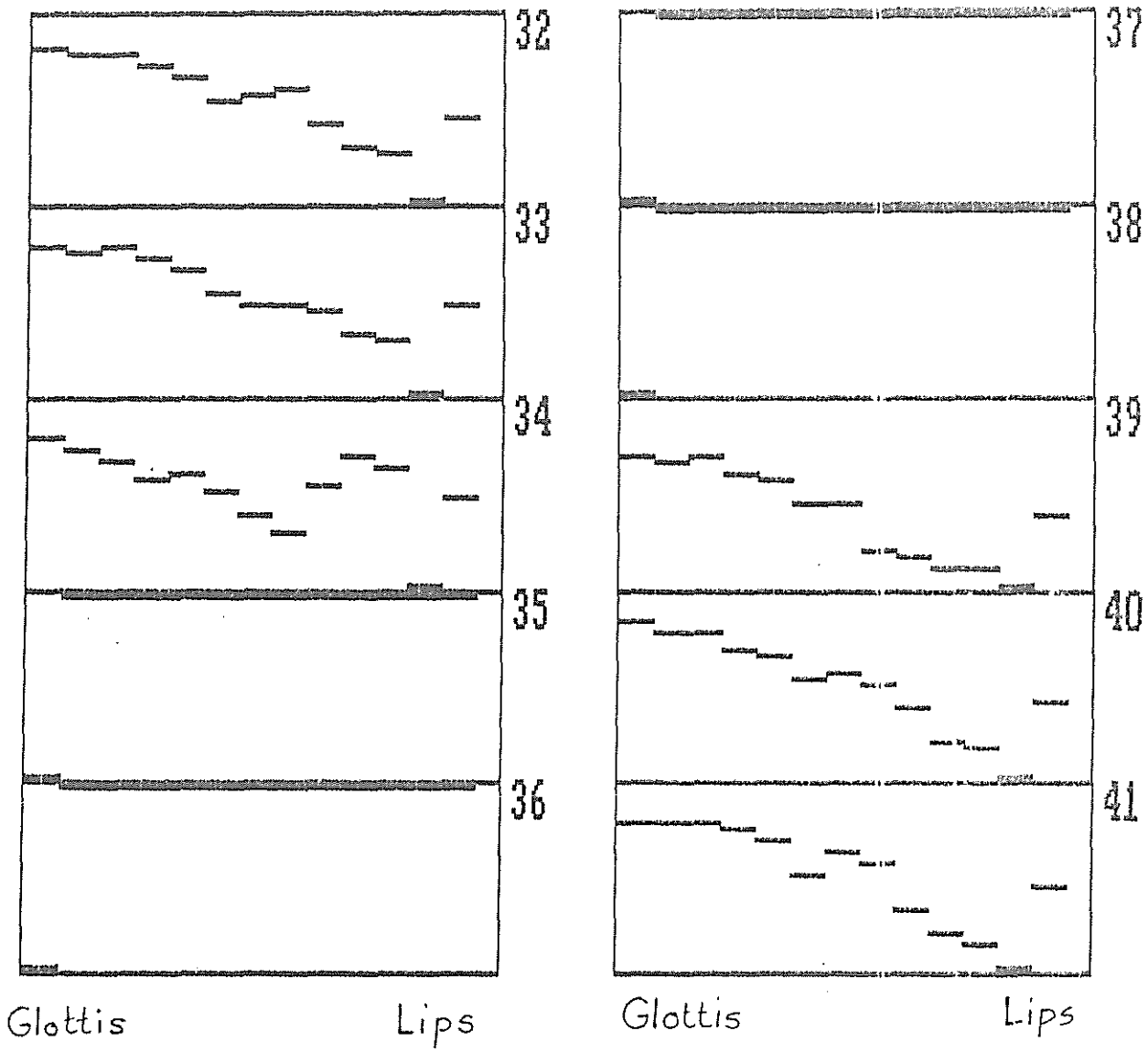


Fig. 4.5. Area function for VCV sequence /ata/.

(Numbers on right indicate frame numbers)

CHAPTER 5

SUMMARY AND CONCLUSION

The objective of this project is to develop a PC based speech training aid using display of vocal tract shape and energy. This can help deaf children to learn the movements of articulators while speaking, since the vocal tract shape indicates the actual position of the tongue, jaw, lips etc. Information about the energy variations is useful in improving upon prosodic characteristics of speech.

As a first stage in the development of the aid, software and hardware requirements for implementation of the aid were studied. Linear predictive coding was chosen as the technique for estimation of vocal tract area function. Leroux-Gueguen algorithm was found suitable for computing reflection coefficients from normalized autocorrelation coefficients in fractional fixed point arithmetic. Area function was then computed from these reflection coefficients.

A PC add-on DSP board PCL-DSP25 (from Dynalog Microsystems Ltd.) using TMS320C25 processor chip (from Texas Instruments) was found suitable for implementation of vocal tract shape display on PC screen in real time. The image of the vocal tract shape, derived for each frame of input speech signal, is

generated in the external memory available on the DSP board and then transferred to display memory of PC along with area function and energy values for storage to a file.

Program for estimation of area function was first developed in programming language C and was tested using discretized speech files. Then a program was written in assembly language of TMS320C25 for real-time estimation and display of vocal tract shape. Finally, a master program running on PC was written to implement the speech training aid in real-time and in slow motion review mode.

