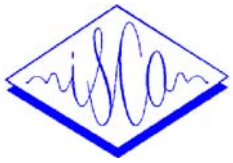

6th European Conference on
Speech Communication and Technology
(EUROSPEECH'99)
Budapest, Hungary, September 5-9, 1999

JavaSpeechLab – INTERACTIVE SPEECH ANALYSIS LABORATORY ON THE WORLD-WIDE WEB

Andrzej Drygajlo and Guy Delafontaine***

Signal Processing Laboratory*, Computer Aided Learning Laboratory**,
Swiss Federal Institute of Technology, Lausanne, Switzerland
Andrzej.Drygajlo@epfl.ch, Guy.Delafontaine@epfl.ch

ABSTRACT

The aim of the paper is to introduce a versatile and user-friendly computer assisted learning (CAL) system in order to support traditional teaching in the Speech Science domain. This system is based on Java as a powerful programming language for developing platform-independent, interactive and computational-based software that can be used on the World-Wide Web through a Java-enabled Web browser. The interactive and computational capabilities of Java are demonstrated through modular speech analysis laboratory (JavaSpeechLab) based on Java application and applet. It is designed to transform the WWW page into an easy-to-use speech analysis workstation for distance learning applications.

1. INTRODUCTION

Recent developments in computer networking technologies offer new ways to teach and learn in the Speech Science domain [1]. Speech processing is inherently multimedia in nature, involving both sound and vision, and Speech Science students need to acquire practical skills in listening, analysis and interpretation performance [2, 9]. Various efforts are directed towards creating educational material in multimedia format using Hypertext Markup Language (HTML) and making it accessible world-wide through the Internet [11]. The introduction of the Java programming language [10] extends the capability of HTML documents by allowing developers to integrate platform and browser independent features into Web documents that were formerly available only through user-installed platform-dependent applications. The key to a successful speech analysis training is to create tools which open up the field to interactive investigation by the student. Java provides users with the freedom to create interactive content for the Web by developing new data types and the methods to operate on them [3]. Java applets give the Web the power of continuous, interactive, real-time, visual, and aural instruction. Web documents come alive because the computer can respond instantly to a user's input. The computer can now become an active participant and a computational tool in the interactive learning process rather than a clumsy screen book.

The authors are currently using Java to develop a prototype interactive learning tool for the speech analysis laboratory. The interactive learning tool is called JavaSpeechLab [4]. This tool can be used for conventional classroom experiments, in the students' laboratory or can provide self-study means for further exploration which can help students understand speech processing phenomena and facilitate their analysis [8].

2. BASIC CONCEPTS FOR SPEECH ANALYSIS TEACHING

Desirable features of CAL systems for spoken language engineering (SLE) training result from an integrated approach to teaching and learning in this domain [2]. To introduce the SLE background in the framework of undergraduate courses at the Electrical Department of the Swiss Federal Institute of Technology Lausanne in Switzerland, a course "Speech Processing" (28 hours of lectures and 56 hours of laboratory for 8th term (final year) students, has been given since 1994.

This course is preceded by digital signal and system theory curricula where the basics of digital filtering and Fourier analysis are presented with their applications to stationary signal processing. During a 5 year pedagogical experiment we faced the problem of how to introduce efficiently a concept of "short-term analysis" that is fundamental to most speech analysis techniques. The assumption made is that, over a long interval of time, the speech waveform is non stationary but that, over a sufficiently short time interval, it can be considered stationary. Thus most speech analysis systems operate on a time-varying basis, using short segments of speech selected at uniformly spaced time intervals or frame. Frame duration is defined as the number of samples over which a set of parameters is valid, and the number of samples used to compute the frame parameters is known as the window duration. This basic concept is described by *Equation 1* and presented in *Figure 1*.

$$Q(M \cdot n) = \sum_{m=M \cdot n-N+1}^{M \cdot n} \mathrm{T}\left[x(m) \cdot w(M \cdot n - m)\right] \qquad (1)$$

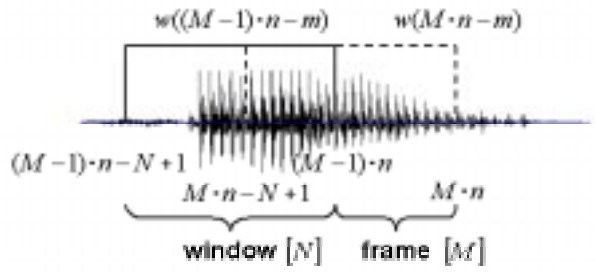

Figure 1. Short-term analysis of speech signal

We also built on our students' digital signal processing (DSP) background and have introduced a method which uses a set of basic block diagrams for creating multirate linear and non-linear processing systems. The presence of down and up samplers allows for different nodes in a diagram to have different respective sampling rates and, in fact, makes them easy to interpret by associating to such structures the concept of frame-by-frame and window-by-window processing. The example of such a diagram for short-term energy analysis, which corresponds to *Equation 2*, is shown in *Figure 2*.

