Interactive learning in speech and hearing

Stuart N. Wrigley, Martin Cooke and Guy J. Brown

Speech and Hearing Group, Department of Computer Science, University of Sheffield,
211 Portobello Street, Sheffield S1 4DP, United Kingdom.
s.wrigley@dcs.shef.ac.uk, m.cooke@dcs.shef.ac.uk, g.brown@dcs.shef.ac.uk

Abstract

As part of an ongoing appraisal of our courses in speech and computational hearing, our traditional teaching techniques were found to be inadequate in some respects. In particular, they offer little opportunity for students to experiment interactively with concepts in these fields. As a result, students cannot reflect upon their own experiences nor learn from them. In this paper, we discuss the design of a more interactive approach based on software applications which allow the student to explore the parameter space of the phenomena under discussion. Several examples (chosen from a range of more than twenty MATLAB applications) which illustrate the philosophical and pedagogical elements of this project are discussed. These include animations of basilar membrane motion, explorations of concurrent vowel perception, interactive visualisations of cepstral processing and computer-based audiometry training.

Keywords: speech, hearing, MATLAB, interaction.

1. Introduction

Most speech and hearing courses introduce a wide range of new material to students with varying academic backgrounds. Topics ranging from digital signal processing to psychophysics are taught to students studying courses as diverse as electrical engineering, computer science, psychology and linguistics. The multimedia nature of the courses does not lend itself to traditional presentation styles. These characteristics combine to promote numerous misinterpretations (such as confusion between spectra and spectrograms).

At Sheffield, the teaching framework is very much the traditional science and engineering environment of lectures and laboratory classes. Laboratory classes should expand experience of the topics covered in lectures by the use of 'hands-on' demonstrations. Our ongoing objective is to provide a comprehensive learning package which will consolidate students' understanding of the course.

Computer-assisted learning (CAL) is ideally suited to solving this problem as it can, if used appropriately, lend itself to interaction and experimentation in a multimedia environment. In this paper, we describe the design issues involved in producing such a package.

Section 2 describes our previous teaching practice. Sections 3, 4 and 5 examine a new initiative - the MATLAB Auditory and Speech Demonstrations (MAD) programme - which attempts to open up a wide range of topics for exploration by the student. The MAD package now consists of over twenty interactive demonstrations which allow key parameters of a process or phenomenon to be investigated. A core design principle is that all the demonstrations have a uniform interface and documentation, and provide some form of immediate visual or acoustic feedback where appropriate.

2. Previous practice

2.1 Context

Based in a computer science department, it is expected that course material will have a computational bias. Unfortunately, there are few good quality books available which treat speech and hearing topics with such a bias, especially when dealing with physiological and psychological phenomena. As a result we are forced to provide our own resources.

The increasing popularity of the speech and hearing courses has put strain on the system of laboratory classes. A solution is to produce laboratory work which is more self-contained in order to reduce the dependence on a limited number of instructors. Taken to its extreme, this could mean that in association with the course lecture notes a distance learning package could be created. In fact this will soon be reality. Sheffield is a partner in a European Union project to develop a European Masters in Language and Speech Technology [1]. Our two existing speech and hearing courses may form part of this degree programme.

2.2 Teaching resources

Before the introduction of the MAD package, we used teaching materials ranging from videos of the McGurk effect [2] and auditory demonstration CDs [3,4] to a public-domain spectrogram labelling tool [5]. Although these tools proved useful, they did not cover the whole range of topics taught to the students. The majority of them (such as the auditory demonstration CD) were also non-interactive. Perhaps the major failing was the lack of uniformity of the materials we used.

3. The MAD design principles

3.1 Guidelines

The temptation when developing CAL applications is to address specific problems or common misunderstandings. Arguably, this should be avoided for a number of reasons. Not only does it take a significant amount of time to identify misconceptions (usually only evident in coursework or examination scripts) but it can lead to overly focused tools. All this assumes that the designer can identify conceptual problems from their manifestations.

As noted above, the key to a successful suite of tools is that it encompasses the range of topics covered in the lectures. Furthermore, these tools should allow the student to interact with parameters in order to expand their understanding and experience. Our design objective is to create tools which open up the field to interactive investigation by the student. They ought to be able to *explore* the parameter space of the phenomenon and learn from the challenges that they pose it - be they knowingly or unwittingly. The application should produce a two-way communication between the user and the system allowing important concepts to be transmitted indirectly. The core design guidelines are described in table 1.