The real-time display of vocal tract shape was successfully implemented using the DSP board. The area function for different vowels spoken by different individuals is found to be matching. Variations in the area function are observed in the frames of the vowel immediately preceding or following the consonant in VC or CV utterances respectively. However, no area function can be obtained at the instant when there is a complete constriction in the vocal tract before a stop consonant is uttered.

Presently system displays the vocal tract shape as sections of cylindrical tubes cascaded to each other and placed in a straight line. This display can be further modified to make

it realistic in appearance by suitably interpolating between the discrete area values. The interpolating algorithm can first be implemented on PC in off-line mode and then tried out on the DSP board for real-time display. Display of pitch can be added to the present display in a suitable form, by implementing a suitable pitch extraction algorithm on the DSP board.

The technique of linear predictive analysis, although successfully derives the area function for vowels, fails to extract information about place of articulation for a stop consonant. This is due to absence of output speech signal during the complete constriction in the vocal tract before pronouncing a stop consonant. By monitoring the energy variations in the speech signal, for a sudden drop in its value, one should be able to detect the frame of the preceding or following vowel having variation in area function due to transitions in VC or CV utterances respectively. It may be possible to derive the place of articulation for the consonant by locating the distance at which the area function has a minimum value in the detected frame. Hence in review mode, one can substitute the display of image acquired during the stop period by display of an image that indicates the place of articulation.

REFERENCES

Atal B. and Hanauer S. (1971). Speech analysis and synthesis by linear prediction of the speech wave, J. Acoust. Soc. Am., vol. 50(2), p 637.

Bernstein L., James B., Fergusson J. III, and Goldstein M. Jr. (1986). Speech training devices for profoundly deaf children, Proc. ICASSP, p 633.

Bristow G., Brooks S., Fallside E., Gulian, and Hinds P., Design and assessment of computer based speech training aids using vocal tract display, (Source not traceable).

Chassing R., and Horning D. (1990). Digital Signal Processing with the TMS320C25, John Willey, New York.

Crichton R., and Fallside F. (1974). Linear prediction model of speech production with applications to deaf speech training, Proc. IEE Control & Science, vol 121, p 865, Reprinted in Levitt et al (1980), p 399.

Furui S. (1989). Digital Speech Processing, Synthesis, and Recognition, Markel Dekker Inc., New York.

Gracias S. (1991). A Speech Training Aid for the Hearing Impaired, B. Tech. project report, Dept. of Elect. Engg., IIT Bombay.

Gupte M. (1990). A Speech Processor and Display for Speech Training of the Hearing Impaired, M. Tech. dissertation, Dept. of Elect. Engg., IIT Bombay.

IBM Corp. (1983). IBM PC/XT Technical Reference Manual (2.02).

King A., Parker A., Spanner M., and Wright R. (1982). A speech display computer for use in school of deaf, Proc. ICASSP, p 755.

Leroux J., and Gueguen C. (1977). A fixed point computation of PARCOR coefficients, IEEE Trans. ASSP, vol.25, p 257.

Levitt H., Picket J., and Houde R., Eds. (1980). Sensory aids for the Hearing Impaired, IEEE Press, New York.

Liberman A.,Cooper F., Shankweiler D., and Studdert M. (1968). Why are speech spectrograms hard to read ?, Am. Anr. Deaf., p 127, Reprinted in Levitt et al (1980), p 287.

Markel J. and Gray A. Jr. (1976). Linear Prediction of Speech, Springer-Verlag, New York.

Nickerson R. (1975). Characteristics of the speech of deaf persons, The Volta Review, p 342, Reprinted in Levitt et al (1980), p 540.

Pandey P., Abel S., and Kunov H. (1986). Effect of auditory delay on audio-visual speech perception, Proc. Int. Cong. Acoust., p A5-3.

Pandey P. (1987). Speech Processing for Cochlear Prosthesis, Ph. D. thesis, Dept. of Elect. Engg., University of Toronto, Canada.

Pardo J. (1982). Vocal tract shape analysis for children, Proc. ICASSP, p 763.

Pardo. J., Aguilera S., Barrojo A., and Munoj E. (1986). Speech analysis based devices for diagnosis and education of speech for hearing impaired, Proc. ICASSP, p 641.

Parsons T. (1987). Voice and Speech Processing, McGraw-Hill, New York.

Picket J., Gengel R., and Quinn R. (1974). Research with the Upton eyeglass speechreader, Proc. Sem. Sp. Comm., Reprinted in Levitt et al (1980), p 324.

Rabiner L. and Schafer R. (1978). Digital Processing of Speech Signals, Prentice-Hall, Englewood Cliffs, New Jersey.

Shigenaga M., and Kubo H. (1986). Speech training systems for handicapped children using vocal tract lateral shape, Proc. ICASSP, p 637.

Stark R., and Goldstein M. (1976). Modification of vocalization of preschool deaf children by vibrotactile and visual displays, J. Acoust. Soc. Am., vol 59(6), p 1477, Reprinted in Levitt et al (1980), p 282.

Takalikar K. (1991). A Speech Training Aid for the Deaf, M. Tech. dissertation, Dept. of Elect. Engg., IIT Bombay.

Texas Instruments. (1987). Second-Generation TMS320 User's Guide, Dallas, Texas.

Upton H. (1968). Wearable eyeglass speechreading aid, Am. Ann. Deaf, vol 113, p 222, Reprinted in Levitt et al (1980), p 322.

