DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

The 1995 Mississippi State University Conference on

Digital Signal Processing

What:	EE 4773/6773 Project Presentations
Where:	Simrall Auditorium, Mississippi State University
When:	December 4, 1995 — 1:00 to 4:00 PM

SUMMARY

The Department of Electrical and Computer Engineering invites you to attend a mini-conference on Digital Signal Processing, being given by students in EE 6773 — Introduction to Digital Signal Processing. Papers will be presented on a wide range of topics including speech and image processing, parallel processing, and acoustic echo cancellation.

Students will present their semester-long projects at this conference. Each group will give a 10 minute presentation, followed by 5 minutes of discussion. After the talks, each group will be available for a live-input real-time demonstration of their project. These projects account for 50% of their course grade, so critical evaluations of the projects are welcome.

Session Overview

1:00 PM — 1:10 PM:	J. Picone, Introduction
1:15 PM — 1:30 PM:	**N. Doss**[1] and T. McMahon, "An Integrated Khoros and MPI System for the Development of Portable Parallel DSP Applications"
1:30 PM — 1:45 PM:	Y. Chen, L. Wang, A.M. Yusuf, and **H. Zhang**, "Noise Reduction in Laser Induced Breakdown Spectroscopy"
1:45 PM — 2:00 PM:	**X. Du**, W. Couvillion Jr., and M.H. Kiu, "Correction of Scan-Line Shifts of Digitized Video Images"
2:00 PM — 2:15 PM:	J. Beard, **S. Given**, and B.Y. Young, "DTMF Detection Using Goertzel's Algorithm"
2:15 PM — 2:30 PM:	C.R. Jones, R. Seelam, M.E. Weber, **S. Wilson**, "Physical Modeling of a String Instrument"
2:30 PM — 2:45 PM:	K. Bush, A. Ganapath, **J. Trimble**, and L. Webster, "Real-Time Speech Endpoint Detection"
2:45 PM — 3:00 PM:	V. Allen, **M.E. Henderson**, E.S. Wheeler, and J. Williams, "Active Noise Cancellation in a Highly Reverberant Chamber"
3:00 PM — 4:00 PM:	Demonstrations in 414 and 434 Simrall

1. The person presenting the paper is shown in bold lettering.

AUTHOR INDEX

Volume I

Digital Signal Processing

Table of Contents

1. The person presenting the paper is shown in italic lettering.

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

proposal for

An Integrated Khoros and MPI System for the Development of Portable Parallel DSP Applications

submitted to fulfill the semester project requirement for

EE 4773/6773: Digital Signal Processing

September 1, 1995

submitted to:

Dr. Joseph Picone

Department of Electrical and Computer Engineering
413 Simrall, Hardy Rd.
Mississippi State University
Box 9571
MS State, MS 39762

submitted by:

Nathan Doss, Thom McMahon, Anthony Skjellum

Parallel Digital Signal Processing Group
NSF Engineering Research Center and
Department of Computer Science
Mississippi State University
Box 6176
Mississippi State, Mississippi 39762
Tel: 601-325-2565
Fax: 601-325-7692
email: {doss,thom,tony}@erc.msstate.edu

I. ABSTRACT

This project combines two public-domain paradigms to create a parallel software environment for DSP programming. MPI (Message-Passing Interface), a message passing system, is an evolving standard for parallel computing. Khoros is an integrated software environment for DSP. This purpose of this project is to describe and demonstrate a software design that exploits Khoros and MPI parallel libraries for the deployment of parallel DSP. The resulting system enables parallel DSP using the Khoros system for development and the MPI system for performance portability. Specifically, the new system provides MPI-based toolboxes containing data parallel modules and utilizes MPI as a means of communication between modules. Evaluation is based upon performance and portability characteristics of the integrated Khoros and MPI system.

II. INTRODUCTION

Khoros[4] is a powerful system, providing a visual programming language and a wide range of development tools for the development of DSP applications. Cantata [5], the visual programming environment, provides a data flow model for building applications from collections of modules organized into toolboxes. The application engineer builds a block-diagram representation of the application using icons that represent the various modules, visually linking the icons together in order to specify the communication flow of the design. Khoros provides several toolboxes containing hundreds of modules for algorithms pertaining to 2D/3D plotting, data manipulation, scientific visualization, geometry, matrix operations, image processing, as well as others. For example, the matrix toolbox contains modules that perform various matrix operations such as matrix addition and LU decomposition.

MPI (Message Passing Interface) [3] is a standard for message passing introduced by the MPI Forum (a collection of researchers from industry, academia, and national labs) in April 1994. MPI has its roots in many previous message passing systems and therefore provides most of the features found in those message passing systems such as point-to-point and collective operations. Two of the most important and relatively new aspects of MPI are its support for libraries and inter-group communication. These two features are very important for the development of parallel data flow applications as proposed in this paper. Using MPI, parallel libraries can be written independently of one another and independent of the architecture, then used together in a single application. Inter-group communication allows groups of processors that have been logically separated during the process of task subdivision to communicate with one another.

The purpose of this project is to integrate Khoros and MPICH [1] [2], a model implementation of MPI developed jointly by Argonne National Labs and Mississippi State University, merging the development features of Khoros with the portability and functionality of MPI in order to provide a platform for the development of portable parallel DSP applications. This will provide application engineers with a powerful and familiar environment for developing and prototyping parallel DSP applications with little input required from the application engineer in order to make the application parallel. Parallel libraries and inter-group communication supported by MPI allow us to achieve two forms of parallelism within the Khoros and MPI system:

- **data parallel modules** MPI is used to build a Khoros toolbox of data parallel operations and algorithms.

- **parallel flow between modules** Inter-group communication and other MPI communication techniques provide a model for parallel flow of data between modules.

III. PROJECT SUMMARY

The integration of MPI and Khoros into a development environment for portable parallel DSP applications consists of the following steps:

- Investigate the Khoros 2 system as it applies to DSP programming in order to recognize the areas that can be effectively and efficiently parallelized through integration with MPI.

- Develop a design for communication between one parallel modules using the inter-group communication techniques provided by MPI.

- Describe the high level design of parallel libraries in MPI that implements the basic Khoros building blocks that have been recognized as lending themselves well to parallelism. This design influences and is influenced by the communication design.

- Integrate the MPI programs back into Khoros and its visual system Cantata by implementing a Khoros toolbox of MPI-based parallel algorithms for use in DSP applications. This implementation derives directly from the high level parallel library designs.

- Define a "runtime" start-up mechanism that Cantata can use to with various parallel hardware.

- Describe the achievable performance and possible limitations of the integrated Khoros and MPI system. The achievable performance of the MPI and Khoros system is determined by building parallel DSP applications using the modules from the parallel MPI toolbox, then measuring the performance of these applications. Possible limitations might be communication bottlenecks and applications that are inherently sequential and thus do not lend themselves well to parallelism.

- In the algorithmic area, we will concentrate on apparently simple, but nonetheless crucial building blocks at two granularities of parallelism. (Algorithms to be considered are motivated by our interest in FFT's.)

First, as shown in Figure 1, we consider the simple, industry-standard technique of parallelizing 2D FFTs by solving embarrassingly parallel 1D FFTs, followed by a "corner turn" (a new block for Cantata/Khoros, and strictly message passing and data massaging code), followed by another 1D FFT. The structure of this algorithm involves principally inter-communication operations.

Second, we consider parallelizing fine-grain algorithm, or the individual FFTs themselves. Parallelization of this algorithm is known to be possible, but communication intensive. Comparison of parallelized 1D FFTs and non-parallelized 1D FFTs will be given.

It is important to note that these deliverables go beyond software architecture; nonetheless the requirements for representing these building blocks will have an affect on the overall software design process noted elsewhere in this proposal.

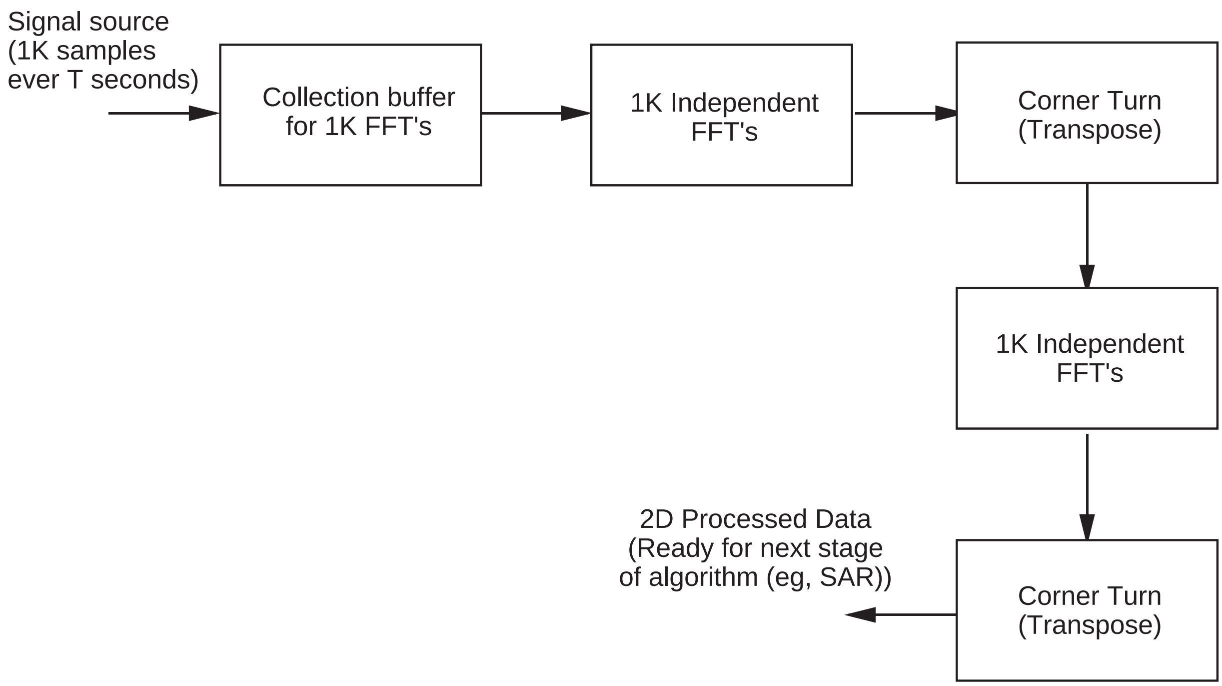


Figure 1Parallelization of FFT using Corner Turns

IV. EVALUATION

There are three aspects to evaluating the Khoros and MPI system. First is a demonstration that the system will work when the MPI-based data parallel modules are included in an application. A parallel module will be substituted for a sequential module in a well-tested existing application to demonstrate that the system will provide the same results as the original system and thus is working correctly. We will also demonstrate MPI communication between modules in the same application. The system will also be used to demonstrate applications that do and do not lend themselves to parallelism.

Since MPI is designed to be a portable message passing interface, we demonstrate that the applications developed using the integrated Khoros and MPI system execute on platforms supported by MPI. This demonstration will take advantage of the "runtime" start-up mechanism developed for Cantata.

One of the promises of parallelism is speed. In order to test the performance of the Khoros and MPI system, we take an existing Khoros application based on FFT and substitute the original modules with MPI-based toolbox modules where possible. We measure the performance of the original and modified applications to demonstrate the performance improvement available when an application takes advantage of parallelism.

