Proceedings of the 6th IASTED International Conference
COMPUTER GRAPHICS AND IMAGING
August 13-15, 2003, Honolulu, Hawaii, USA

iSIGN: MAKING THE BENEFITS OF READING ALOUD ACCESSIBLE TO FAMILIES WITH DEAF CHILDREN

Tony Scarlatos
Computer Science Department
Stony Brook University, SUNY
Stony Brook, NY 11794
USA

Lori Scarlatos
Computer and Information Science
Brooklyn College, CUNY
Brooklyn, NY 11210
USA

Francesco Gallarotti
Computer Science Department
Stony Brook University, SUNY
Stony Brook, NY 11794
USA

ABSTRACT

In this paper, we describe an application that helps hearing-impaired students learn to read and form words by translating speech into video clips of American Sign Language (ASL). The narrator, either a parent or teacher, reads aloud to the student, and the application displays the ASL clip(s) along with the written word(s). The student is thus able to observe the speaker form the words and then see a sign they recognize, the written word, and an illustration of the meaning of the word. We have developed a prototype system along with several "albums" containing the supporting multimedia.

KEY WORDS
Multimedia education, speech recognition, digital video

1. INTRODUCTION

Communication within the hearing-impaired and deaf community has historically been done through American Sign Language (ASL). Yet in recent years emphasis has been placed on hearing-impaired students acquiring oral communication skills. Many schools, like the Cleary School for the Deaf, have adopted a bilingual strategy of providing instruction in ASL while teaching students to read and speak English.

The challenges are numerous. Deaf and hearing-impaired children learn to sign at an early age, and can receive academic instruction through ASL. But the pace of instruction is constrained by the reading level of the student. Even the best twelve-year-old students can only read at a first or second grade level. Although "mainstreaming" deaf students into public high schools is now the trend, it's difficult to "mainstream" a student whose reading skills are poor.

Hearing-impaired students cannot relate letter groupings to phonetics, and instead learn to read by recognizing written words as graphic shapes associated with a meaning (similar to learning to read Chinese). They learn to form words by observing a speaker's mouth and lip movement, as well as facial expressions. They have to simultaneously associate word shapes, lip movements, ASL gestures, and the meaning of the words.

Outside of the classroom, most parents and caregivers do not know ASL, and thus are limited in their ability to supplement instruction provided in school. Schools like the Cleary School for the Deaf provide ASL tutors who make "house calls" to give some instruction to parents, but their resources are limited. Most of the instruction is focused on "survival" ASL (i.e. "hungry" and "cold") and not on lesson plans.

We are developing iSign to address these issues. Our approach is simple and unique. Our application allows parents to communicate naturally (verbally) with their hearing-impaired child, teaching them to recognize and form words simultaneously. We believe our application empowers parents of deaf children by facilitating their participation in their child's education. We also believe that our approach will prove to increase the child's comprehension and retention, while increasing the pace of vocabulary acquisition. In essence, iSign makes the joy of reading aloud accessible to families of hearing-impaired children with all of its associated educational benefits.

2. THE NATURE OF THE PROBLEM AND RELATED WORK

Sign language in America was originally imported from France, and American Sign Language inherits its noun/adjective syntax from this lineage. Consequently, a straightforward word-by-word translation of a sentence in English to ASL will be syntactically incorrect. Furthermore, there is a fluidity of gesture in ASL that is difficult to render in a word-by-word translation. And there are many words in English (like "bagel") for which there is no equivalent ASL sign. These words must be finger spelled.

At the same time, speech recognition technology is still in its infancy. Most speech-to-text systems require extensive "training" of the software to the narrator's unique speech patterns for even modest recognition rates. Additional narrators have to go undergo the same lengthy training. And should the narrator's voice change (for example, if they have a cold), recognition rates drop. Furthermore, words with similar sounds like “bow” and “bough”, or “reed” and “read” can only be translated by the software as a “best guess” based on the context of the other translated words in the sentence.

With any translation software there is a need for a large database of words. Since the most straightforward representation of ASL terms is a video of an ASL signer, this introduces the issue of managing an enormous amount of static media.

Since much of the meaning in ASL is dependent on the expressiveness of the signer, the usefulness of a dynamic synthetic signer is diminished.

Within these constraints, several developers have attempted in recent years to translate English to sign language.

iCommunicator, developed by Interactive Solutions, Inc. [1] relies on Dragon NaturallySpeaking software for speech recognition to trigger the display of over 9000 videotaped signs. It attempts to translate anything the narrator says into a string of ASL video clips. But there are some drawbacks. There is the need for training, which can take several hours. The software only recognizes the voices it was trained for. Each word is translated to a discreet video clip (the signer returns to a neutral gesture after each word) and the clips are shown in the English syntax order. Even with a relatively good recognition rate of 80- 85%, these translations are more of a “gist” than a fluid translation. Finally the system has a high license cost of $4000, and relies heavily on proprietary and specialized hardware and software.