Wakita H. (1973). Direct estimation of the vocal tract shape by inverse filtering of acoustic speech waveform, IEEE Trans. Audio & Electroacoustics, vol 21, p 417.

Wakita H. (1979). Estimation of vocal tract shapes from acoustical analysis of speech wave: the state of art, IEEE Trans ASSP, vol. 27, p 281.

APPENDIX A

A.1. DURBIN'S ALGORITHM

This algorithms is used to obtain predictor coefficients from the autocorrelation coefficients in recursive manner. In each step, we already have a solution $\alpha_{n-1}(i)$ for a predictor order (n-1), and we use this solution to compute the coefficients $\alpha_n(i)$ for n^{th} order predictor. We can start recursion at n=0, for which the equations are

$$[R_o][1] = [E_o]$$

Hence $E_o = R_o$

The algorithm can be stated as follows:

1. For n = 0, $E_o = R_o$.
2. For step n,

   a. $$k_n = \frac{-1}{E_{n-1}} \sum_{i=0}^{n-1} \alpha_{n-1}(i)\, R_{n-1}$$

   b. $$\alpha_n = k_n$$

   c. For i=1 to n-1

   $$\alpha_n(i) = \alpha_{n-1}(i) + k_n\, \alpha_{n-1}(n-i)$$

   d. $$E_n = E_{n-1}(1 - k_n^2)$$

A.2. LEROUX-GUEGUEN ALGORITHM

The normal equations for the order p linear predictor are

$$\sum_{i=0}^{p} \alpha_p(i)\, R_{i-j} = 0, \qquad j = 1,2,\ldots,p$$

If we consider a general sequence

$$q_p(j) = \sum_{i=0}^{p} \alpha_p(i)\, R_{i-j}$$

This sequence has following properties.

1. If p=0, $q_p(j)$ is the same as R_{i-j}.
2. If p>0, $q_p(j)$ is zero for j=1,2,...,p.
3. $q_p(0)$ is the order p prediction error E_p.
4. For successive values of p, q obeys the recursive solution

   $$q_p(j) = q_{p-1}(j) + k_p\, q_{p-1}(p-j)$$

   where, k_p is the p^{th} reflection coefficient.
5. $|q_p(j)| \leq R_0$, with equality only if p = j = 0.

These properties lead to a recursive solution to the autocorrelation equations.

1. For $-p < i < p$, $q(i) = R_{|i|}$
2. For step n;
   a. $k_n = - q_{n-1}(n) / q_n(0)$.
   b. For i = n-p to p,

      $$q_n(i) = q_{n-1}(i) + k_n\, q_{n-1}(n-i)$$

This recursion was devised by Schur in 1917 and subsequently rediscovered by Leroux and Gueguen in 1977 (Parsons, 1989). If we normalize the R's by dividing through by R_o, then all the quantities in the recursion are of magnitude <1 (except R_o) and can be represented as fractions. For computation by hardware, fixed point arithmetic is simpler and faster than floating-point arithmetic. In fixed point arithmetic, we would prefer to work with either all-integer or all-fraction data and thus this algorithm is suitable for fractional fixed point implementation used in this project.

Algorithms for extracting vocal tract area function were first developed in programming language C and are available in library file SPANLS.C. The functions available in this library file are as follows.

1. Autocorrelation function - autocor (p1,p2,p3,p4)

p1 - no. of samples in the input signal.

p2 - predictor order.

p3 - pointer to input signal array.

p4 - pointer to array of autocorrelation coefficients.

2. Durbin's algorithm - lpcd (p1,p2,p3,p4,p5)

p1 - predictor order.

p2 - pointer to array of autocorrelation coefficients.

p3 - pointer to array of predictor coefficients.

p4 - pointer to array of reflection coefficients.

p5 - pointer to array to residual error energy.

3. Leroux Gueguen algorithm - lpclg (p1,p2,p3)

p1 - predictor order.

p2 - pointer to array of autocorrelation coefficients.

p3 - pointer to array of reflection coefficients.

Some other miscellaneous functions in this library are as follows.

1. Memory allocation - allocate (p1)

p1 - size of the memory to be allocated in bytes.

This function returns the pointer to the allocated memory only if sufficient memory is available.