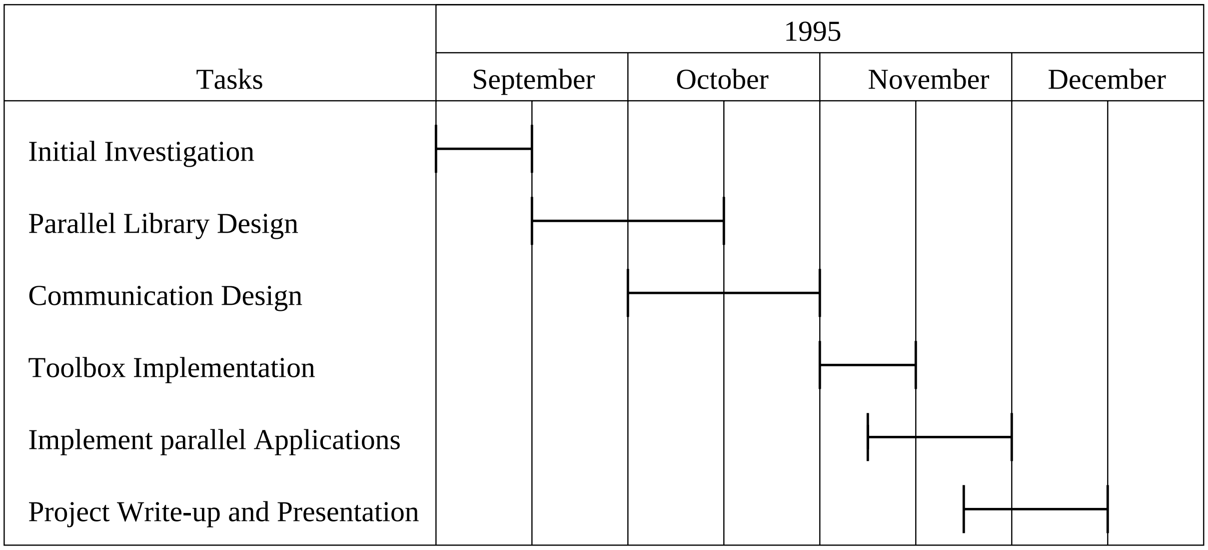


Figure 2 Development Schedule

V. SCHEDULE

Figure 2 illustrates the expected development schedule for this project.

VI. REFERENCES

[1] Patrick Bridges, Nathan Doss, William Gropp, Edward Karrels, Ewing Lusk, and Anthony Skjellum. Users guide to MPICH, a Portable Implementation of MPI, 1994. MSU/Argonne Joint Documentation.

[2] Nathan Doss, William Gropp, Ewing Lusk, and Anthony Skjellum. A Model Implementation of MPI. Technical Report MCS-P393-1193, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439, 1994.

[3] Message Passing Interface Forum. MPI: A Message-Passing Interface Standard. Technical Report Computer Science Department Technical Report CS-94-230, University of Tennessee, Knoxville, TN, May 5 1994.

[4] Rasure and Kubica. The Khoros Application Development Environment. In H.I. Christensen and J.L. Crowley, editors, Experimental Environments for Computer Vision and Image Processing. World Scientific, 1994.

[5] Young, Argiro, and Kubica. Cantata: Visual Programming Environment for the Khoros System. Computer Graphics, 29(2):22-24, May 1995.

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

proposal for

Adaptive Signal Processing

submitted to fulfill the semester the project requirement for

EE 6773: Digital Signal Processing

September 1, 1995

submitted to:

Dr. Joseph Picone

Department of Electrical and Computer Engineering
413 Simrall, Hardy Rd.
Mississippi State University
Box 9571
MS State, MS 39762

submitted by
Yuan Ping Chen, A M Yusuf, Liang Wang, Hansheng Zhang

Adaptive Signal Processing group
Department of Electrical and Computer Engineering
Mississippi State University
Box 9571
216 Simrall, Hardy Rd.
Mississippi State, Mississippi 39762
Tel: 601-325-3149
Fax-601-325-3149
email: ychen@erc, yusuf@erc, SPwang@borg, hz1@ra

I. ABSTRACT

Noise reduction of Laser Induced Breakdown Spectroscopy system (LIBS) is presented. The noise characteristics will be investigated and a proper digital filter will be chosen to reduce the noise. The result of this scheme will be compared with that of the analog method. The result will show an improvement of the ratio of the signal over noise and reduction of the data collection time. The improvement will enhance the capability of LIBS system to perform *in situ* measurement.

II. INTRODUCTION

Laser induced breakdown spectroscopy (LIBS) is a laser based, sensitive diagnostic technique used to detect certain atomic and molecular species in various environment [1]. A pulsed laser beam (532nm) is focused with a lens to the target, which can be gas, liquid, or solid to induce a micro-plasma in the focal area. The induced plasma is of very high temperature (for example 10,000 K). Any material in the plasma is excited and it produces strong optical emission. Spectroscopy analysis of the emission gives information about the properties of the material present in the laser induced plasma.

The emission from the plasma is collected and coupled to an optical fiber bundle with a series of lenses. The other end of the optical fiber bundle, which is set very close to the entrance slit of the spectrograph, transmits the emission signal to the spectrograph. The signal is dispersed with a grating and sent to a 1024 channel diodes array detector which is controlled by optical multichannel analyzer (OMA). The detector converts the optical signal into electric signal. The electric signal is processed analogically and finally converted to digital counts. The count values of 1024 channels form a spectrum.

The spectrum usually is very noisy. The noise comes from many different mechanisms, which include dark current of the detector, heat noise of the detector and the electric circuit, stray light in the spectrograph, the continuous radiation of the plasma, and the serious fluctuation in the emission intensity which is influenced by the plasma propagation in the form of shock wave. Sometimes, especially in field test, the influence of the environment such as site vibration and unexpected light also induces some noise. Reducing the noise is an important aspect in the instrumental development. The recent trend of diagnostic technique is *on site* analysis. Low noise and quick response are becoming more and more important.

III. PROJECT SUMMARY

The LIBS system converts the light signal to electric signal which is recorded by different channels, and different channels correspond to different frequencies. Through careful observation of the recorded signal, we found that it is narrow band signal which is embedded in broadband noise. The adaptive line enhancer is one of the most useful applications of adaptive filtering. It is a method of optimal filtering which can be applied to enhance noise corrupted signals. The adaptive line enhancer can suppress the noise component of a noisy signal, and pass the noise-free signal with little distortion [2]. It can enhance one or more narrow band signals of unknown and

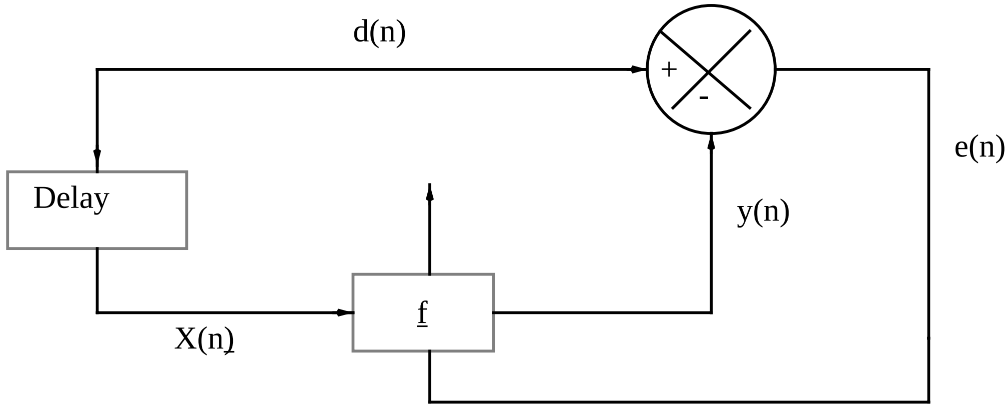


Figure 1: Adaptive Line Enhancer

possibly drifting amplitudes and frequencies which are embedded in broad band noise.

Figure 1 illustrates a block diagram of the adaptive line enhancer, where a delayed version of the input signal is used as an input to the adaptive filter whose coefficients are adapted to best fit the portion of the input signal which is uncorrelated with the noise, i.e. the noise-free signal portion of the input, by minimizing the mean squared error. The filter output is an enhanced version of the input.

The computational Algorithm for the ALE is [3]

$$\begin{aligned} y(n) &= \sum_{m=0}^{2} f_m(n)x(n-m) \\ x(n) &= d(n-\Delta) \\ e(n) &= d(n)-y(n) \\ f_m(n+1) &= f_m(n)+2\mu e(n)x(n-m) \end{aligned}$$

A direct implementation of the algorithm requires a prior knowledge of delay factor D, which is also known as the decorrelation parameter, the adaptation parameter u, and the filter length L. The above parameters should be chosen with careful consideration. The performance of the ALE in terms of stability and convergence depends on appropriate selections of these parameters.

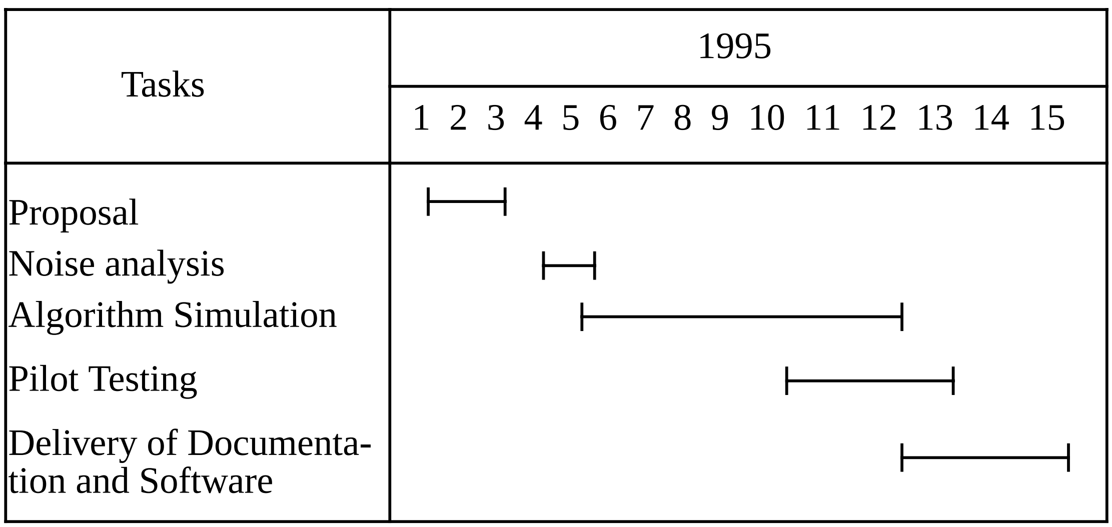
IV. EVALUATION

LIBS system will be used to measure three sets of samples: a stable sample, a typical sample that contains some type of dynamic component, and a simulation of truly time-varying sample. The noise-reduced spectra will be compared with the original spectra. The minimal mean square error will be estimated.

Additionally, for stable sample, the results of this noise reduction method will be compared with that of the analog method. A long time (several seconds) averaged (this analog method is used now) spectra will be used to record the spectrum as "standard." Several shorter time (less than one second) averaged spectra will be processed with this technique and be compared with the "standard" spectrum.

These comparison will give an overall evaluation of this DSP noise reduction method.

V. SCHEDULE



VI. REFERENCES

[1] Dennis F. Flangan, *Discrepancies between two formulations of signal-to-noise ratio for background-limited detection.* Applied Optics, Vol.34, No. 15, pp.2721-2723, May, 1995.

[2] B. Windrow and S.D. Stearns, *Adaptive signal processing*, Prentice-Hall, New Jersey, 1985

[3] P.M. Clarkson, *Optimal and Adaptive signal processing*, CRC Press, 1993.

proposal for

Correction of Scan-Line Shifts of Digitized Video Images

submitted to fulfill the semester project requirements for

EE 6773: Digital Signal Processing

September 1, 1995

submitted to

Dr. Joseph Picone

Department of Electrical and computer Engineering
413 Simrall, Hardy Road.
Mississippi State University
Box 9571
Mississippi State, MS 39762

submitted by

Xiao-song Du, Warren C. Couvillion Jr, Ming-Hoe Kiu

Department of Electrical and Computer Engineering
Engineering Research Center
Mississippi State University
Box 6176
Mississippi State, MS 39762
Tel: (601) 325-7097
Fax: (601) 325-7692
email: {billy,xsdu,kiu}@erc.msstate.edu

I. Abstract

The United States Forestry Service (USFS) has several images taken from a moving airplane using a conventional camcorder, and then digitized. Either from motion of the camcorder, the digitization process, or both, some of the scan-lines in the digital image became shifted several pixels.