To solve the problem of fluid translation from English speech to ASL gestures the TESSA system [2], developed in England by the School of Information Systems at the University of East Anglia, bypasses digital video and combines speech recognition technology with a 3D “puppet” whose gestures are interpolated from one gesture to the next. The software was developed to enable Post Office assistants to communicate with deaf customers. The assistant speaks into a microphone and the speech is converted to British Sign Language and signed by the virtual signer for the customer on a screen mounted on the service counter. There are a limited number of responses the system can process, and the customer must submit the query or request in English, somewhat obviating the need for sign language interaction.

ASL Personal Communicator, developed at the Comm Tech Lab at Michigan State University [3], avoids speech recognition altogether, instead translating text strings input from the keyboard. The MSU application provides an ASL reference of 2500 terms, which can be accessed online for free. Words not in the database of video clips are finger spelled. The video clips are small and heavily compressed, and the application shares the problems of discontinuity and fluidity in translation. Still the MSU application works well as a reference for those interested in learning ASL.

For teaching deaf students to speak, a synthetic actor named Baldi was developed at the Perceptual Science Lab at the University of California in Santa Cruz [4]. Baldi can be viewed with transparent cheeks to reveal tongue movement as the words are formed. Responding to text input from the keyboard Baldi has been successful in field tests teaching deaf children to speak and recognize words. The system is distributed freely.

3. iSIGN

Because our approach is focused on deaf students learning to recognize individual words in a vocabulary lesson, we were able to optimize our system to listen for and translate only a small set of words, not the entire English dictionary. Our application therefore doesn’t have problems of syntax or discontinuity in translation, and our recognition rate is practically 100%.

We wanted the benefit of translating speech to assist deaf students in learning to lip-read as well as recognize written words. And we wanted a system that recognized any narrator with no software training. Our implementation on Apple’s OSX platform provides that functionality with near-perfect recognition rates.

Our application requires no special hardware or software other than the iSign application, a DVD of the ASL video clips, the OSX operating system, and the Macintosh built-in microphone.

3.1 The Application

We have developed a prototype system that responds to a limited set of related words, as are found in board books for young children. These board books typically focus on some theme such as farm animals, household objects, food, or parts of the body. Each page contains an illustration and a single word.

We have designed our application to be used by a parent or caregiver. This “reader” can choose from several available “albums”, which can be thought of as the equivalent of the board books, or vocabulary lessons. Each album contains a collection of related words and the corresponding visual media, which include the word shape, an illustration of the meaning of the word, and a video clip showing the ASL sign for each word.

After loading an album, the program in “listening” mode is ready to use the speech recognition engine to recognize the word spoken by the user. A list of the words in the album is displayed on the screen close to the main window. When the reader says a word, the child gets the first visual input by watching the reader's lips.

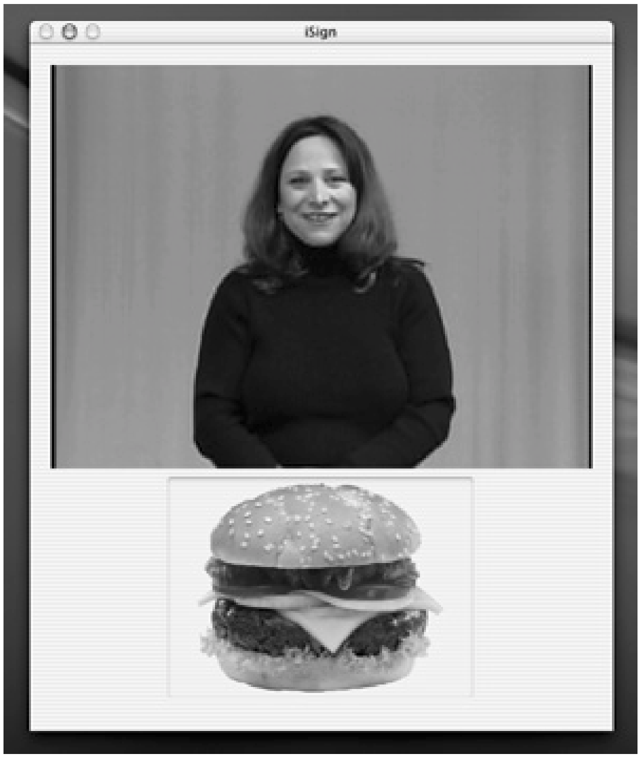

Figure 1. When a word is recognized, the corresponding ASL video clip is played.