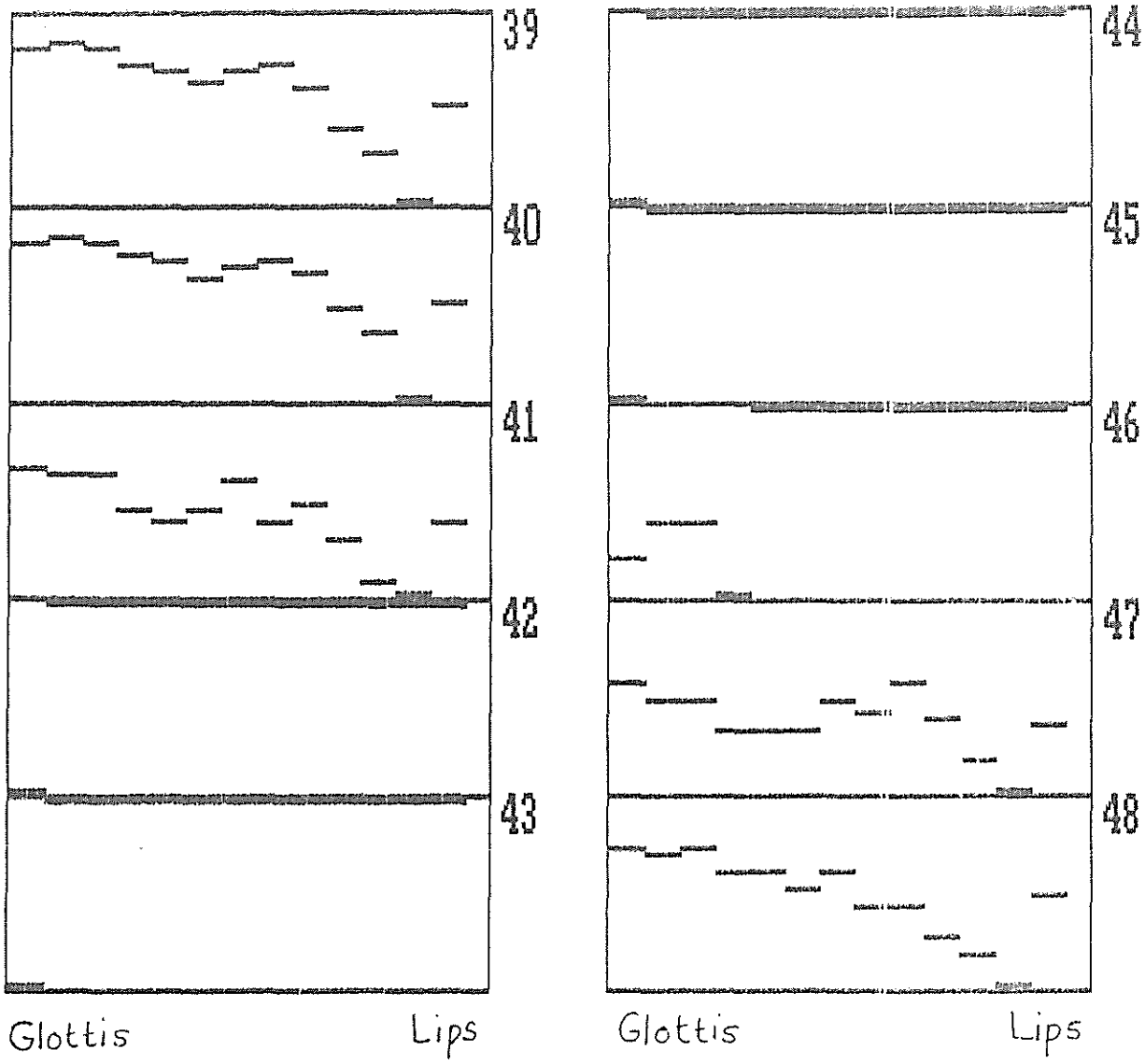
2. Initialize graphics mode (CGA high resolution) - cgainit().

This function initializes the graphics driver and set the graphics mode to high resolution (640 * 200 pixels) B/W mode.

This library is compiled to object file SPANLG.OBJ and is linked with the programs using these functions.

