INTERACTIVE DSP COURSE DEVELOPMENT/TEACHING ENVIRONMENT

Chaouki Abdallah[1] *Dalton Arantes*[2] *Gregory L. Heileman*[1] *Don R. Hush*[1] *Ramiro Jordan*[1]
Roberto Lotufo[2] *Neeraj Magotra*[1] *Howard Pollard*[1] *Edl Schamiloglu*[1] *Robert Whitman*[1]

[1]Department of Electrical and Computer Engineering
The University of New Mexico, Albuquerque, NM 87131, USA
[2] Faculdade de Engenharia Elétrica,
Universidade Estadual de Campinas, Campinas, SP 13081-970, Brazil
E-mail: istec@eece.unm.edu

ABSTRACT

The authors have developed an interactive environment for the creation and maintenance of dynamic, active multimedia-based teaching mechanisms. The Environment is designed to be user-friendly and to facilitate the creation of educational material. This tool has already been used to create courses in Multi Dimensional Signal Processing (MDSP) by researchers working together while geographically separated .

1. INTRODUCTION

New advances introduced by the World Wide Web (WWW) impose new ways of building learning materials. New technologies have a great impact on developing and defining new teaching concepts that expose students to real case situations. In fact, many conferences now contain a special session on education using some of these tools [1, 2, 3]. In ICIP'94 [4, 1] and ICIP'96 [5], we reported the use of Khoros as an authoring system for image processing and that we were developing new advances integrating Khoros, the Hypertext Markup Language (HTML) and the WWW. As a result, Digital Image Processing (DIP) Khorosware as a toolbox for the Khoros environment [6] was released. Since then several versions of this toolbox have been released and became the nucleus of a series of courses on MDSP. The first such course was developed by researchers working together at the University of New Mexico, USA, and the Universidade Estadual de Campinas, Brazil.

The purpose of the courses is to give users a hands-on approach to MDSP through an extensive number of experiments. Theory and practice are mixed in such a way that students can immediately relate experimental results to theoretical concepts. This tool is unique in that it: 1) is interactive, 2) can be accessed by Internet browsers running on any platform (workstations or PCs), 3) is freely available, and 4) depends on free access software (Linux, GNU, JAVA, and Khoros software). In addition, some of the applications utilized in the Environment have been created to utilize Matlab, which is commonly available for educational activities.

This paper is organized as follows: section 2. contains a discussion of the organization and the tools used in the course. In section 3. we present the Digital Image processing (DIP) course, while section 4. contains our conclusions and future work.

2. MDSP COURSE ORGANIZATION

The course is divided by topics: The following preliminary topics have been identified.

1. Software Issues
2. Mathematical Preliminaries
3. Filtering
4. Adaptive Methods
5. Nonlinear Techniques
6. Pattern recognition
7. Control
8. Communications

So far, we have developed the DIP module which covers components across (1-3), and (5,6). In section 3. we discuss the DIP course in more detail.

2.1. Objectives

The objectives of the Interactive Environment are 1) to provide tools for education on readily available hardware and software, 2) to create tools in the environment that are user-friendly for producer and consumer, 3) to allow for user-supplied data, as well as providing extensive data sets for experimentation, 4) to develop a methodology that is adaptable to allow presentation of a wide range of materials, from simple to complex, and to allow self-paced study, and 5) to create a mechanism which can be utilized to rapidly incorporate research results into the classroom.

2.2. Development/Teaching Environment Tools

In order to achieve the above mentioned objectives, a set of tools has been developed. The tool set utilized for this environment allows professors to generate material with familiar tools, which minimizes the learning curve for the creation of didactic materials. For text files, the facilities of the tool set are accessed by placing markers in the material which invoke appropriate responses from the computer system. Thus, facilities such as Hypertext, simulation engines, analysis software, and display routines can be easily incorporated into a multi-media teaching system.