Several algorithms exist for enhancing digitized images. These algorithms can be applied to test images distorted in a manner similar to the USFS images. The enhanced test images can be compared to the original test images to objectively measure the amount of distortion remaining. The best algorithm for correcting the test images can be applied to the distorted forestry images.

A good algorithm for enhancing aerial images recorded and digitized using inexpensive equipment would have widespread applications in the areas of forestry, agriculture, and crowd estimation.

II. Introduction

For our project, we will attempt to remove errors from still images created by "digitizing" video images from a camcorder on a moving airplane. In the images, some scan-lines are shifted by an indeterminate amount. We must determine the amount of shift in pixels, and hopefully a "real" distances. We also hope to determine where the error was introduced, for example, was the error produced by the motion of the camera, the digitization process, or a combination of the two.

The images we will be dealing with are aerial shots of forested areas. The United States Forestry Service wishes to use these images for their own mysterious purposes, perhaps counting trees or identifying marijuana plantations.

Since the images were taken videotaped using inexpensive, easily available video and digitizing equipment, a method of refining such images would have widespread applications. Reducing the price of quality aerial images would have many uses in agriculture, forestry, and rock-concert and/or political rally attendance estimation.

Attempts have been made to enhance the USFS images before [1], but with little progress. Consequently, the images collected by the Forestry Service have been gathering virtual dust for almost a year.

III. Project Summary

We will first clarify the problem. We will determine what information the Forestry Service wishes to derive from their images, and to what degree of accuracy. We will also gather as much information as we can about how the images were created, in hopes of identifying possible sources and types of errors introduced. This will also allow us to generate test images which we can distort in a manner similar to the USFS images. For example, if the airplane motion caused the camera to "jitter", a distorted test image could be created by ray-tracing, but shifting the eye-coordinates by slight, random amounts.

After we have sufficiently clarified the problem, we will perform a literature/Internet search to determine if anyone has solved this problem before, and if so, what were their methods and degree of success. Ideally, we will find publicly available "freeware" that corrects the errors introduced into the images. If no one has approached this particular problem before, we will determine what methods of image processing are applicable to the errors in the Forestry Service images.

Although the images are color, we will probably convert the color images to black and white [1], using a simple matrix multiplication [2]:

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

The "Y" value of each pixel corresponds to it "luminance" or brightness ("I" and "Q" values are not necessary for the black and white image). The "RGB" triple represents the red, green and blue intensity of each pixel for the color image.

One approach is to use edge detection to determine the amount of scan-line shift. If the image contains a straight edge, such as a building or farm border, or the video taping or digitization procedure left a "mark" at the beginning of each scan-line, we can use image edge detection algorithms to find the vertical edges in the image. For every pair of adjacent lines, the differences of the shift of a vertical edge intersecting the scan-lines must be determined. One of the scan-lines is shifted until the vertical edge intersections match up, and the process is repeated for subsequent line pairs.

Several methods exist for finding the amount of shift between scan-lines once the edges have been detected. The simplest is to assume that odd or even scan-lines are unshifted, and find the minimum sum of the absolute differences of the shifted lines, and then shift every other line by this amount. The process can be repeated until the shift is minimized. The assumption that half of the scan-lines are not shifted does not make this a very robust algorithm, but the USFS believes that the assumption is true for their images.

Another approach is to repeat the process described above, but using minimum squared error, rather than absolute distances as the amount to shift the scan-lines. Another similar approach that would work in areas containing an edge with a vertical component would be to treat the left- or right-most pixel locations of the edge as points falling near a line. The least squares approximation method could be used to determine the equation of the line, and then the difference between the line's values and the actual pixel values would be the amount of shift.

All the methods for detecting the amount of shift described above can be used for "autocorrelation" of the image, i.e., comparing the same scan-lines of adjacent frames, or "cross-correlation" of the image, comparing different scan-lines in the same frame. Auto- or cross-correlation functions using absolute minimum or least squared error for the amount of scan-line shift. These values are then used to interpolate the shifted scan-lines.

Most of the methods described above rely on the assumption that even scan-lines are unshifted, which may not be a good assumption. All of the methods rely on the assumption that differences in values of pixels of adjacent frames are caused by scan-line shifts. Unfortunately, other sources of error, such as motion blur, may be distorting the images.

All of these methods rely heavily upon the ability to detect vertical edges in the images. The following is a brief introduction to edge detection [3][4]:

1. Detection of discontinuities:

A common method of detecting discontinuities, such as points, lines and edges, is to use a mask matrix (usually a 3x3 matrix) and "slide" it along the image, calculating the sum of the products of the matrix coefficients with the image gray level. For example, the two matrices:

$$\underset{Image\ Gray\ Level}{\begin{bmatrix} I_1 & I_2 & I_3 \\ I_4 & I_5 & I_6 \\ I_7 & I_8 & I_9 \end{bmatrix}} \qquad \underset{Mask\ Matrix}{\begin{bmatrix} W_1 & W_2 & W_3 \\ W_4 & W_5 & W_6 \\ W_7 & W_8 & W_9 \end{bmatrix}}$$

are used to calculate the "response", "F" for the center point I_5. Response is given by:

$$\boldsymbol{F} = \sum_{i=1}^{9} \boldsymbol{I}_i \boldsymbol{W}_i \qquad (1)$$

The response to every pixel can be used to build a response image. We can use the response image to detect points, lines, or edges by choosing the appropriate mask matrix.

2. Line Detection:

For detecting horizontal, vertical, or 45^0 lines, these mask matrices can be used:

$$\underset{\dot{a}.\ Horizontal}{\begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix}} \qquad \underset{b.\ 45°}{\begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix}} \qquad \underset{c.\ Vertical}{\begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix}} \qquad \underset{d.\ -45°}{\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}}$$

Using equation (1) and the four matrices above, four response images, R_a, R_b, R_c, and R_d can be created For a pixel (i, j), if $|R_a(i,j)|$ has the maximum value of all the response images, that pixel may be located on a horizontal line. Similarly, if $|R_b(i,j)|$ has the maximum value, [i, j] may be located on a 45^0 line, and so on.

3. Edge Detection:

An edge is a boundary between two regions whose gray levels are obviously different. Most of edge detection algorithms involve gradients, because the gradient value indicates the rate of change of the gray value in a region. This change also may indicate a change in orientation of the image.

For edge detection, the gradient is given by:

$$\nabla f = [G_x, G_y] = \left[\frac{\Delta f}{\Delta x}, \frac{\Delta f}{\Delta y}\right]$$

where f represents the brightness values of the pixels, and G_x, and G_y represent the Sobel operators.

We use the Sobel operators to calculate the partial derivatives in digital form. The Sobel operators have the advantage of providing both a differencing and a smoothing effect. Because derivatives enhance noise, the smoothing effect is very useful. The equations for the Sobel operators are:

$$G_x=(I_7+2I_8+I_9) - (I_1+2I_2+I_3)$$
$$G_y=(I_3+2I_6+I_9) - (I_1+2I_4+I_7)$$

Or, stated another way, the mask matrices for the Sobel operators are:

$$\underset{G_x\ Matrix}{\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}} \qquad \underset{G_y\ Matrix}{\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}}$$

We can use these matrices and equation (1) to generate G_x and G_y response images. The G_x response image has a strong response along horizontal lines, while the G_y response has a strong response along vertical lines. The G_y response is very useful in horizontal shift detection.

4. Pel Recursion:

Similar to the method described above is pel (or pixel) recursion [4]. Since the images we have are subsequent frames of a video, the amount of shift occurring for each scan-line between adjacent frames can be computed. (One frame would contain the even scan-lines, and the subsequent frames would contain the odd scan-lines, i.e., the two frames are actually part of a single, interlaced image.) If f(i, j, n) is the brightness (or luminance) at pixel (i, j) at time n, then assuming that the pixel was shifted by a translation vector, **D**, then the brightness of corresponding pixel in the next frame is given by:

$$f(i, j, n+1)=f(i+d_x, j+d_y, n)$$

The frame difference, FD(i, j) is the differences in the brightness values between the current and previous frames, i. e.:

$$FD(i, j) = f(i, j, n) - f(i, j, n-1)$$

which is approximately the difference between f(i, j, n) and $f(i+d_x, j+d_y, n)$. Thus:

$$FD(i, j) = f(i, j, n) - f(i+d_x, j+d_y, n)$$

This equation can be written in terms of the brightness gradient, and then in pixel-to-pixel and line-to-line difference:

$$\mathbf{D'}_i = \mathbf{D'}_{i-1} + \mathbf{U}_i$$

where:

i = stage of iteration,
$\mathbf{D'}_i$, $\mathbf{D'}_{i-1}$ = new and previous estimates,
$\mathbf{U}_i$ = adjustment.

The displaced frame difference (DFD) for step "i" is defined as:

$$DFD(i, j, \mathbf{D}_i) = f(i, j, n) - f(i+d_{xi-1}, j+d_{yi-1}, n)^2$$

If G is a positive constant controlling the rate of convergence of the displacement vector, **D**, then a (hopefully) converging series of displacement vectors can be computed using the formula:

$$\mathbf{D}_i = \mathbf{D}_{i-1} - G\, DFD(i, j, \mathbf{D'}_{i-1}) o f(i+d_{xi-1}, j+d_{yi-1}, n-1)$$

If the displacement vector for a pixel in a frame converges, it can be determined where that pixel is in the subsequent frame should be located, and thus the subsequent frame can be corrected.

IV. Evaluation

If the Forestry Service has accurate data gathered via other means, we will evaluate our success by comparing the data derived from the enhanced images to the accurate data, and how big an improvement it is from the un-enhanced images.

If no accurate data are available, we will take images and introduce error similar to that seen in the Forestry Service images. We will then enhance it, and compare the resulting images to the originals. Since we will have the original images available to us, we will be able to exactly measure the amount of scan-line shift remaining in the enhanced images.

Method for measuring the quality of the enhanced images begin with a pixel-by-pixel comparison of the brightness values of the original image and the enhanced images. Several types of error measures could then be made. The simplest would be simply calculating the absolute differences between the brightness values of the original and enhanced images, or:

$$Error = \sum |f(i, j) - f'(i, j)|$$

where f(i, j) is the brightness value of the original image at pixel (i, j), and f'(i, j) is the brightness value of the enhanced image.

An objective criteria more closely modeling the human eye would be the signal to noise ratio (SNR):

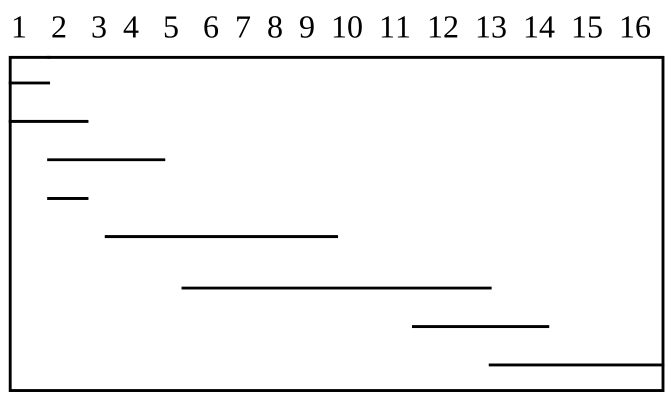
$$SNR = 10\log_{10}\left(\frac{\sum f(i,j)^2}{\sum |f(i,j) - f'(i,j)|^2}\right)$$

Unfortunately, no undistorted images of the USFS images to be enhanced exist (if they did, there would be no need for this project). One possible way to evaluate the quality of enhancement of the USFS images would be to derive some known value, e.g., forest acreage, from both the original and enhanced images. The differences between these derived values and the actual value would indicate the quality of enhancement. For the USFS images, if no data are available to quantify the error remaining in the enhanced images, we will rely on subjective criteria, i.e., how much better does it look to the viewer.