$$E(M \cdot n) = \sum_{m=M \cdot n-N+1}^{M \cdot n} \left|x(m) \cdot w(M \cdot n - m)\right|^2 \qquad (2)$$

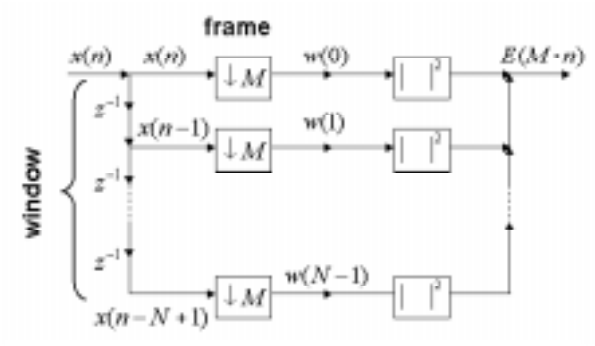

Figure2. Short-term energy analysis diagram

This new digital approach based on the "graphical language of diagrams" allows not only to interpret correctly the classical short-term algorithms eliminating some ambiguities common in speech processing text books but also to construct easily new modern multirate and multiresolution speech analysis algorithms based on filter bank structures [6].

This theoretical modeling and corresponding short-term analysis are presented in the expository mode during lectures [5]. For lectures to be easily understood, we illustrate them through many examples using the JavaSpeechLab package which in this case plays the part of a computer assisted teaching tool [7]. Such a program is practically and theoretically as powerful a representation as an equation. This is particularly appealing in engineering and digital signal processing, which typically use a strong mathematical formulation [12]. Another major advantage is that simulations with sound and/or graphics utilize the learner's sensory input which makes understanding of the material more natural.

In the LTS Speech Processing Laboratory, working with real signals and practical algorithms is necessary to grasp fully the difficulties of speech signal processing, particularly speech analysis. Only a limited amount of time is available for laboratory work, moreover, each experimental set-up requires a non-negligible learning period. For that reason it was necessary to develop a highly user-friendly laboratory work-bench that can be used with most speech analysis experiments and to be completed by new algorithms developed by students themselves. On the other hand the students have access to several computer platforms and the Internet, and they can continue the investigations outside the laboratory. At the beginning, it was not obvious what software choice would be the best. Finally, we have chosen the Java language with its in-built handling of the network protocols required to send and receive data across networks such as the Internet.

3. THE JavaSpeechLab

JavaSpeechLab software package, developed in the LTS, allows the student to build and test new speech analysis algorithms in the laboratory and experiment with speech signals without having to learn a great deal about irrelevant mechanics of programming when these algorithms are already implemented. In this second case, it is designed to transform the WWW page into an easy-to-use speech analysis workstation. It is interactive, easy to learn, and encourages exploration. The main double window allows a user to load and to edit a speech signal (*Figure 3*). A portion of the signal can be selected using two cursors and then zoomed. The zoomed signal is highlighted in the top window while the zoomed zone is displayed in the bottom window. A menu selection and pull-down lists allow the user to specify a speech analysis application in terms of control blocks. Once the user has chosen the parameters in the control block the program can execute and display the results. A reasonably complete set of short-term analysis procedures (time-domain analyses, spectral and time-spectral analyses (Fourier and multi-resolution) (*Figure 4*) [6]) permits a surprising variety of speech decomposition and analysis concepts to be demonstrated. Using playback function (Play button) the user gets immediate feedback about the acoustic content of a chosen signal and therefore can reason about the correlation between this content and the results of the analysis.