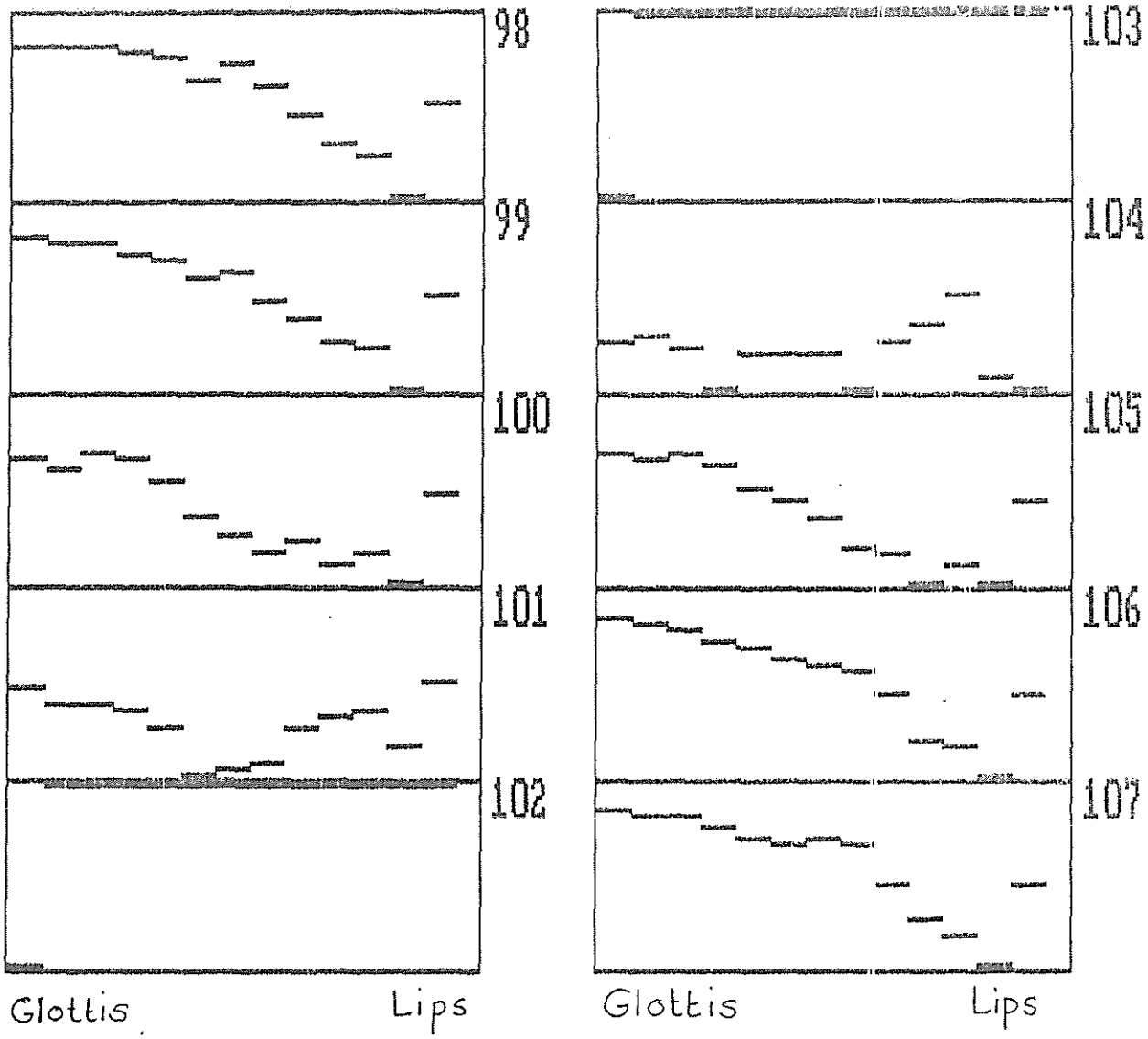
APPENDIX B

AREA FUNCTION FOR VOWEL CONSONANT VOWEL SEQUENCES



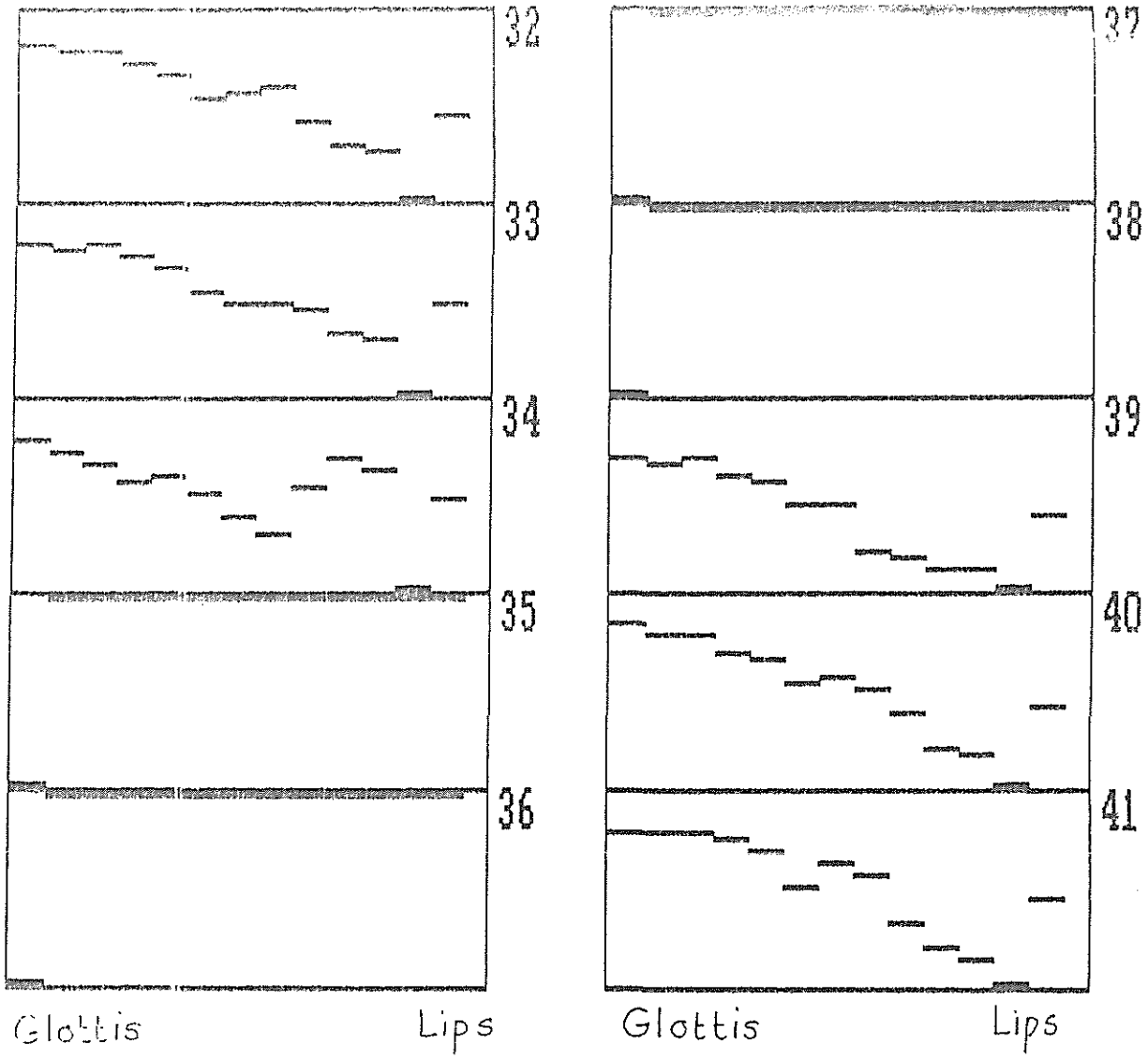
B.1. Area function for /a k a/.