V. Schedule

Tasks **Week**

Proposal
Problem Clarification
Literature Search
SUN Raster to YIQ Programs
Test Images Software
Enhancement Software
Document and Report
Presentation



VI. References

1. S. C. Saxena, "Correcting Interfield Shifts in Interlaced Imagery," Unpublished report submitted to Dr. Robert Moorhead, Fall, 94.

2. J. D. Foley, A. v. Dam, S. K. Feiner, J. F. Hughes, *Computer Graphics: Principles and Practice, Second Edition in C*, Addison-Wesley Publishing Company, Inc., Reading, Massachusetts, USA, 1996.

3. R. J. Clarke, *Digital Compression of Still images and Video,* Harcourt Brace & Company, London, UK, 1995.

4. Gonzalez, Woods, *Digital Image Processing*, 1992.

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

proposal for

DTMF Detection Using Goertzel's Algorithm

submitted to fulfill the semester project requirement for

EE 4773/6773: Digital Signal Processing

September 1, 1995

submitted to:

Dr. Joseph Picone

Department of Electrical and Computer Engineering
413 Simrall, Hardy Rd.
Mississippi State University
Box 9571
MS State, MS 39762

submitted by:

Jimmy Beard, Steve Given, and Brian Young

DTMF Detection Group
Department of Electrical and Computer Engineering
Mississippi State University
Box 9571
216 Simrall, Hardy Rd.
Mississippi State, Mississippi 39762
Tel: 601-325-2220
email: {beard,given,byoung}@ee.msstate.edu

I. ABSTRACT

The goal of this project is to develop DSP software that will facilitate Dual Tone Multifrequency (DTMF) detection. The method that the group will use for DTMF detection is Goertzel's algorithm, which calculates Discrete Fourier Transforms (DFT's). The implementation of this algorithm will be written in C++. The testing of this algorithm will be done using a standard DTMF tape from MITEL and one from Bell Communications Research. Some testing may also be done using data taken by the DTMF group.

II. INTRODUCTION

In the telecommunications industry, DTMF detectors are becoming a topic of interest as telephone keypads are used to communicate with everything from stock brokers to credit card companies. As with most modern industries, DTMF detection is progressing from analog to digital processing.

Historically, DTMF detectors such as Western Electric Type A3 Touch-Tone receivers utilized analog circuits, including bandpass filters, to provide detection capabilities [1]. The analog technology is mature and costs for these systems are essentially flat. The rapid decline of the cost-to-performance ratio of digital systems now makes digital a viable option. There are in fact several vendors with DTMF chips on the market, each claiming to be better than the next. Digital DTMF detection is not an exact science yet. Looking at data sheets for existing technology reveals that there is room for improvement in many of the DTMF areas of concern.

Detecting DTMF signals in a noiseless environment is relatively simple. Goertzel's algorithm has been proven effective at this task. It performs a DFT and essentially one can pick out the frequencies of interest by setting a threshold for acceptable power levels. If the threshold is met or exceeded at a certain frequency, this frequency is accepted. Problems occur when random noise is inserted into the communication link. Noise in this case is anything other than DTMF signals (speech, white Gaussian noise, etc.). All of these things raise the noise floor thus causing the designer to be very selective about what level the threshold is set so as to avoid false detection. This only begins to describe some of the problems that this project will deal with.

Digital DTMF detection presents some very unique challenges. At a typical sampling frequency of 8000 Hz, the CPU load can get very high under real-time conditions. We will use Goertzel's algorithm to compute the Discrete Fourier Transform at 8 selected frequencies to determine the presence of the DTMF signals. This algorithm requires 2N multiplications and 4N additions for each frequency at which the DFT is to be computed [3], where N is a function of the sample frequency and the signal duration (# of samples). At a sample frequency of 8000 Hz, this requires each sample be processed for all eight frequencies in 125 microseconds or less. This requires substantial processing capability.

Goertzel's algorithm also requires a certain amount of memory. The program by Burris [5] requires four N-element floating point arrays. With the value of N in the range 200-500, the memory requirements get substantial. Obviously, real-time systems don't have the luxury of reading in all of its input values into an array before processing any of them, but the memory requirement of the detector will nonetheless be a major concern.

III. PROJECT SUMMARY

Detection of DTMF signals requires decomposition of the signal into its spectral components and then analyzing this spectral composition to determine if the frequencies of interest are present at the appropriate power levels. Decomposition of the input signal will require the calculation of the Discrete Fourier Transform of that signal at the eight DTMF frequencies, which will be accomplished using Goertzel's Algorithm [2,3,5]. Power levels for each of the signals must be determined to fall within the specified range in order for them to be accepted. Detection of the DTMF signals must also occur at or above the minimum required duration of the signal for acceptance.

The input data will be acquired from three sources: 1) computer generated sample signals, 2) MITEL Standard Test Tapes, and 3) BellCore's Series 1 Digit Simulation Test Tapes for DTMF receivers. Computer programs will be written to generate signals of the form

$$y(N) = A_1 \cos\left(2\pi f_1 \frac{N}{f_s} + \Theta_1\right) + A_2 \cos\left(2\pi f_2 \frac{N}{f_s} + \Theta_2\right) \quad (1)$$

This data will be represented as 16-bit numbers and stored in an ASCII file, where they will later be used as input to our detector for simulation and test purposes. Secondly, we will use the MITEL Standard Test Tapes for DTMF receivers as the primary data source for our detector. This will require converting the signals recorded on these tapes into a digital format and storing the results on disk. Third, the BellCore Series 1 Digit Simulation Test Tapes, which contain six half-hour sequences of speech samples that are known to contain energy at or near valid DTMF frequency pairs [1], will be used as a standard test source for measuring the accuracy of the detection capabilities of the software.

IV. EVALUATION

The primary goal of this project is to design a digital DTMF detector that will meet or exceed all BellCore requirements specified in TR-TSY-000181 [4]. This technical reference specifies performance requirements for the following parameters:

- **minimum accept level** - This is the lowest level at which a receiver recognizes signal tones. This level must be no greater than -36 dBm.
- **maximum accept level** - This is the highest level at which a receiver recognizes signal tones. This level must be no less than 0 dBm.
- **reject level** - This is the level at which signal tones are guaranteed not to be recognized. The reject level is lower than the minimum accept level but greater than -55 dBm.
- **twist** - Twist is the level difference between the high and low tone levels. It is said to be positive if the high tone is at a higher level than the low tone and is said to be negative if the opposite is true. The twist must be between -8 dB and 4 dB, assuming that other accept levels are met.

- **frequency accept bands -** The band at which signal tones are accepted must be plus or minus 1.5% of the nominal frequency, but must not be wider than the frequency reject band.

- **frequency reject bands** - The band at which signal tones are guaranteed to be rejected. It must be wider than the accept band but not more than 3.5% wider than the nominal frequency.

- **signal duration** - The signal must be present for at least 40 ms for acceptance and must be rejected for times less than 23 ms.

- **interdigit time** - The time in between tones must be at least 40 ms.

- **cycle time** - The time between the start of one pulse and the start of the next mst be at least 93 ms.

- **echo** - The DTMF detector must adhere to certain signal to echo ratios (S-E) at certain delays. The detector should tolerate 16 dB at 20 ms and 24 dB at 45 ms.

- **DTMF non-linear distortion** - This is the total power of extraneous frequencies that must be tolerated above the 500 Hz voiceband. This is expressed in signal to distortion ratio (S-D). The specifications are S-D of 20 dB relative to 2-tone power level and a S-D of 16 dB relative to any single tone power level.

- **presence of dial tone** - The detector must ignore the dial tone which is the sum of two sinusoids at 350 and 440 Hz with power of -10 dBm plus or minus 1.5 dBm.

- **message circuit noise** - The DTMF detector must tolerate signal to noise ratio (S-N) of 23 dB at a signal level 3dB above the minimum accept level. It must have an error rate no greater than 1 in 10,000.

- **impulse noise** - The detector must tolerate a S-N ratio of -12 dB at a signal level 3dB above the minimum accept level.

- **speech immunity** - The DTMF detector must not recognize any speech patterns.

- **adaptive sensitivity -** The receiver automatically adapts the threshold according to the power level of the first digit received and resets following each call.

We will evaluate the performance of the detector using the MITEL standard test tapes, BellCore standard test tapes, and data generated or collected by the group. The MITEL and BellCore tapes contain DTMF signals presented under a variety of circumstances simulating real-world conditions. There are specific performance requirements that must be met in order for the DTMF detector to meet BellCore standards and proper detection using these tapes should guarantee a good detector. Statistical performance data will be collected. Error rates will be calculated and compared against the BellCore requirements.

V. SCHEDULE

A time-line providing details about project emphasis and goals can be seen in Figure 1. The first thing that will be done is to submit a proposal for DTMF detection before the end of the fourth week of the semester. The proposal will outline all the goals of the project and the problems that the project is expected to solve. It will also be specific as to what constitutes a successful project. Once the proposal is approved work can begin on the actual project.

The first phase is to begin development of the I/O software. No code can be written until the group is sure that data is being input in a proper and logical manner. Once this is accomplished the algorithm can be coded and tested on the input data. The algorithm will undoubtedly need some modification and tweaking to elicit the proper response to the input data. The final part of this stage is the output. Good output is essential to ensure a good project. If the algorithm performs flawlessly but the output form the program is cryptic or unusable then the project is a loss. A good interface is essential.

A production demo will begin during algorithm testing to ensure that proper progress is being made on the project. Pilot testing will begin as soon as the algorithm is well on its way to proper implementation. This testing will involve providing the program with industry standard input and checking the output for proper adherence to BellCore requirements. The production demo will evolve as the program progresses, culminating in the finished product which will subsequently be covered in a complete report and presentation.

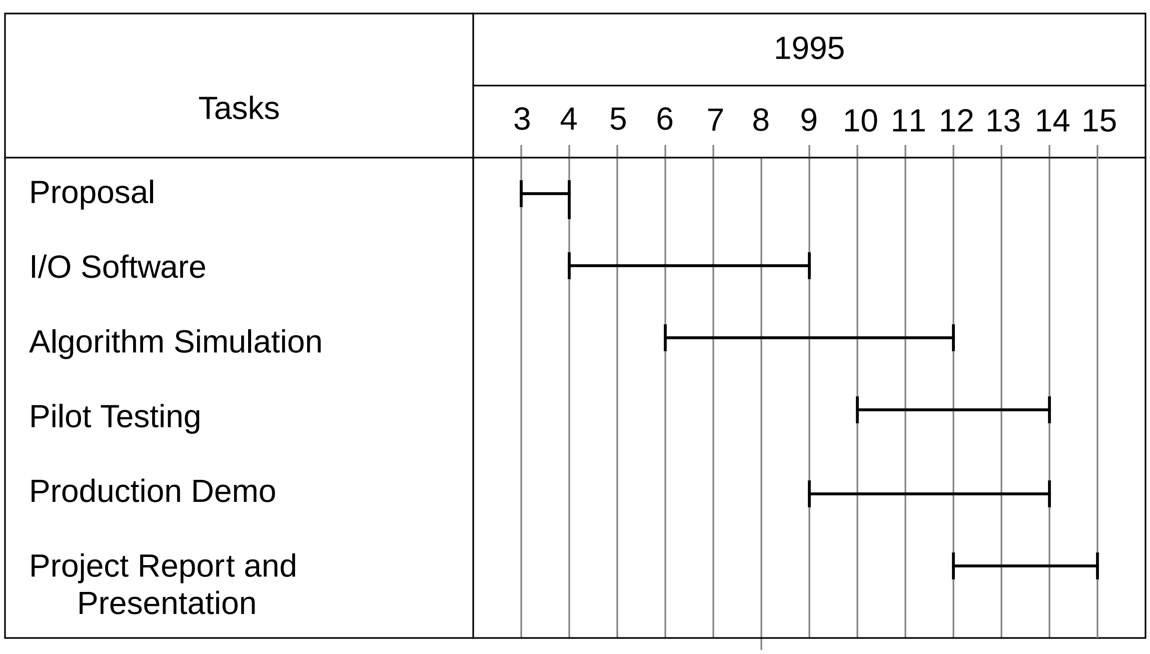


Figure 1. A time-line displaying the projected schedule for the Sun Sparcstation-based DTMF detector.