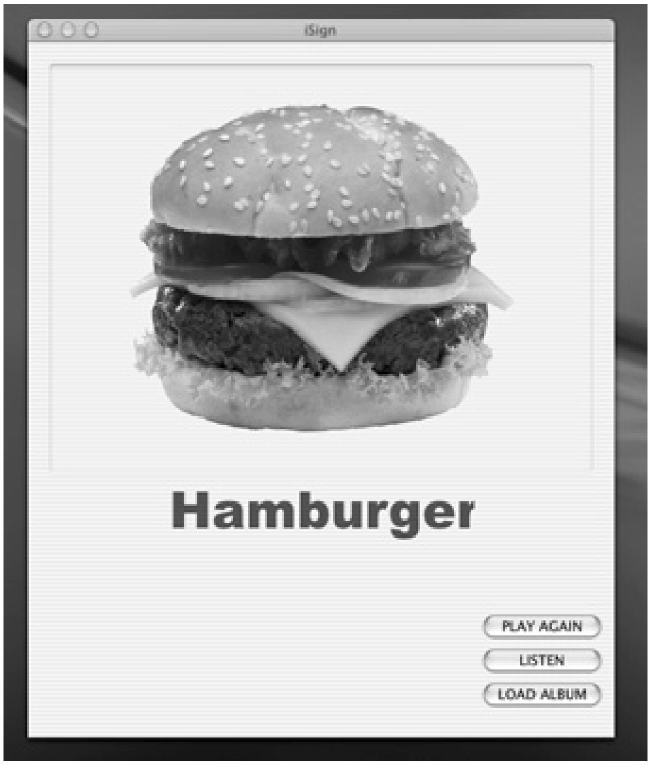

Figure 2. The illustration and word help the student to recognize the word shape.

After a short delay – which gives the child time to shift his or her focus to the computer screen – the ASL video clip of the word is displayed on the main window of the application (figure 1). This is the second visual input for the child: a well known ASL sign that will help him or her in the process of associating all these inputs to the final word shape. Immediately following the ASL video clip, a big illustration of the meaning of the word, together with the word shape itself, is displayed on the screen (figure 2). We decided to use very vivid pictures in order to allow the child to create a strong association between the image and the word shape. The word is displayed in mixed typeset (initial letter in uppercase and body of the word in lowercase). The upper/lower case presentation of the word shape is more distinctive than all capitals, making the word shape easier to recognize and remember.

3.2 The Implementation

We explored several different approaches in our efforts to produce a robust software system that is accurate yet equally sensitive to different voices.

We decided not to employ one of the popular speech recognition programs available on the market, like IBM's ViaVoice or MacSpeech's iListen. These programs require a long training period that must be performed in order to achieve even a minimal level of voice recognition. The resulting voice profile is then specific to a single narrator, and the training must be repeated for all other narrators. Complex installation, software licensing costs, and high system requirements were other downsides.

A more robust solution was to employ the built-in speech recognition engine in the Macintosh operating system, which is used to implement the "Speakable Items" (speech command) facility. This engine is capable of understanding any type of voice, with no training necessary. It also precludes the need for proprietary software. The only limitation is that the engine can only recognize words from a pre-defined set. This limitation does not affect our application directly. Our application defines the vocabulary to be taught that is appropriate to the learner, just like any other vocabulary lesson.

To drive the display, we used Apple Script to both recognize the voice input and to control the display of the ASL video clips in QuickTime, using a scriptable feature called "chapter tracks". So with the installation of a short script and the loading of a single QuickTime movie our application was ready to work "right out of the box". Anyone with an Apple computer can use our software without having to purchase or install any additional software. Fortunately our client, the Cleary School, is an all-Macintosh environment.

The limitation of our initial prototype under OS 9 was that we were stuck with multiple windows (one for Apple

Script, one for QuickTime, one for speech recognition), and the graphical user interfaces (GUI) each function provided, which were not customizable. So we migrated to OS X (the new operating system for Macintosh computers) and started using the integrated software development environment (IDE) that is provided with the OS, called Project Builder. This IDE allows the developer to produce code easily in Java, C/C++ and Apple Script, using all the Cocoa classes that are provided to build a professional GUI. We now faced a critical decision of whether to use Apple Script for the coding of the entire application or to move to either Java or C++ limiting the use of Apple Script only for the interaction with the voice recognition engine.