Table 1. Common core design guidelines

Guideline	Definition
interactive	The user can do more than simply press and watch.
immediate	If possible, the system responds to user actions in real time.
exploratory	An important and relevant section of the parameter space underlying any effect or process is made available to the user for direct manipulation.
uniform	A consistent style of interface design is followed.
easy-to-use	If possible, enable direct manipulation of objects.
documentation -rich	Comprehensive and uniform HTML documentation is available for each demonstration, complete with suggestions for exploration and further reading.

4. Implementation using MATLAB

MATLAB is a high-level programming language which provides facilities for numerical computation, user interface creation and data visualisation.

The ability to create Graphical User Interfaces (GUI) using a drag-and-drop tool coupled with high-level support for sound handling, signal processing and graphical visualisation allow complex multimedia tools to be created relatively quickly. When constructing such a large suite of tools, development time is an important consideration. However, experience has shown that a demonstration tool can be constructed in approximately two to three days once sufficient experience of MATLAB and its GUI creation techniques has been acquired.

Not only does MATLAB provide a level of abstraction which allows the student to concentrate on the fundamental principles, it also promotes 'vectorised' thinking.

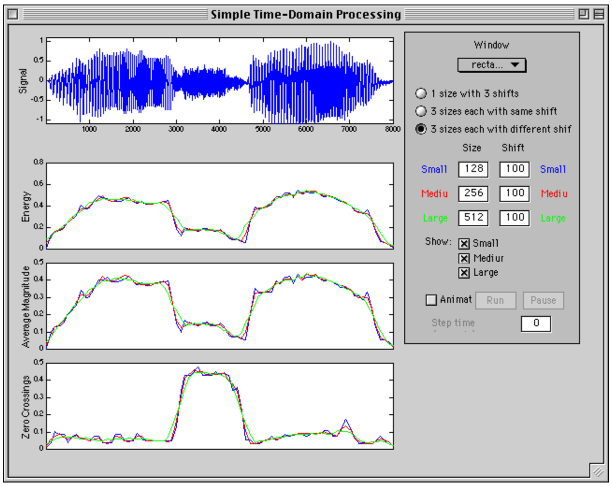
Arguably the most potent reason for choosing MATLAB is the fact that versions are available for most major platforms - Windows, MacOS and UNIX - and no dedicated DSP boards are required as in [7]. Furthermore, a cheap academic edition is available allowing students to use the demonstration tools at home.

The nearest contender to MATLAB appears to be Java. Certainly, its high portability is attractive, but Java has a number of disadvantages when compared to MATLAB. The lack of an equivalent to the MATLAB signal processing toolbox within the Java Applications Programming Interface (API) would significantly increase development time and effort.

The ability of Java applets to be used within WWW pages certainly offers advantages for distance learning. MATLAB cannot run within a web browser. However, the usefulness of any such applet will be limited due to the security limitations enforced by web browsers: applets cannot write to the local disc. Hence, any demonstration which requires the storage of intermediate or final results is not possible.

5. Showcase

This section will briefly introduce six of the MAD demonstrations which have been chosen to illustrate the guidelines in table 1.



Window: recta...

○ 1 size with 3 shifts

○ 3 sizes each with same shift

● 3 sizes each with different shif

	Size	Shift	
Small	128	100	Small
Mediu	256	100	Mediu
Large	512	100	Large

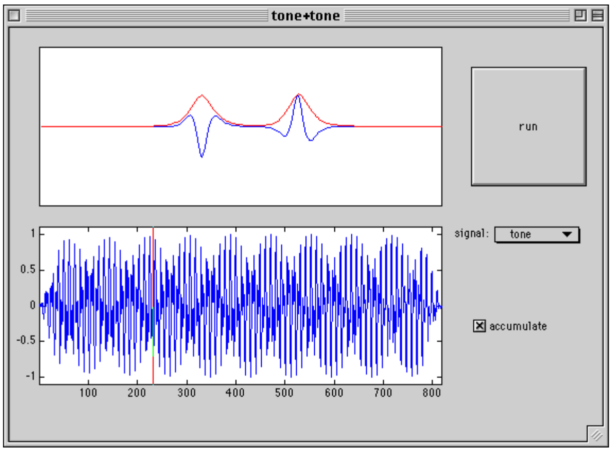
Show: ☒ Small ☒ Mediur ☒ Large

□ Animat

Step time 0

Figure 1. Time-domain signal processing.

The demonstration shown in figure 1 concentrates on the time-domain measurements of average zero-crossing rate, energy and average magnitude. This application enables to student to become familiar with basic digital signal processing in this domain. This demonstration also allows the student to both revise their understanding of the compute-and-shift nature of the windowing process and to see how various windowing types and shifts influence the processing techniques. An animation is included which is designed to reinforce the concept that each point on the feature plots is produced from a windowed segment of the signal.