VI. REFERENCES

1. "Digital Simulation Test Tape." Bell Communication Research Technical Reference TR-TSY- 000763, Issue 1, July 1987.
2. Blahut, Richard E. *Fast Algorithms For Digital Signal Processing.* Addison-Wesley Publishing Co., Inc., 1985, pp. 131-2.
3. Manolakis, Dimitris G. and John G. Proakis. *Digital Signal Processing.* 2nd edition. Macmillan Publishing Co., 1992.
4. "Dual-Tone Multifrequency Receiver Generic Requirements for End-to-End Signaling Over Tandem-Switched Voice Links." Bell Communications Research Technical Reference TR-TSY-000181, Issue 1, March 1987.
5. Burrus, C. S. and T. W. Parks. DFT/FFT and Convolution Algorithms: Theory and Implementations. Texas Instruments, Inc., 1985, pp. 32-36.

proposal for

Physical Modeling of a String Instrument

submitted to fulfill the semester project requirement for

EE 4773/6773: Digital Signal Processing

September 18, 1995

submitted to:

Dr. Joseph Picone

Department of Electrical and Computer Engineering
413 Simrall, Hardy Rd.
Mississippi State University
Box 9571
MS State, MS 39762

submitted by:

Mary E. Weber, C. R. Jones, Scott Wilson, Raja Seelam

Digital Audio Processing Group
Department of Electrical and Computer Engineering
Mississippi State University
Box 9571
216 Simrall, Hardy Rd.
Mississippi State, Mississippi 39762
Tel: 601-325-3149
Fax: 601-325-3149
email: {seelam, weber}@isip.msstate.edu, {cjones, wilson}@ee.msstate.edu

I. ABSTRACT

The outcome of this project will be a high fidelity physical model of a stringed instrument using flexible digital signal processing algorithms, giving the artist complete control over the spectral content of the instrument.

A model will be developed for each functional part of the instrument including the string, the resonating cavity, and the transducer which transmits the energy of the string to the cavity. Live data will be captured and used to develop these models through analysis of harmonic content and dynamics.

II. INTRODUCTION

The sound of physics; to generate musical notes based on a physical model of a stringed instrument. To have at your finger tips not only a five-hundred-dollar student violin but also the sound of a two-million-dollar, 18th-century, masterpiece instrument. This is now possible with today's technology.

Music synthesis has always had the goal of creating a realistic sound of an instrument from a mathematical description of the way that instrument behaves. Through the years, an increased understanding of both the instruments and the technology has lead to an increased realism of the virtual instrument.

The first synthesizers developed consisted of racks of modular analog equipment with patch cables used to connect the different parts. As synthesizer technology became more compact, commercial models of portable analog synthesizers were made available. These synthesizers used a basic square, saw, or triangle waveform to emulate the sound of different instruments. A single filter was used to control the harmonic content of these waves, and a four point envelope generator was used to mimic the attack and decay of a particular instrument. This process is shown in Figure 1 and is known as subtractive synthesis.

Fourier analysis was used to develop additive synthesis. Instead of starting with too much harmonic content, the harmonic content is built from the ground up.

Digital systems were first used in synthesis to improve upon the stability of analog systems which tended to be unstable, and often need retuning. The oscillators and filters were digitally controlled, but remained in the analog domain. As computing power became more readily

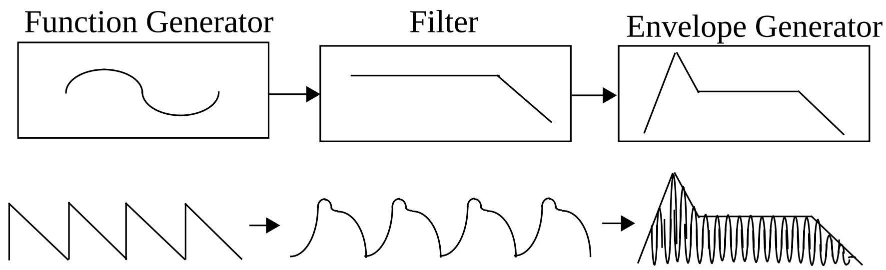


Figure 1: Subtractive Synthesis

available, The entire system became digital. With the move to software, new types of synthesis were developed easier. A couple of examples of unique digital synthesis methods include Yamaha's FM synthesis, Korg's vector synthesis, and Waldorf's wave sequencing.

The last major step in synthesizers was the move from pure additive or subtractive synthesis to that of sample playback synthesis. Instead of mimicking an instrument, a sample of the instrument being played is used as the output of the synthesizer. This produces a very realistic sound, but the quality is limited by the storage space available: the more samples of the instrument, the more convincing the sound. This method of synthesis is the most widespread today, ranging from toy-store keyboards to electronic grand pianos costing thousands of dollars.

Though these different systems of synthesis seem very different, they share a common thread: They are merely imitating the sound of a given instrument through various oscillators, filters, and envelope generators. As computing resources become more readily available, it is possible to model the physical behavior of the waves travelling through the instrument. With the technology available today, one could model the acoustic wave travelling through an instrument from and exciter, such as a bow, pick, or reed, through the instrument's resonating cavity, such as a bore or sound board, to the air and into the listener's ears. The model would optimally take into account the effects of the different parameters of the instrument such as type and age of the wood from which the instrument is constructed.

The scope of this project is to exercise control over the harmonic content of a virtual music instrument by modeling its physical properties.

III. PROJECT SUMMARY

The physical modeling of a musical instrument begins with breaking down the instrument into its functional parts. For a stringed instrument, this would include four parts: the exciter, the string, the resonating body, and the string to body transducer. The breakdown of the physical model for a stringed instrument is shown in Figure 1 This method of modeling the instrument provides extreme flexibility. Different models of each module can be developed and interchanged easily, giving the artist control over the specific harmonic content of the sound produced from the instrument.

Although the exciter is not a part of the instrument, it manifests itself in the model as the method by which the string becomes excited. This can be either a pluck, modeled by a single impulse with given energy, or a bow, an impulse train constantly exciting the string.

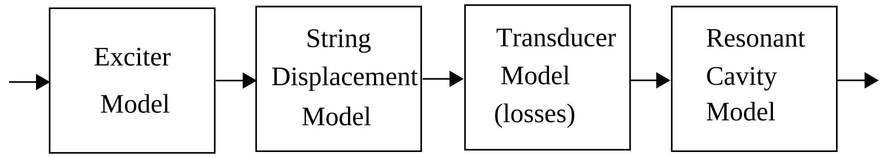


Figure 2: Breakdown of the Physical Model

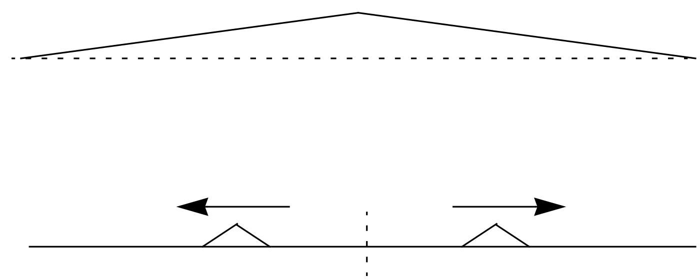


Figure 3: Behavior of a Plucked String

The model of the string has been solved in the past with brute force calculation of the specific solution to the partial differential equations associated with a tensile string. A more computationally efficient method is to use tap delay lines to simulate a general solution to these equations.

In a lossless case, this solution would be two waves traveling in opposite directions from the point of stimulus and the physical output at any point on the string is the summation to the two waves. This general behavior of a plucked string is shown in Figure 1. Due to the linear time invariant nature of the system, losses and dispersion can be lumped at discrete points on the tapped delay line. The frequency dependent losses and dispersion will be modeled by digital filters placed at a discrete point along the delay line. The modeling for the digital filtering is shown in Figure 1.

The resonating body of the instrument can be modeled in different methods as well. Another brute force model would be developed by analysis of the shape of the resonating cavity, analysis of the wood, and the properties in which acoustic waves travel through that material due to the grain, age, cracks, glue, varnish, and other parameters, and using these parameters to define the differential equations to solve.

The body model will be simplified in a similar manner in which the string model was simplified. Rather than using brute force, analysis of the instrument's frequency response and dynamic response will be used to create a delay line modeling a resonant filter equivalent to the resonant filter created by the body of the instrument.

The model of the various instruments will be accessible through a graphical user interface, and musical input will be through a manipulative picture of the instrument. More complex musical

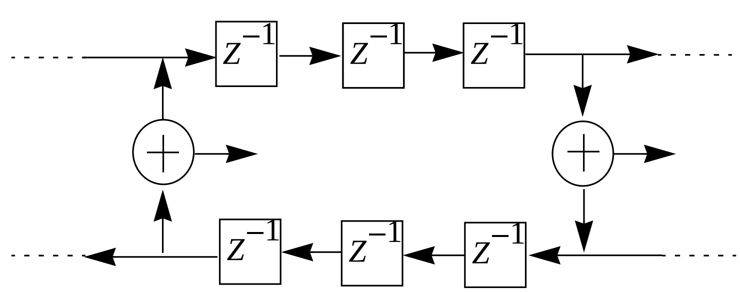


Figure 4: Model of Plucked String

data will also be available through a script file for prepared music, and through MIDI for real time performance.The MIDI interface will also allow for real time parameter changes in the model. Output will be available directly to digital outputs or to a file.

IV. EVALUATION

The wave equation for a flexible string between two rigid supports with tension K and mass density E is:

$$\frac{\partial^2 y}{\partial t^2} = \frac{K}{E} \cdot \frac{\partial^2 y}{\partial x^2}$$

Damping is a function of string velocity and, according to Morse (1976), when added to the above equation we have:

$$\frac{\partial^2 y}{\partial t^2} + R_m \frac{\partial y}{\partial t} = \frac{K}{E} \cdot \frac{\partial^2 y}{\partial x^2}$$

An explicit finite difference numerical solution to the above partial differential equation is given below:

$$y_m^{\ 0} = f(x, 0)$$

$$y_m^{\ 1} = \frac{1}{2 + kR_m}\left[2\left(1 - \frac{K}{E}P^2\right)U(m, 0) + P^2\frac{K}{E}(U(m+1, 0) + U(m-1, 0))\right] + g(0, t)$$

$$y_m^{\ n+1} = \frac{1}{2 + kR_m}\left[4\left(1 - \frac{K}{E}P^2\right)y_m^{\ n} + 2P^2\frac{K}{E}(y_{m+1}^{\ \ n} + y_{m-1}^{\ \ n}) + (kR_m^{\ -2})y_m^{\ n-1}\right]$$

The function $f(x, 0)$ describes the initial string shape and $g(0, t)$ describes the initial string velocity. The above numerical system will be used to generate displacement versus time data for a damped plucked or bowed string. This data inherently contains the frequency depended losses which are lacking in the "digital wave guide" methodology.

The displacement versus time model will be validated via comparison with actual recorded data from an acoustical violin and guitar. The electric violin and guitar provide data for viscosity damped strings which are not "loaded" by a resonant cavity which is consistent with the "displacement versus time" model.

The validated "displacement versus time" model will be the standard by which the "digital waveguide" model will be evaluated.

An interactive device created in TCL to provide the model in a real-time situation where the user can chose a note or a place on a string to be played. Where the user also has the choice of either a violin, an acoustic guitar, or a regular string.

V. SCHEDULE

A time-line for the project showing major milestones for the thirteen week project planned by which these will be achieved is shown in Figure 1. An initial version of the system should be available by the midpoint of the project.

VI. REFERENCES

6. Morse, P.M., *Vibration and Sound,* American Institute of Physics for the Acoustical Society of America, Woodbury, New York, USA, 1976.
7. Ohlson, Harry F. *Elements of Acoustical Engineering,* D. Van Nostrand Co., Inc., New York, New York, 1947
8. Reid, Christopher E. and Thomas B. Passin, *Signal Processing in C,* John Wiley & Sons, Inc., New York, 1992.
9. Rorabaugh, C. Britton, *Digital Filter Designer's Handbook: Featuring C Routines,* McGraw-Hill, Inc., Blue Ridge Summit, PA, 1993.
10. Smith III, Julius O., "Physical Modeling Using digital Waveguides," *Computer Music Journal,* vol. 16, No. 4, pp 74-91. Winter 1992.
11. Watkinson, John, *The Art of Digital Audio,* Focal Press, London, 1988.