Apple Script is most commonly known as a simple scripting language for automating repetitive tasks using the "scriptable" commands that many Apple applications provide. "Apple Script Recorder" allows a user with no programming knowledge to record a sequence of commands and/or mouse clicks into an executable script. This script can be loaded later on to reproduce the exact same sequence at any time speeding up the user's interaction with the operating system and the most common applications. Yet under OS X, Apple Script has a program structure, similar to LISP, that makes it very easy for the developer to construct complicated data structures as lists of lists. Also, Apple Script inherits from the Cocoa structure the capability to react to the user interaction in the same way Java does through the use of "listeners". Each user action (keyboard strokes, window resizing, interaction with buttons and other GUI elements) can be "listened to" by a registered method that can bring the application into a different state of execution. The only drawback of Apple Script seemed to be that it is an interpreted language. This should have caused a severe decay in the performance of the application, especially since the application loads and unloads from memory frequently large video clips (each file is larger than 20MB). In fact, we were happy to notice that loading the clips in memory is an almost instantaneous process, even on machines with the older G3 processor (333mHz).

3.3 The Content

The video clips of ASL constitute the core content of our application. We wanted the video to be large and clear. We were fortunate to have the Cleary School provide us with a signer who is their parent and infant instructor. Deaf from birth, Mary DiGiovanna is an expert signer who is also very telegenic and personable. The superb video quality was the result of shooting at the ECC studios under the direction of Dini Zimmerman. DV was the format of choice for its quality and reasonable compression. The video was shot at 720 X 480, although it is scaled down in real time in the iSign application.

3.4 Known Voice Recognition Issues

In the first testing phase we encountered some minor problems when we tried to use very short monosyllabic words. Apparently the voice recognition system has problems recognizing words like "egg", but has no problems recognizing polysyllabic words like "hamburger". We tried to understand better why this is happening. We started by recording several different people saying the word "egg". In figure 3 we can see the resulting waveforms produced by a 30 year-old Asian woman and a 32 year-old European man. We think that the differences in the waveforms are too obvious and the length of the sample is too short to give the speech recognition engine enough data to always produce correct interpretation.

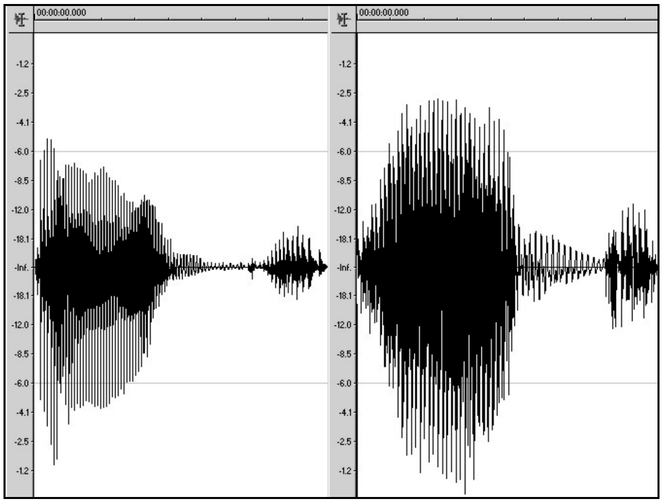

Figure 3. Waveforms for the word "egg" spoken by a woman (left) and a man (right).

On the contrary, we see that the waveforms produced by the same two individuals while pronouncing the word "hamburger" (figure 4) present more similarities both in the wave structure and in the shape of the wave patterns for the same letter. We think that especially the wave structure, which differs greatly among different polysyllabic words, allows the engine to recognize words more accurately.

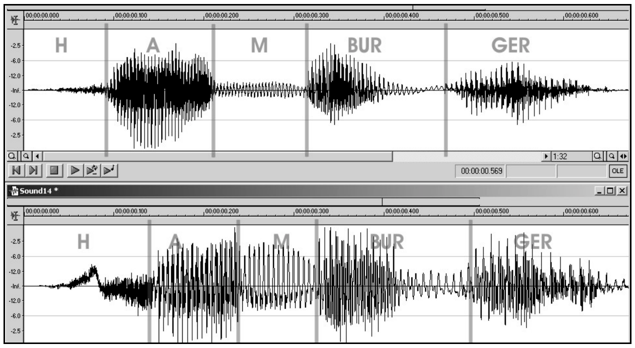

Figure 4. Waveforms for the word "hamburger" spoken by a woman (top) and a man (bottom).

Figure 5 tries to show the strong similarities in the waveforms of two words like “egg” and “bread”. Because this similarity can confuse a voice recognition system, words to be put together in the same “album” must be selected carefully in order to have the best recognition performances. On the other hand, as shown by the very different waveforms presented in figure 6, polysyllabic and longer words are easier to be recognized for the voice recognition system, therefore less attention must be paid when building new albums for these type of words.