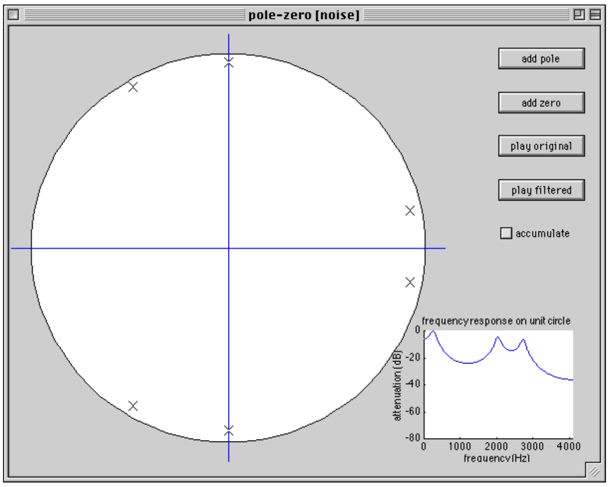


signal: tone

☒ accumulate

Figure 2. Animation still from the basilar membrane response to two tones: 700Hz and 1500Hz.

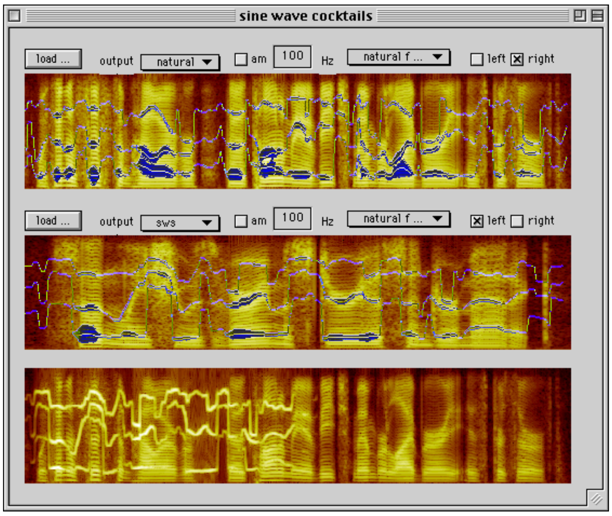
Figure 2 shows a demonstration which displays the response of a basilar membrane (BM) simulation to tones, impulses and noise (and combinations thereof). When the simulation is initiated, the upper trace (representing the basilar membrane) is animated. At the same time, a cursor moves along the lower signal window to indicate the current input sample being processed. The maximum BM displacement is also displayed, allowing students to investigate both place and timing theories of pitch perception.



□ accumulate

Figure 3. Linear systems design via a pole-zero diagram. The magnitude response of the system is displayed in the bottom right panel.

The tool shown in figure 3 allows users to create linear systems via a pole-zero diagram. The magnitude response of the linear system is displayed and updated as the user drags individual poles or zeroes. The linear system can be applied to signals and the filtered and unfiltered signals can be played back. Tones, uniform noise and impulse trains can be generated. In addition, preset pole-zero diagrams (in fact, all-pole systems) for vowels can be loaded.

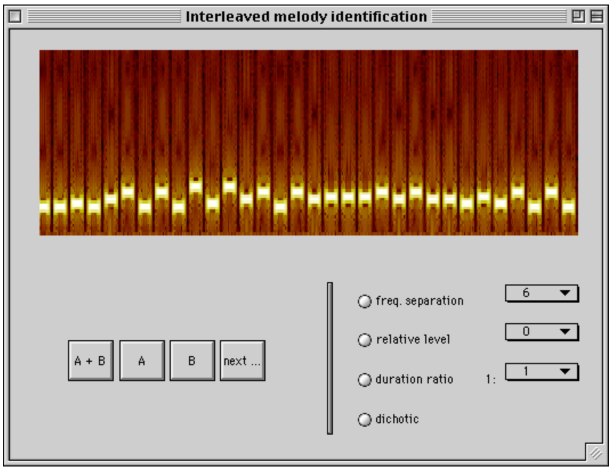


output natural □ am 100 Hz natural f ... □ left ☒ right

output sws □ am 100 Hz natural f ... ☒ left □ right

Figure 4. Sine wave speech mixtures.

Sine-wave speech is a synthetic analogue of natural speech produced by a small number of time-varying sinusoids. The present demonstration was motivated primarily by studies into the perception of simultaneous sine-wave speech utterances. In these experiments, listeners were asked to transcribe pairs of sine-wave sentences presented simultaneously. The demonstration allows a wide range of preferences to be set, including which signal is to be used in playback (sws, natural or silent) and the depth of amplitude modulation applied.



○ freq. separation 6

○ relative level 0

○ duration ratio 1: 1

○ dichotic

Figure 5. Interleaved melody identification.