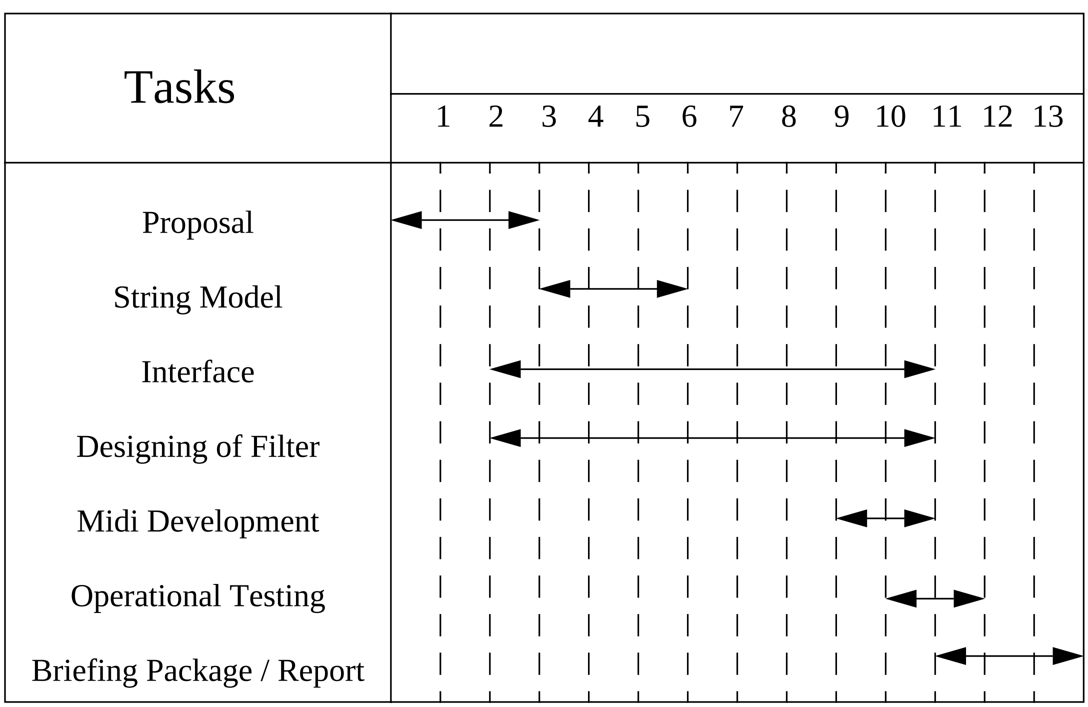


Figure 5: A Time-Line for the Thirteen Week Project

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

proposal for

Real--Time Speech Endpoint Detector

submitted to fulfill the semester project requirement for

EE 4773/6773: Digital Signal Processing

September 29,1995

submitted to:

Dr. Joseph Picone

Department of Electrical and Computer Engineering
413 Simrall, Hardy Rd.
Mississippi State University
Box 9571
MS State, MS 39762

submitted by:

Aravind, Kevin Bush, Paul Kornman, Jim Trimble, Leigh Webster

Speech Processing Group
Department of Electrical and Computer Engineering
Mississippi State University
Box 9571
216 Simrall, Hardy Rd.
Mississippi State, Mississippi 39762
Tel: 601-325-3149
Fax: 601-325-3149
email: {ganapath,kornman,trimble}@isip.msstate.edu
{SPbush,SPwebste}@ee.msstate.edu

I. ABSTRACT

This proposal describes a plan to design and implement a real-time endpoint detection software package using the C++ language. Accurate endpoint detection is vital for robust speech recognition systems. In fact, it has been shown that speech recognition performance is directly related to endpoint accuracy [1]. C++ virtual functions will be used to allow run-time modification of the detection algorithms. Real-time energy-based, zero crossing, least square periodicity estimate (LSPE), and spectral information detection methods will be implemented and their performance evaluated. Performance will be measured against hand-marked data and public-domain endpoint detection software using an objective scoring algorithm.

II. INTRODUCTION

One of the challenges in speech recognition is determining the beginning and end of each speech utterance. A particular problem in speech recognition is the high computational load on the system. Inaccurate endpoints can lead to wasteful computations.

In the past, several methods have been used for endpoint detection. Among these are zero crossing rate[3], energy distribution [2], spectral information, periodicity measures [4], and hidden Markov models (HMM)[5].

Through the use of phone models that are HMMs schooled on phonetically-balanced sentences, word pronunciations in a conversation become more distinct. The time-alignment of each conversation is obtained through a hierarchical-grammar speech recognition algorithm that utilizes corresponding conversation, word, and phone models. In this process, a word is characterized by its beginning time and duration. After each word is characterized, all words are combined to reproduce the original conversation which in essence produces a time-aligned record. Instead of using two single-channel signals, a combined-channel signal has proven to be more efficient in time and error prevention in the alignment process. The frequency of errors occurs more in the single-channel signal because of the empty portions of each signal when analyzing a conversation. Using the combined-channel signals to align the entire conversation requires an effective way of manipulating the simultaneous speech. For example, in a conversation between two people, words for the first speaker or the second speaker are aligned, but not for both. The recognizer decides which of the two paths will align better and selects that path. The drawback to this approach is that only data from one speaker is realized during simultaneous speech, but, simultaneous speech is normally brief whenever it occurs. This method does prove to be extremely apt in enabling the alignment procedure to decipher simultaneous speech.

Speech endpoint detection is trivial when used under ideal conditions; a simple energy calculation can be used. However, in noisy environments, the lowest level speech sounds barely exceed the background noise energy, and so energy level alone will not suffice to determine the correct speech endpoints. In many applications of speech recognition, such as speech recognition over telephone lines, the signal-to-noise ratio is relatively low thus requiring a robust endpoint detection algorithm.

In order to operate in real-time, speech detection algorithms must be efficient both in terms of the speed of computation and memory consumption. Also, the input signal must be buffered since the exact start and end of live input is unknown. Buffering allows the real-time endpointer to run in parallel with the live input.

III. PROJECT SUMMARY

This project will implement algorithms for endpoint detection in real-time. Software will be developed using the C++ language and virtual functions. A C++ class will be developed that will include a set of standard calls necessary for opening files, reading parameters, and detecting endpoints. This class, 'signal_detection,' will have sub-classes that are implementations of the different endpoint detection algorithms to be compared. Using virtual functions will allow the user to switch between algorithms without recompiling the code.The ISIP standard programming style will be adopted.

An extensive set of reference data from various sources both noisy and near-ideal will be collected and the endpoints will be hand marked. Because the goal of the reference data is to make the endpointer work, different types of reference data were selected. Speech data from the JEIDA database project will be used initially. Detecting endpoints on the JEIDA data should be relatively easy, because it was recorded under strict conditions. The next set of data will be live speech recorded in the ISIP demo room. This data should prove to be more difficult to endpoint than the JEIDA data, due to the poor acoustic characteristics of the demo room, utterance duration, uniqueness of the speaker's voice, and gaps between words and utterances. Finally, spontaneous speech telephone data will be used. Since telephone speech is inherently noisy, this data will provide the most rigorous test of the endpoint detecting algorithms.

The first endpoint-detecting algorithm that will be tested is an existing endpointer developed previously by ISIP. This code is based on energy and noise thresholds and does not operate in real time. Non-real-time algorithms have significant advantages over real-time algorithms, both in terms of complexity and accuracy.

Next, a real-time endpoint detection algorithm will be developed. This algorithm will essentially be a real-time version of the ISIP energy-based algorithm. Special data structures are necessary to store data in real-time code. This algorithm will employ a circular buffer for data input operations.

For the next algorithm, existing public domain code will be scored against the real-time algorithm. This code is a real-time endpoint algorithm written by Bruce Lowerre of Carnegie Mellon University. His C++ based endpointer has evolved from his twenty years experience working with live input speech signals. The algorithm runs in parallel with the input of utterances. It uses RMS energy calculations and zero-crossing counts to achieve the endpointing. Because the exact start and end of live input is unknown, the algorithm implements a circular buffer. The circular buffer with the real-time endpointer allows the starting silence before the first utterance to be thrown away. This buffer has to be large enough to store the maximum utterance length. As the start of the utterance is detected in real-time, the utterance is buffered until the end is found.

Next, an algorithm will be developed using least squares periodicity estimation (LSPE) technology. This method will apply a least-squares periodicity estimator to the input signal. It will trigger when a significant amount of periodicity is found. The input signal is determined to be speech whenever the periodicities exceed a fixed threshold.

Zero crossings will be used in the next algorithm. Zero crossing rate is a crude measure of the frequency content of speech. The average zero crossing rate provides a reasonable method to estimate the frequency on narrowband signals. Speech signals are classified as broadband signals. Due to this classification, the interpretation of the average zero crossing rate is much less accurate. Also, if the signal is "corrupted with interference" multiple zero crossings may occur; therefore, the signal must be "smoothed" or filtered to eliminate the multiple zero crossings. Zero crossings are known to work poorly with noisy data.

For the last algorithm, the spectral content of the data will be examined and searched for speech features.

Finally, a real-time demonstration of the endpoint detector will be given at the end of the semester. Speech data for this demonstration will be selected at the discretion of the instructor.

IV. EVALUATION

Evaluation and benchmarking is an integral part of any software development. One challenge in benchmarking endpoint detection algorithms is that there are no hard rules for determining endpoints. For this reason, the results of the hand-marked database will be a list of endpoint ranges, and not absolute endpoints. Another problem one faces while benchmarking endpoint algorithms is that the algorithm may not detect the exact number of utterances that are listed in the hand-marked database. This is known as the insertion/deletion problem. If the endpoints are compared sequentially, then one missed or added utterance will skew all following endpoints, and the error rate will be artificially large.

For these reasons, an objective scoring function will be developed. This function will determine and record the accuracy of the endpoints, both on binary and linear scales. It will also determine the best match of the calculated endpoints with the hand-marked endpoints to eliminate the insertion/deletion problem. The scoring function will score one file per speaker. The scoring function receives the range of the start point and the range of the end point of the speech utterance. Once received, the output from the endpoint detector (the start point and the end point) will be compared to the start range and the end range. If the value is in the desired range, a score of '1' will occur; otherwise, a score of '0' occurs.

The reference data recorded in the ISIP demo room involves several factors. First various speakers (3 male, 3 female) with unique voices will be used. Each speaker will speak the same utterances, words, or phrases. No two speakers vocalize the same utterance in the same fashion. the speakers also choose the order in which each word in a list is spoken. Each speaker will record seven types of data and store respective data in a respective file. An example of each file type is shown in Table 1.

Table 1

File #	Data Type	Sample Data
1	single number	"zero" "two"
2	large number	"eighteen" "sixteen"
3	two word number	"65" "44"
4	very large number	"300,188"
5	phrase	"high definition television"
6	sentence pattern	"we hold these truths to be self-evident, that all men are created equal"
7	spontaneous speech	********* (approx. 5 sentences)

The different algorithms will be compared against each other, and each algorithm's performance will be evaluated using the three aforementioned speech data types. Testing on the data collected from vastly different environments should give some measure of the robustness of the algorithms. The algorithms will then be rated based on their relative performance on each of the speech types.

V. SCHEDULE

The project was divided into several key sections to aid in the development of an effective endpoint detector. First, the reference data is to be generated and hand marked. Next, various algorithms are to be enhanced and/or developed. The first algorithm is the existing non real-time energy based algorithm and a real-time energy based algorithm (alg.00). The second algorithm is the zero crossing algorithm (alg.01). The public domain code by Bruce Lowerre (alg.02), LSPE algorithm (alg.03), and the spectral information algorithm (alg.04) comprise the final algorithms. A scoring function will be developed to compare the algorithms. In the midst of the above processes, a formal presentation and report will be prepared. Finally, a real-time demo will be made.