(Numbers on right indicate frame numbers)



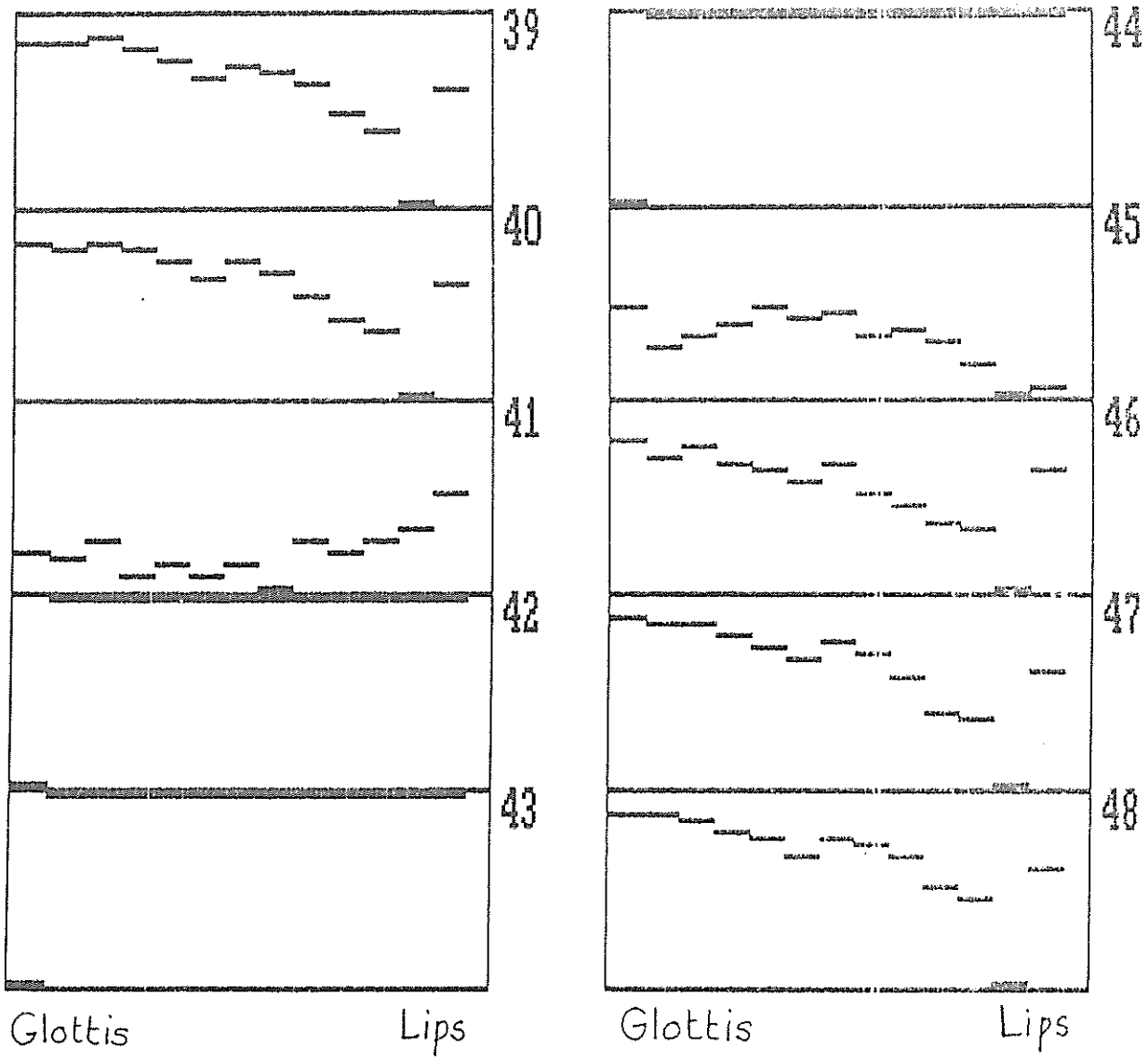
B.2. Area function for /a ch a/.

(Numbers on right indicate frame numbers)