Interleaved melodies have been used to illustrate the phenomenon of auditory scene analysis. In conditions in which the notes of one melody are acoustically distinguished from the other (e.g. by having different intensity or timbre) the two melodies become easier to identify. This demonstration has a dozen melodies preloaded into the tool, allowing the melody combination to be determined by the student.

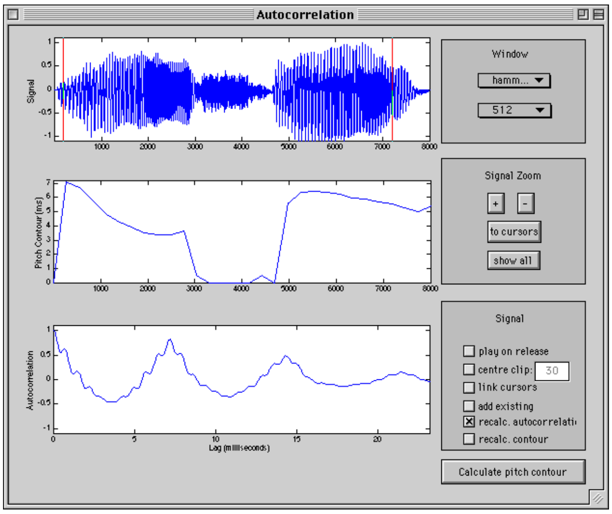


Figure 6. Autocorrelation-based pitch detection.

One method for deciding if a signal is periodic and then estimating its period is the autocorrelation function. The autocorrelation demonstration is shown in figure 6. The signal is displayed in the upper axes. Moving the cursors in the signal axis updates the autocorrelation plot displayed in the lower axes. As with a number of other demonstrations, the windowing details - type, size and shift - can be altered to see the change in the autocorrelation output. In addition to this, a pitch period contour of the entire signal can be calculated and displayed in the middle axes.

6. Discussion

Once a change in practice has occurred, a period of evaluation should follow. Currently, we have only collected informal feedback from students, and this has been encouraging. A formal evaluation might involve statistical comparison of examination and coursework marks with those of previous cohorts.

Although MATLAB is a highly portable language it is still necessary to test the demonstrations on all platforms as there are inevitable platform-dependencies. The most significant of these is variation in the graphical user interface on different platforms. However, it should be noted that Java exhibits similar problems [8].

We had anticipated that the development time of the demonstrations would be quite high but in fact each tool only requires two to three days effort. We see the development of the MAD package as an ongoing concern to produce tools for the teaching of both introductory level and advanced level speech and hearing concepts. In addition, we hope that inclusion of demonstrations on recent research will increase the exposure of leading-edge developments in the field.

The MAD package (and associated documentation) is available via http://www.dcs.shef.ac.uk/~martin. Contributions from individuals or organisations are welcomed.

Acknowledgements

In addition to those produced by the authors, Dan Ellis contributed to the collection. Stuart Cunningham and Ljubomir Josifovski helped with the testing. Funding for some of the development work was provided by the ELSNET LE Training Showcase, 98/02. An earlier status report on the MAD package has also been published by Cooke and Brown [9].

References

[1] Bloothooft, G, van Dommelen, W, Espain, C, Hazan, V, Huckvale, M and Wigforss, E (1998), *The landscape of future education in speech communication sciences*, Utrecht Institute of Linguistics Publications, Utrecht, 1998. ISBN 90-5434-069-X

[2] McGurk, H and McDonald, J (1976), Hearing lips and seeing voices, *Nature*, 264: 746-748.

[3] Houtsma, AJM, Rossing, TD and Wagenaars, WM (1987), *Auditory Demonstrations Compact Disc*, Available through the Acoustical Society of America.

[4] Bregman, AS and Ahad, P (1995), *Demonstrations of auditory scene analysis: the perceptual organisation of sound*, MIT Press, Cambridge, Massachusetts.

[5] See http://svr-www.eng.cam.ac.uk/comp.speech/ Section1/Labs/ogi.html.

[6] Bregman, AS (1990), *Auditory Scene Analysis*, MIT Press, Cambridge, Massachusetts.

[7] Taylor, F, Mellott, JD and Lewis, M (1996) SPECtra - a hands-on DSP learning experience. *Proceedings of ICASSP-96*, 1125-1128.

[8] Schaffer, J, Hamaker, J and Picone, J (1996), Visualisation of signal processing concepts, *Proceedings of ICASSP-98*, 1853-1856.

[9] Cooke, M and Brown, GJ (in press), Interactive explorations in speech and hearing, *Journal of the Acoustical Society of Japan*.