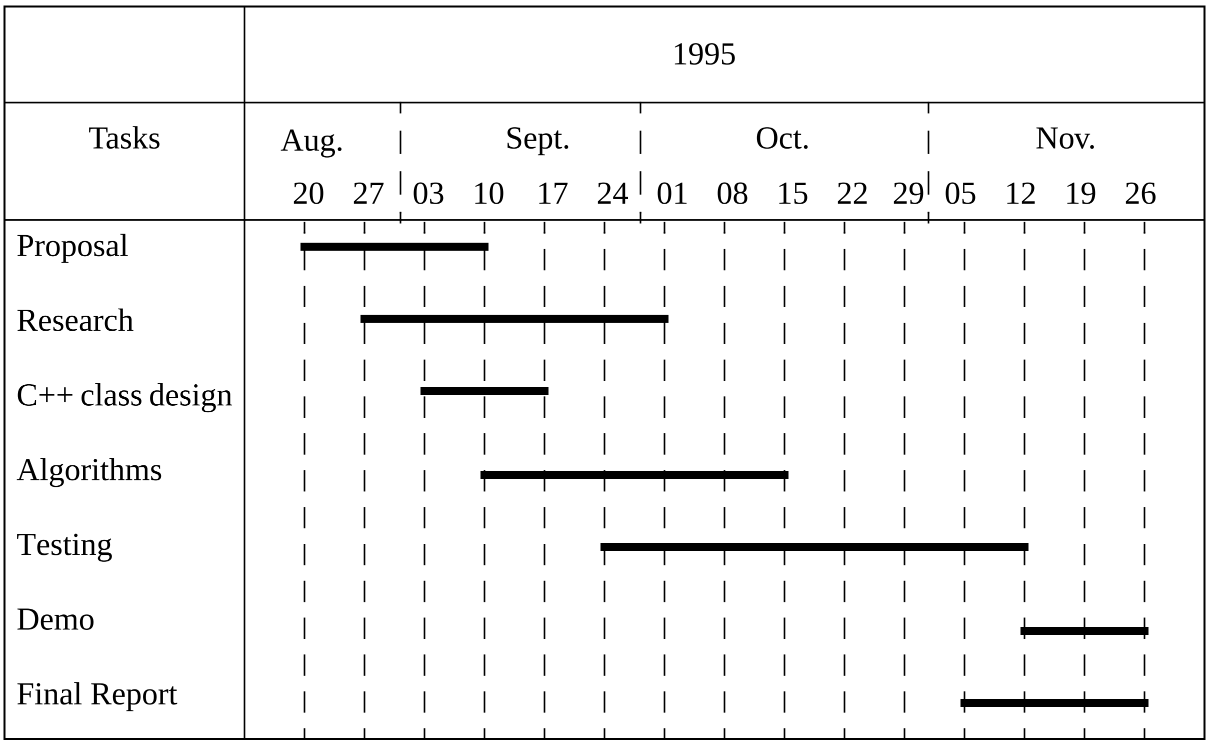


Figure 1. A time-line displaying the projected schedule for the Real-Time Speech Endpoint Detector Project.

VI. REFERENCES

1 T. Martin, "Applications of Limited Vocabulary Recognition Systems," *Rec. 1974 Symp. Speech Recognition*, pp. 55-71, 1975.

2 L. F. Lamel L. R. Rabiner, A. E. Rosenberg, and J. G. Wilpon, "An improved Endpoint Detector for Isolated Word Recognition," IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-29, pp. 777-785, Aug. 1981.

3 L.R. Rabiner and R.W. Schafer, Digital Processing of Speech Signals, pp. 127-130, 1978.

4 R. Tucker, "Voice Activity Detection Using a Periodicity Measure," IEE Proceedings, Part I, Communications, Speech and Vision, vol. 139, pp. 377-380, Aug. 1992.

5 B. Wheatley, G. Doddington, C. Hemphill, J. Godfrey, E. Holliman, J. McDaniel, and D. Fisher, "Robust Automatic Time Alignment of Orthographic Transcriptions with Unconstrained Speech," IEEE Proceedings ICASSP, vol. I, pp. 533-536, 1992.

proposal for

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Active Noise Cancellation of a Stairwell

submitted to fulfill the semester project requirement for

EE 4773/6773: Digital Signal Processing

September 1, 1995

submitted to:

Dr. Joseph Picone

Department of Electrical and Computer Engineering
413 Simrall, Hardy Rd.
Mississippi State University
Box 9571
MS State, MS 39762

submitted by:

Mark E Henderson, Erik S Wheeler, John Williams, Vincent Allen

Active Noise Cancellation Group aka (The Silent Echoes)
Department of Electrical and Computer Engineering
Mississippi State University
Box 9571
216 Simrall, Hardy Rd.
Mississippi State, Mississippi 39762
Tel: 601-325-3149
Fax: 601-325-3149
email: henders@isip, wheeler@isip, SPwillia@ee, SPallen@ee

I. ABSTRACT

In an increasingly noisy society methods of reducing noise are becoming more important. Noise, or unwanted sound, may be reduced in an environment by two basic means: passive noise control and active noise control (ANC). Active noise control is a method of reducing noise by canceling a sound wave with an inverted copy of itself. This process works best in a simple environment: one in which the wavelength of the noise is long in relation to the dimensions of the space. ANC has been most successful in reducing noise in ducts and headphones (essentially one dimensional problems). This project will center around applying ANC techniques to reducing the reverberation of low frequency noise in the stairwells of Simrall.

II. INTRODUCTION

In the Electrical Engineering building here at Mississippi State, namely Simrall, there are three large, four story stairwells. These stairwells are very reverberatory, and sustain a low frequency noise for long periods of time (~30 seconds). The reverberation is, in fact, so bothersome that conversations must be suspended while traversing the halls because the conversation quickly deteriorates. The frequencies amplified by the room are mostly at the lower end of the audio spectrum.

There are several ways that the reverberating action of a large space can be reduced. One option is to install baffles and other types of physical damping to the room. Using physical methods is most appropriate for high frequencies as the size and weight requirements of low frequency baffles tends to outweigh their usefulness. For example, an ideal location for global ANC is airplane cabins where the noise is very bothersome and in some instances medically harmful but the weight of the necessary baffles would severely restrict the capabilities of the airplane.

This project will focus on echo cancellation, starting with the elimination of simulated reverberation in the computer. This first stage will consist of the inputting of an artificial sound file with simple echo response, and canceling the echo through adaptive filtering techniques. Adaptive filtering examines an input, attempts to cancel it, and adjusts the filter coefficients to compensate for the error.

Upon completion of the simulated echo cancellation phase, the software will be modified to handle real signals recorded in the Simrall stairwell. These modifications will probably be minor, since the adaptive filter should be able to handle these more complex signals.

The canceller will probably not be tested in a real time experiment this semester due to the time constraints. But if it were, the reduction in noise would only be evident within a foot or two of the speaker, depending on the wavelength of the signal. This is due to the single input-single output filter we are designing. Since the signal to be canceled will not follow the same path as the canceling signal, the approximately 160 - 200 degree phase difference required for cancellation will only occur at particular locations in the room. It should be noted that, due to the varying phase difference, the signal will also be amplified in some areas. This problem could be controlled by the use of additional sensors and speakers.

The following information is a generic echo cancellation algorithm quoted from [5]. This publication describes a standard transversal filtering algorithm:

The reflected echo signal r(i) at time i can be written as the convolution of the far-end reference signal y(i) and the discrete representation h_k of the impulse response of the echo path between port C and D.

$$r(i) = \sum_{k=0}^{N-1} h_k y(i-k) \quad (1)$$

Linearity and a finite duration N of the echo-path response have been assumed. An echo canceller with N taps adapts the N coefficients a_k of its transversal filter to produce a replica of the echo r(i) defined as follows:

$$\hat{r}(i) = \sum_{k=0}^{N-1} a_k y(i-k) \quad (2)$$

Clearly, if $a_k = h_k$ for k=0, 1, ..., N-1, then $\hat{r}(i)$ = r(i) for all time i and the echo is cancelled exactly.

Since, in general, the echo-path impulse response h_k is unknown and may vary slowly with time, a closed-loop coefficient adaptation algorithm is required to minimize the average or mean-squared error (MSE) between the echo and its replica. It can be determined that the near-end error signal u(i) is comprised of the echo-path error r(i) - $\hat{r}(i)$ and the near-end speech signal x(i), which is uncorrelated with the far-end signal y(i). This gives the equation

$$E(u^2(i)) = E(x^2(i)) + E(e^2(i)) \quad (3)$$

where E denotes the expectation operator. The echo term $E(e^2(i))$ will be minimized when the left-hand side of (3!!!) is minimized. If there is no near-end speech (x(i) = 0), the minimum is achieved by adjusting the coefficients a_k along the direction of the negative gradient of $E(e^2(i))$ at each step with the update equation

$$a_k(i+1) = a_k(i) - \beta\frac{\partial E(e^2(i))}{\partial a_k(i)} \quad (4)$$

where β is the step size. Substituting (2) and (3) into (4) gives from (5) the update equation

$$a_k(i+1) = a_k(i) + 2\beta E[e(i)y(i-k)] \quad (5)$$

In practice, the expectation operator in the gradient term $2\beta E[e(i)y(i-k)]$ cannot be computed without a priori knowledge of the reference signal probability distribution. Common practice is to use an unbiased estimate of the gradient, which is based on time-averaged correlation error. Thus, replacing the expectation operator of (6) with a short-time average, gives

$$a_k(i+1) = a_k(i) + 2\beta\frac{1}{M}\sum_{m=0}^{M-1} e(i-m)y(i-m-k) \tag{6}$$

The special case of (7) for M=1 is frequently called the least-mean squared (LMS) algorithm or the stochastic gradient algorithm. Alternatively, the coefficients may be updated less frequently with a thinning ratio of up to M, as given in

$$a_k(i+M+1) = a_k(i) + 2\beta\sum_{m=0}^{M-1} e(i+M-m)y(i+M-m-k) \tag{7}$$

Computer simulations of this "block update" method show that it performs better than the standard LMS algorithm (i.e. M=1 case) with noise or speech signals[6]. Many cancellers today avoid multiplication for the correlation function in (8), and instead use the signs of e(i) and y(i-k) to compute the coefficient updates. However, this "sign algorithm" approximation results in approximately a 50% decrease in convergence rate and an increase in degradation of residual echo due to interfering near-end speech.

The convergence properties of the algorithm are largely determined by the stepsize parameter β and the power of the far-end signal y(i). In general, making β larger speeds the convergence, while a smaller β reduces the asymptotic cancellation error.

It has been shown that the convergence time constant is inversely proportional to the power of y(i), and that the algorithm will converge very slowly for low-power signals[7]. To remedy that situation, the loop gain is usually normalized by an estimate of that power, i.e.,

$$2\beta = 2\beta(i) = \frac{\beta_1}{P_y(i)} \tag{8}$$

where β_1 is a compromise value of the stepsize constant and $P_y(i)$ is an estimate of the average power of y(i) at time i.

$$P_y(i) = (L_y(i))^2 \tag{9}$$

where $L_y(i)$ is given by

$$L_y(i+1) = (1-\rho)L_y(i) + \rho|y(i)| \tag{10}$$

The estimate $\rho_y(i)$ is used since the calculation of the exact average power is computation-expensive.

III. PROJECT SUMMARY

The most obnoxious echoes in all of Mississippi occur in the rather spacious confines of the Simrall Electrical Engineering Building stairwells. The physical damping, by the room, of low frequencies is very small. In general the damping of the room could be increased through physical means, but to absorb the energy in the low frequencies it would be necessary to have very thick and heavy baffles. Another solution to the problem of reverberation is active noise control.