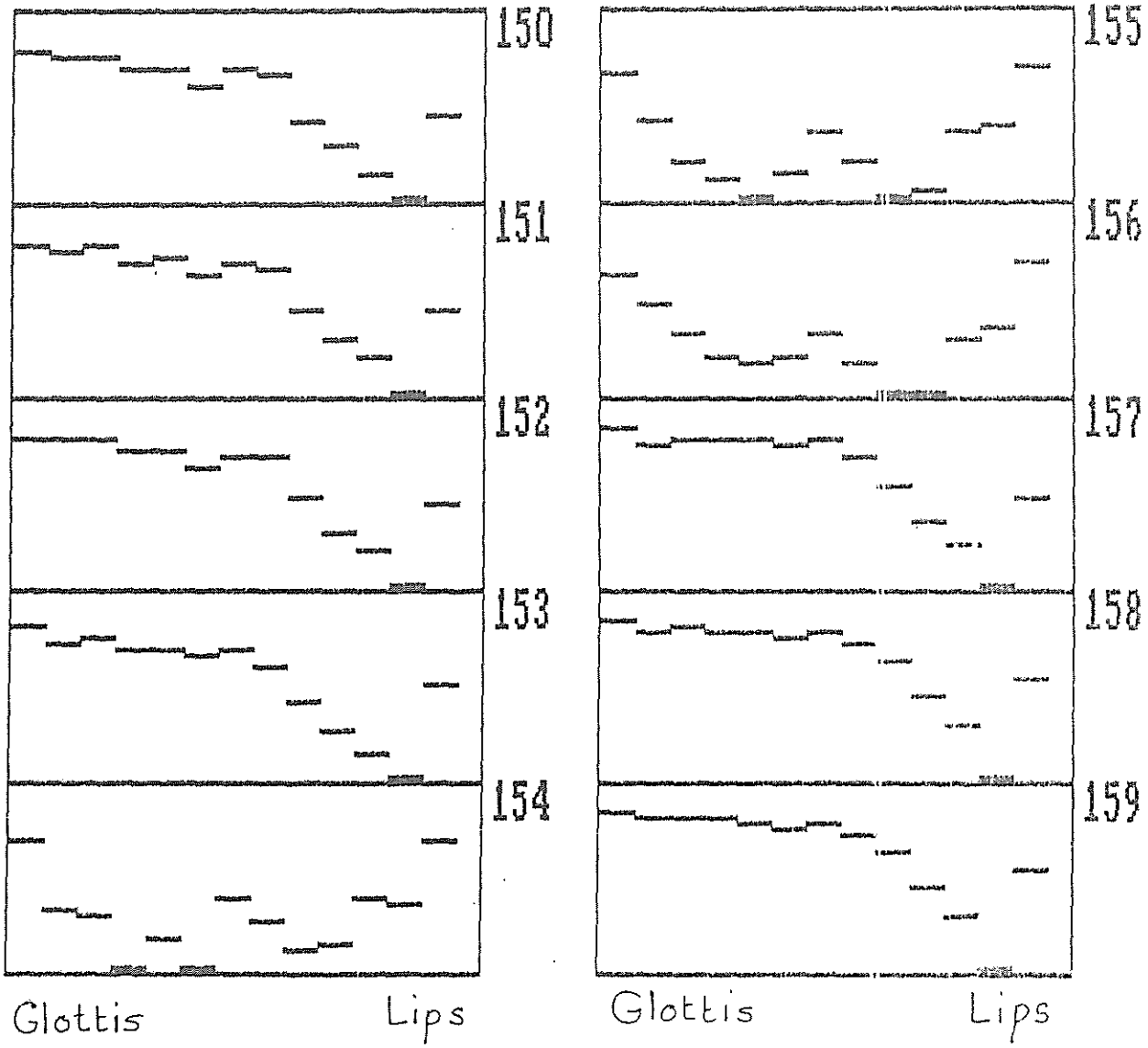
B.3. Area function for /a t a/.

(Numbers on right indicate frame numbers)



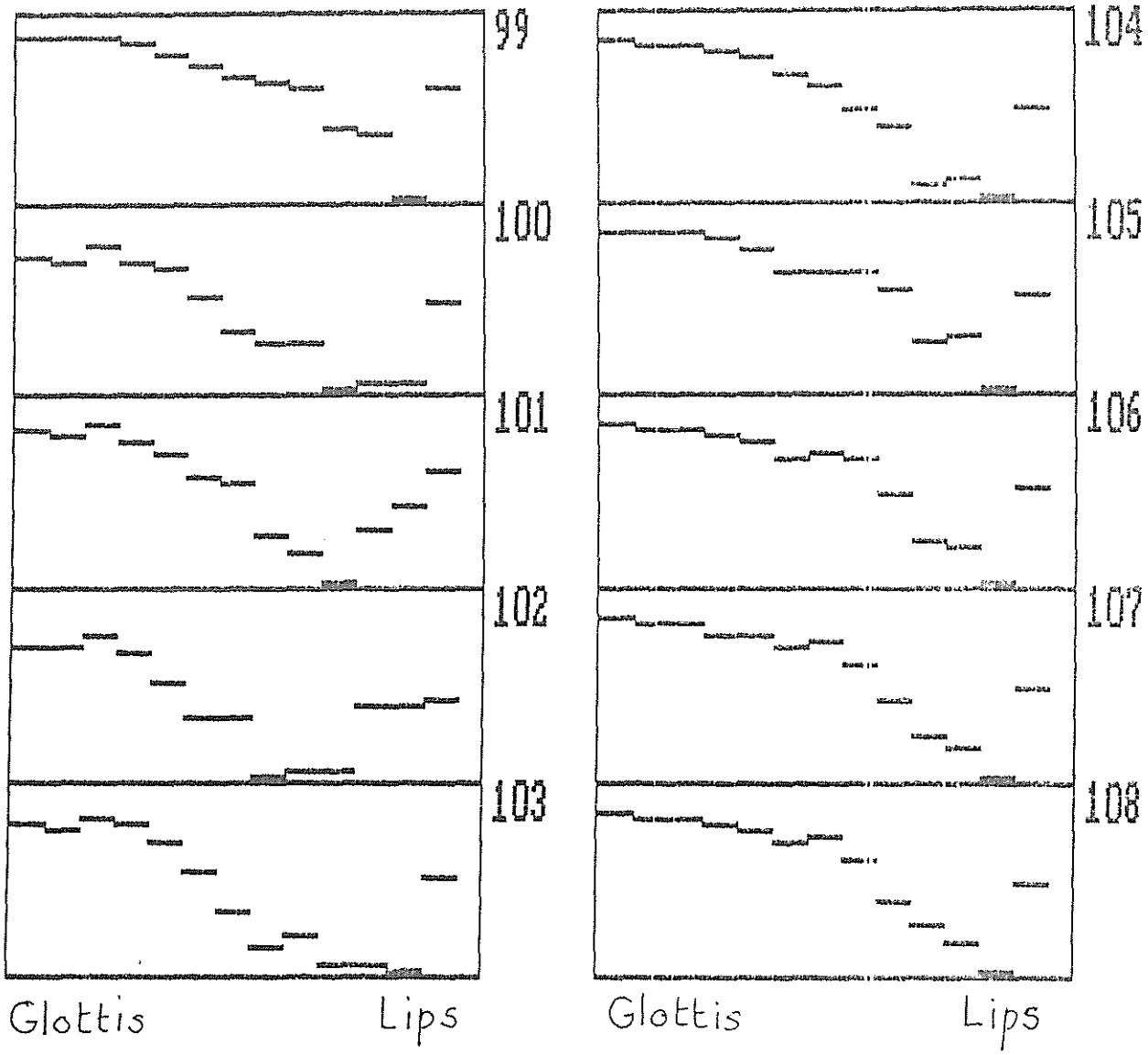
B.4. Area function for /a p a/.