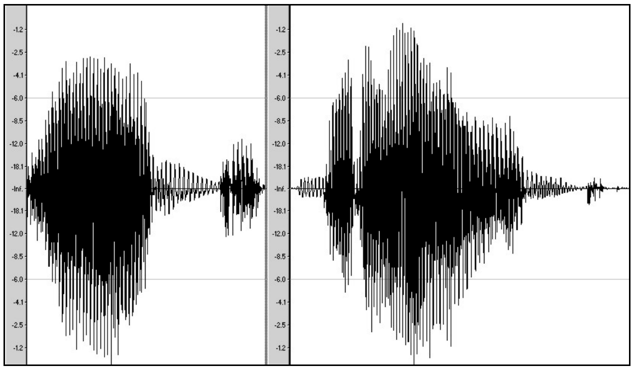

Figure 5. Waveforms for the words “egg” (left) and “bread” (right), spoken by the same person.

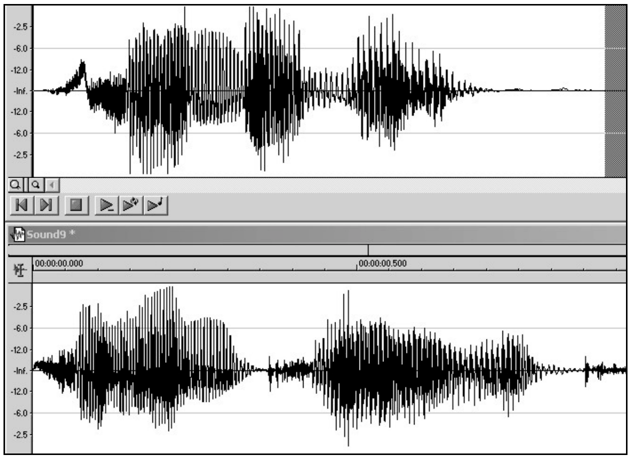

Figure 6. Waveforms for the words “hamburger” (top) and “pancake” (bottom), spoken by the same person.

4. CONCLUSION

We are currently working on expanding the range of words that our application will respond to. We have four albums now of about 15 words each, and each album is about 300MB in size. Our goal is to have at least 20 lessons of about 20 words each distributed on a DVD. MPEG2 compression from the current DV format will reduce file size. The DVD format will preclude the downloading of content to the local drive.

The additional lessons will expand the age groups addressed by the application, and will include math and science concepts, abstract concepts (i.e. government organization), and idiomatic phrases (i.e. “raining cats and dogs”).

iSign is currently being tested at the Cleary School for the Deaf in Ronkonkoma, NY. Following this initial testing we will make the application available to parents of Cleary students to take home on a laptop for further testing.

In the next phase of this work, we intend to extend these ideas so that the system responds to phrases rather than individual words. Once the phrase is recognized and translated to a string it will have to be parsed into the object/property syntax model of ASL. This will allow the storytelling to become more fluid and expressive, and expand the range of material that can be read.

5. ACKNOWLEDGEMENT

We would like to thank Sister Eileen Kelly and Sister Catherine Fitzgibbon of the Cleary School for the Deaf for their guidance and suggestions. We would like to thank Mary DiGiovanna, the signer, Suzanne Dooley, the interpreter, and Dini Zimmerman, the video director, for their help in the shooting of the ASL video clips. This work is being supported by the National Science Foundation under Grant No. 0203333.

REFERENCES

[1] Forster, S. Some Cutting-Edge Gadgets To Even the Playing Field, *The Wall Street Journal*, Nov. 7, 2002. Available online at http://online.wsj.com/article_email/0,,SB1033423870144086393,00.html.

[2] Safar, E. and Marshall, I. The Architecture of an English-Text-to-Sign-Languages Translation System. *Recent Advances in Natural Language Processing (RANLP),* G. Angelova et al (ed), Tzigov Chark Bulgaria, Sept 2001, pp223-228 Available online at http://www.visicast.sys.uea.ac.uk/Papers/confbulgarianew.pdf.

[3] Simon, V. Sign Language on Demand Helps Deaf and Hearing People Communicate. *The New Educator 1*(1), Spring 1995. Available online at http://ed-web3.educ.msu.edu/newed/Spring95/ne5709~5.htm.

[4] McNulty, J. UCSC psychologist teams up with Oregon school to help deaf children. *University of California, Santa Cruz, Currents*, January 19, 1998. Available online at http://www.ucsc.edu/oncampus/currents/97-98/01-19/massaro.htm.