The goal of this project is to demonstrate the feasibility of a system capable of increasing the damping of low frequency sounds in the stairwells of Simrall. Such a system would require multiple sensors and multiple transducers to cancel reverberation over a significant space. The system we are proposing, which will cancel echoes only in a small region around a single transducer, as shown in Figure 1, could be expanded to cover a larger area with an increase in the number of microphones and speakers.

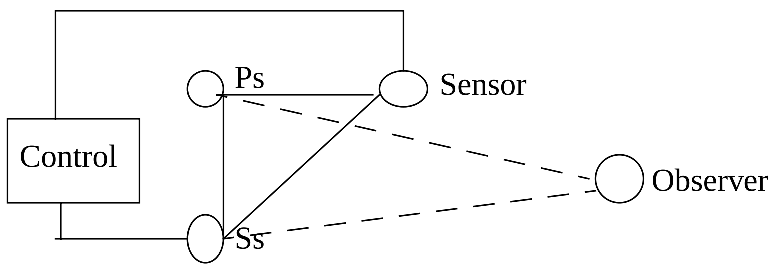


Figure 1. Block Diagram for Control System

The approach we will take will be to cancel the echoes using a technique common in telecommunication systems -- echo cancellation with adaptive filtering. The echo canceller works by reproducing the reference signal and applying it to a tapped delay transversal filter, as shown in Figure 2. In order to determine the transfer function, the canceller adapts the coefficients until a suitable modeling of the path is obtained. In the case of a reverberatory room in which the reference signal may be severely obscured by echoes, it may be necessary to "prime" the system with a series of short sinusoidal pulses. This should establish the proper coefficients for cancellation when the system is too saturated to distinguish the reference signal from the reverberation.

Assuming a physical implementation is feasible, the necessary control law, as derived from Figure 3, is shown below. In essence the echo canceller will model the path transfer functions (E(s) and G(s)) and will actively alter them so that the control goals are met, namely the reduction of echo energy. Equation (11) indicates the necessary control law for the physical echo canceller to work [4].

$$C(s) = \frac{G(s)}{M(s)T(s)[F(s)G(s) - E(s)H(s)]} \tag{11}$$

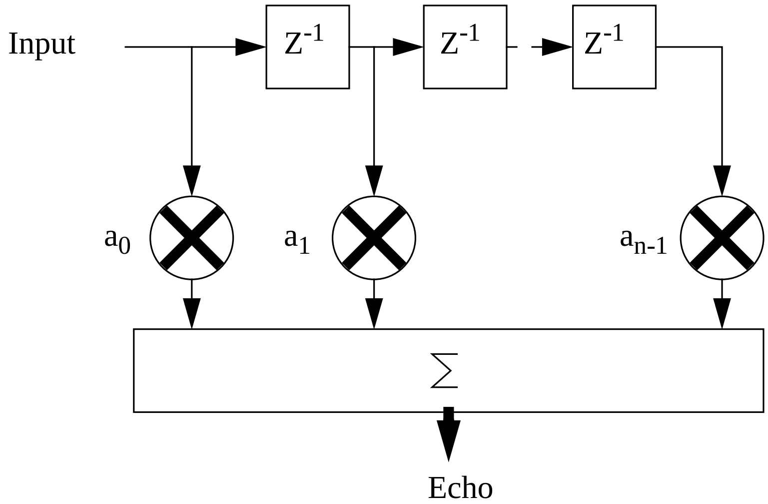
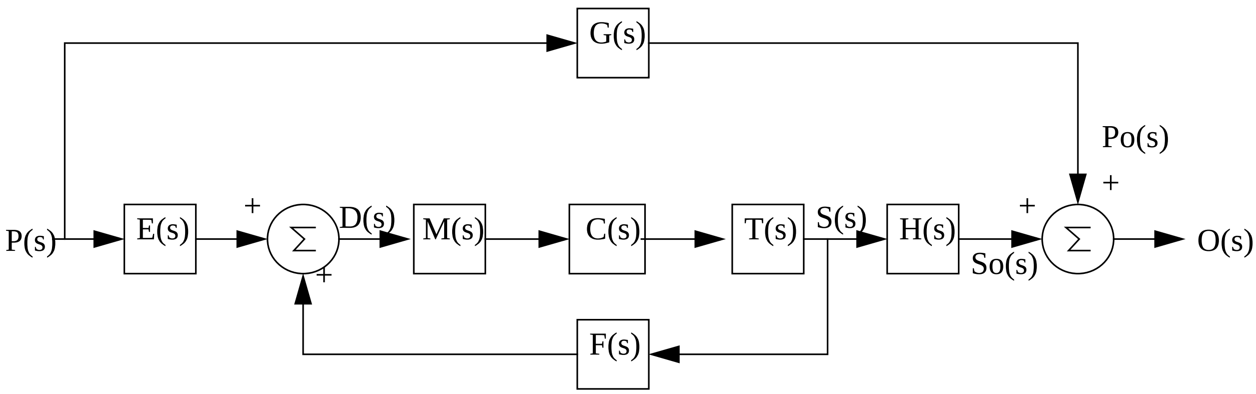


Figure 2. Diagram of an n-tap transversal filter as an echo modeler



From [4] :

P(S) is the primary source or noise source
O(s) is the observed sound
S(s) is the secondary source or control source
D(s) is the detected sound from both the primary and secondary sources
E(s) is the transfer function from the primary source to the detector
F(s) is the transfer function from the secondary source to the detector
G(s) is the transfer function from the primary source to the observer
H(s) is the transfer function from the secondary source to the observer
M(s) is the transfer function of the detector
C(s) is the transfer function of the controller
T(s) is the transfer function of the secondary source

Figure 3. Geometrical System Design

One early concern in the development of this filter is the number of taps required to effectively cope with the multiple reverberations of a single signal. At a sampling rate, f_s, the number of taps, n, required to eliminate a finite impulse response of t seconds is $n = t\ (f_s)$. For the system in question, which has an impulse response on the order of 30 seconds, the number of taps required at a sampling frequency of 8 kHz is 240,000 taps (which is more taps than even Fred Astaire could handle). This is the necessary number of taps to accurately model the system and its performance using the computer, but using the fact that if the first echo is cancelled then all other echoes are non-existent making a physical implementation possible. Hence the system will be concerned with only modeling the first echo.

The stability of the system is also a major concern. As discussed in the Appendix, there are several parameters to the system that will affect how quickly the system can respond and, in fact, whether or not the system will converge at all. Another parameter that will affect system stability is the output of the echo canceller. If the output of the canceller does not adequately attenuate the noise level it can set up a feedback loop that will quickly diverge.

IV. EVALUATION

The first step is to collect real data from the stairwell and small room adjacent to the stairwell in Simrall Electrical Engineering Building. The data will be collected using a digital audio tape recorder with a sampling frequency of 48 kHz. An analog tape player will be used to play a variety of generated data into the room. Test sounds will include a chirp (from 0 to 500 Hz in 30 seconds), a second chirp (from 0 to 20 kHz in 30 seconds), a 1000 Hz sine wave, a sum of three sine waves, and an impulse train at 100 Hz. This data will be used to evaluate the testing method and determine if other tests will be required to collect appropriate data. If such tests are deemed necessary a second trial will be run using new test sounds.

The second step will be to code the echo canceller as described in the appendix, and to test if the echo canceller can accurately predict echoes at a specific spot in the room. If the echo canceller can accurately predict the echoes then the input signal will be filtered using the generated echoes and the result should be a sound wave without the repeated echoes. This will be evaluated with a variety of test signals that are generated by summing a signal and multiple, time-delayed, attenuated copies of itself. If the echo canceller can accurately reduce the signal to its main component then this step is a success.

The third step is to optimize the code so that a physical system could be realized to run the code and generate the necessary outputs in real time. This will be evaluated using timing logs for the code after it is optimized and computing required space for the code (memory consumption).

V. SCHEDULE

Figure 4 shows a graphical example of our proposed time line of events. The first step will be to develop the model for the 1D nd 3D cases followed by a testing of the model versus the actual response of the room. Once the model is completed we will investigate a variety of control laws to determine which will work the best. Once the simulations prove that the design will work then the final step of actually trying out the steady state model will be implemented.

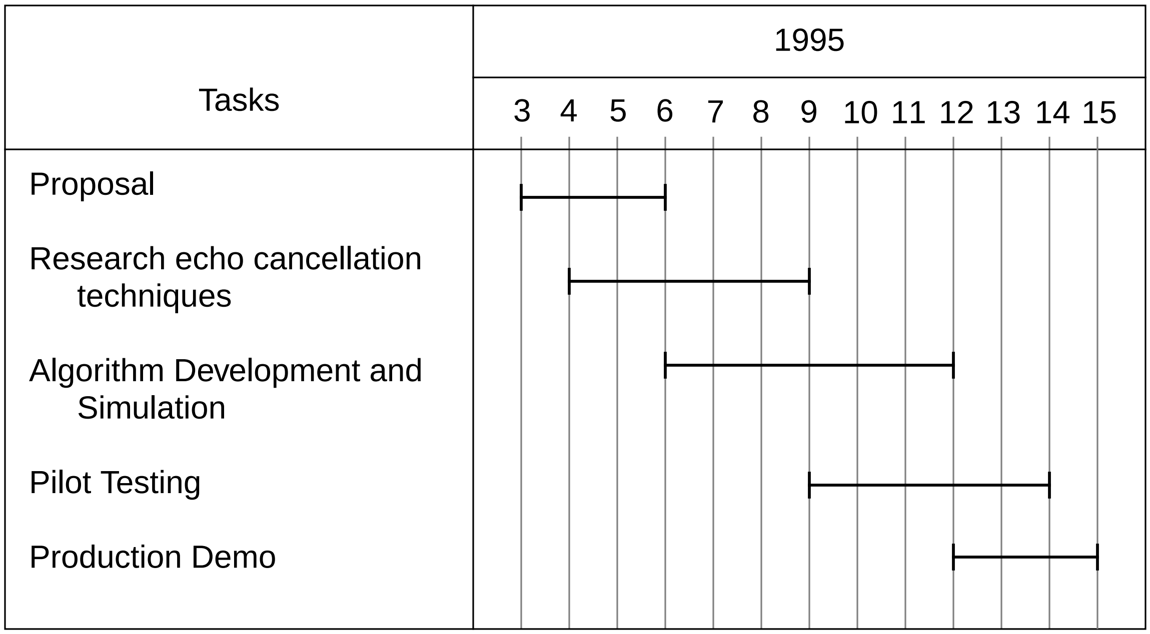


Figure 4. A time-line displaying the projected schedule for the Active Noise Cancellation of a Stairwell project. A one-semester project is planned in which an initial version of the system should be available by the end of the project.

VI. REFERENCES

1. Shieh, Chifong and Mingsian R. Bai, "Active Noise Cancellation by Using The Linear Quadratic Gaussian Independent Modal Space Control," *Acoustical Society of America.*, vol. number., pp. 2664-2671, May 1995.
2. Clark, Robert, "Active Damping on Enclosed Sound Fields Through Direct Rate Feedback Control," in *Acoustical Society of America*, vol. number. pp. 1710-16, March 1995.
3. Ruckman, C.E., "Active Noise Control FAQ," Available via anonymous ftp as from ftp://rtfm.mit.edu/pub/usenet/news.answers/active_noise_control_faq.,
4. Tokhi, M.O., R. R. Leitch, *Active Noise Control*. Oxford Science Publications, Clarendon Press 1992.
5. D. Messerschmitt, D. Hedberg, C. Cole, A. Haoui, and P. Winship, "Digital Voice Echo Canceller With a TMS32020," *Digital Signal Processing Applications With the TMS320 Family*, pp. 415-437, Texas Instruments, Inc., 1986.
6. M.J. Gingell, B.G. Hay, and L.D. Humphrey, "A Block Mode Update Echo Canceller Using Custom LSI," *GLOBECOM Conference Record*, Vol 3, 1394-97 (November 1983).
7. D.G. Messerschmitt, "Echo Cancellation in Speech and Data Transmission," *IEEE Journal on Selected Topics in Communications*, SAC-2, No. 2, 283-303 (March 1984).