(Numbers on right indicate frame numbers)



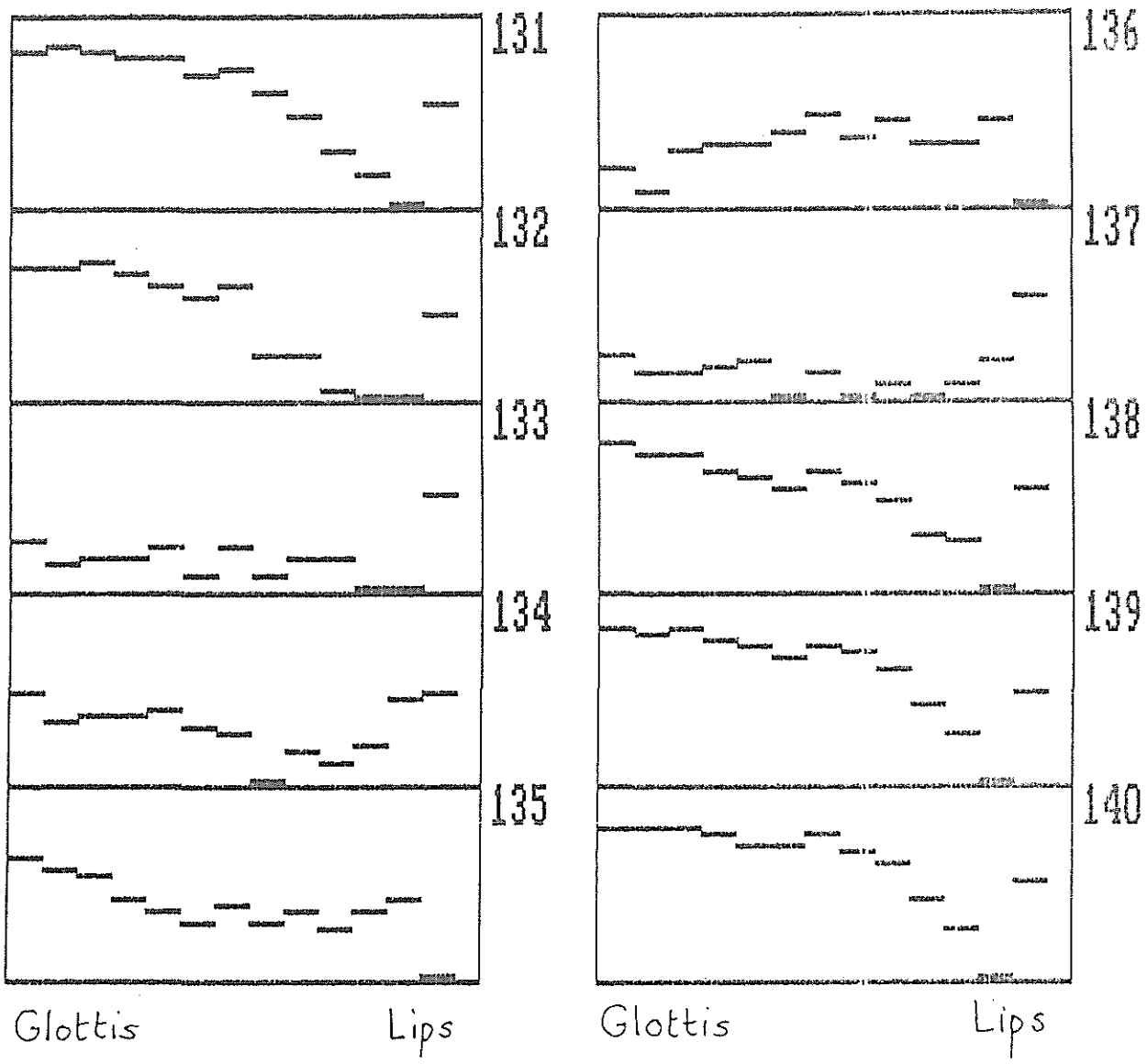
B.5. Area function for /a m a/.

(Numbers on right indicate frame numbers)



B.6. Area function for /a y a/.

(Numbers on right indicate frame numbers)



B.7. Area function for /a s a/.

(Numbers on right indicate frame numbers)