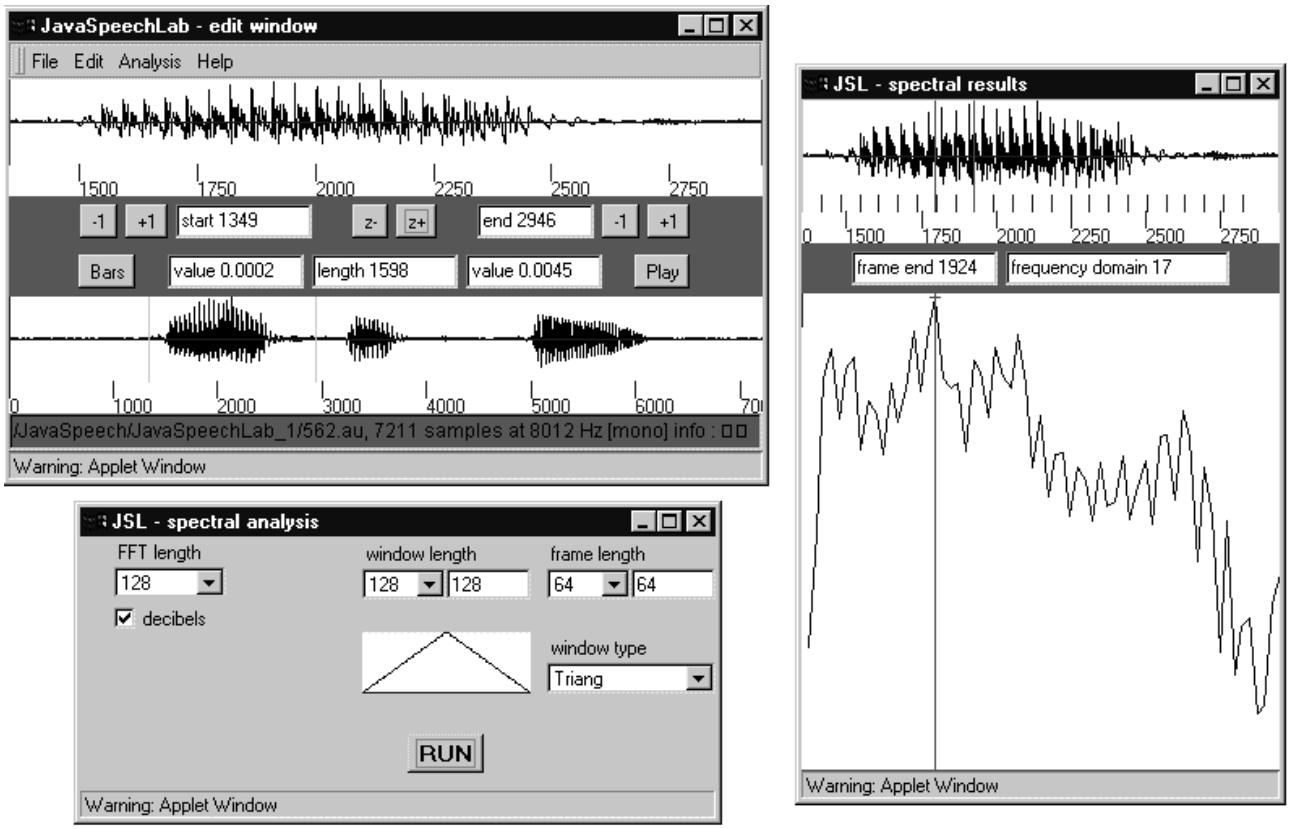

Figure 3. Edition, control and results windows of JavaSpeechLab

In our software laboratory the common frame-window environment is provided for all types of analysis from the simplest one (e.g. short-term energy or zero-crossing rate) to the most complex based on multi-resolution wavelet packet transforms which use different windows and frames at different frequency subbands (*Figure 4*).

Realizing that laboratory application programs are like instruments, except that the user interacts with the terminal rather than a front panel, has led to programmed "virtual instruments" of the JavaSpeechLab package developed for WWW usage. JavaSpeechLab is an excellent CAL work-bench because of its conceptual ease. Students can master it on the Web and then concentrate on their applications without being needlessly burdened with programmatic details. This modular package can be completed by new speech analysis or processing modules using a plug-in technique.

A JavaSpeechLab applet has been implemented to demonstrate how Java could be used to develop an interactive learning environment for speech analysis training on WWW. The Java Developers Kit JDK 1.1 for Windows NT has been used in applet development. A Java-enabled Web browser such as Netscape Navigator 4.0 or Microsoft Internet Explorer version 4.0 is required to run the applet on the Web. Further development is taking place at EPFL and we now use JDK 1.2 which offers many additional facilities such as improved handling of audio files [10].

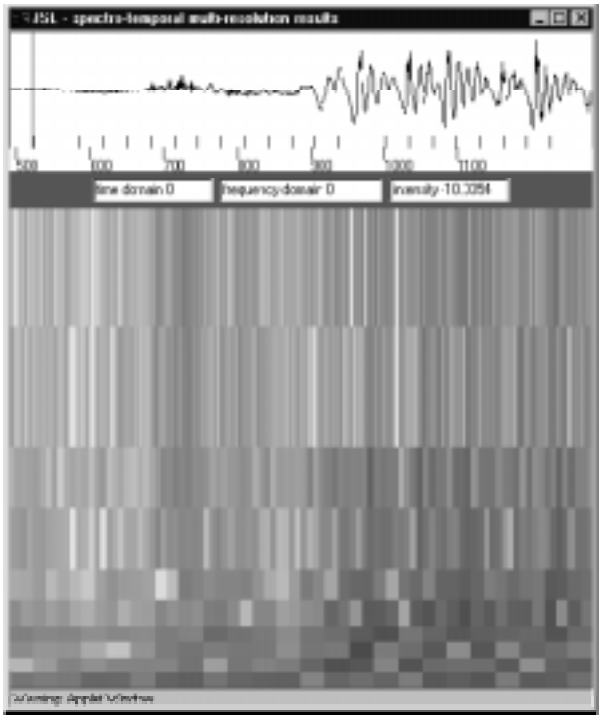

Figure 4. Multiresolution analysis

4. CONCLUSION

The Java programming methodology is new and there remain some inadequacies and bugs. Programming experience and/or great patience and free time is required to turn ideas into Java code. There is a performance penalty resulting from the required code interpretation, which has been reduced by the incorporation of Just In Time (JIT) compilers into Web browsers. Unfortunately, Java is in its infancy and educators working with it must create many of their own classes to handle tasks for which they would be able to obtain ready-to-use libraries in other languages.

However, Java supports a new paradigm in computer programming by allowing users to create platform-independent and Web-browser executable software. The future of development effort on the Web utilizing Java appears to be very promising. The ability of Java applets to be used within WWW pages certainly offers advantages for distance learning [13]. A demonstration applet written in the Java language has been developed for the speech analysis application. It provides fast manipulation of speech analysis algorithms and gives immediate feedback when answers are being compared with concepts. It reduces computational effort and therefore allows students to understand principles. The applet program can be executed on any computer connected to the Internet and running a Java-enabled Web browser by accessing the URL http://scgwww.epfl.ch/JavaSpeechLab.

5. REFERENCES

[1] Bork A. (1997). The Future of Computers and Learning, *T.H.E. Journal*, 24/11: pp. 69-77.

[2] G. Bloothoft et al. (1997). *The Landscape of Future Education in Speech Communication Sciences: 1 Analysis*, SOCRATES/ERASMUS Programme, OTS Publications, Utrecht, 1997.

[3] Cabell B. (1997). Using Java to Develop Interactive Learning Material for the World-Wide Web, *The International Journal of Engineering Education*, http://www.ijee.dit.ie/articles/999971/article.html.

[4] Drygajlo, A. and Delafontaine, G. (1998). JavaSpeechLab – Interactive Assisted Learning Work-Bench for Speech Analysis Training on the World-Wide-Web, *Proc CALISCE' 98*, June 1998, Göteborg, pp. 138-145.

[5] Drygajlo A. (1998). *Traitement de la parole*, EPFL, Lausanne, 1998.

[6] Drygajlo, A. and Thévoz, N. (1995). Multiresolution Speech Analysis Using Fast Time-Varying Orthogonal Wavelet Packet Transform Algorithms, *Proc Eurospeech' 95*, Sep 1995, Madrid, pp. 255-258.

[7] Fleming, M. and HoWard Levie, W. (1993). *Instructional Message Design*, Educational Technology Publications, Englewood Cliffs, 1993.

[8] Guzdial, M. (1998). Making Project-Based Learning Work in Undergraduate Education Technological Support: Lessons in Computer Supported Collaborative Learning, *Proc CALISCE' 98*, June 1998, Göteborg, pp. 3-7.

[9] Huckvale, M. et al. (1997). Opportunities for Computer-Aided Instruction in Phonetics and Speech Communication Provided by the Internet, *Proc Eurospeech' 97*, Sep. 1997, Rhodes, pp. 1919-1922.

[10] JavaSoft (1999). *The Source for Java*, http://www.javasoft.com.

[11] Neilson J. (1994). *Hypertext and Hypermedia*, Academic Press, San Diego, 1994.

[12] Principe, J. et al. (1998). Teaching Adaptive Systems with Interactive, Electronic Book, *Proc CALISCE' 98*, June 1998, Göteborg, pp. 319-324.

[13] Drygajlo, A. and Delafontaine, G. (1999). Using Java to Develop Interactive Learning Work-Bench for Speech Analysis Basics on the World-Wide-Web, *Proc MATISSE' 99*, April 1999, London, pp. 25-28.