The Interactive Environment consists of a suite of programs, filters, facilities, techniques, and mechanisms which can be called upon by developers of teaching systems to generate a variety of different materials. The instructor simply disciplines himself/herself to utilize the guidelines and mechanisms of the Interactive Environment in the preparation of the teaching materials, and the net result is a set of computer files which will be presented at the appropriate time in the desired format. When the instructor is ready, the source file(s) are utilized to generate a set of HTML files for utilization on the World Wide Web (or a local version of the web), to generate Postscript files for printing hardcopy, or to create appropriate LaTeX files for utilization in various publication software systems. Not only does the Interactive Environment allow instructors to include markers in the textual files which form the basis of the didactic material, the Environment also includes aids in the creation of simulation systems, display mechanisms, and the inclusion of interactive programs within the teaching environment.

0-8186-7919-0/97 $10.00 © 1997 IEEE

This permits the examination of techniques and methods by having real-time access to programs and data sets, and the demonstration of the effectiveness of algorithms by implementing the algorithms on both canned data sets and user-supplied data sets.

As an example, the visual programming facility *Cantata* of Khoros 2.1 allows instructors to prepare algorithms and data sets for inclusion in teaching material. This material need not be static in nature, leading to figures and text; the materials can be dynamic, allowing full use of the facilities available in a multimedia environment. The instructor need not know about data formats, programming languages, or input/output systems. The algorithm is simply implemented by connecting blocks or "glyphs" together with appropriate links. The glyphs represent an underlying program or algorithm, and the links specify the required data flow for the system being demonstrated. Hence, an instructor can quickly generate desired prototype systems, calling upon glyphs for functions such as transforms, filters, input/output, or display. Once the algorithm has been properly prepared, the entire system can be coupled together and represented by a new glyph, allowing students to invoke the facilities of the system by calling upon the glyph, in the appropriate computer environment. This ease of data manipulation leads to quick and easy development and maintenance of course materials and teaching systems. It also allows students to test, debug, and experiment with algorithms and data as they explore and gain insight into the concepts of signal processing.

Since the system is created to facilitate the educational process, teachers and students need to apply the algorithms to a wide variety of data types to gain an understanding of the applicability of the processing in different situations. For this reason, care has been taken to provide a wide variety of data sets which can be operated on by the algorithms. In addition, user-supplied data sets can easily be substituted for the canned data sets, allowing students to test algorithms on real problems, rather than rely on the data sets provided.

The methodology of the Interactive Environment allows teachers to prepare materials for use in assorted presentation formats. This includes traditional textbooks for use in the classroom, textual material with Hypertext links, animations, graphics displays, and other mechanisms for the presentation of information.

One of the by-products of the Interactive Environment is the ability to rapidly introduce into a teaching situation, results of research being conducted. That is, as the algorithms, data sets, and display mechanisms become available they can be used to support and extend the concepts being presented to students. And, since the methodology inherently delivers information quickly and easily, collaboration between colleagues across geographical distances will be facilitated and contribution of students and others to ongoing projects will be encouraged.

3. DIP COURSE

One example of this system is a course on Digital Image Processing that is available over the Internet for self-study. The material has been presented in short-course form and is also being published in textbook form, with accompanying CDROM for data processing and visualization. The course is also being used as a more in-depth course in the curriculum of the University of New Mexico and the Universidade Estadual de Campinas in Brazil. Many other sites are making use of these materials. More information about the course is available at http://www.eece.unm.edu/dipcourse.

The DIP course is comprised of chapters which are then divided into lessons. Each lesson provides both theoretical and practical sessions. The theoretical session describes the problem or the concept for the experiment, and is typically full of images and graphics showing the original data and the processed data or final result. It explains the fundamentals of image processing in general. This session is independent of the software used and can be used as a non-practical illustrative course, i.e., without the "hands-on" touch. At the end of each Web page, there are two hyperlinks: 1)`Other Examples` and 2) `Laboratory`. `Other examples` provides additional examples of the same concept, but applied to other practical areas. For instance, if the first part is focused on medical imaging, these additional examples may focus on document image processing. `Laboratory` is the practical portion of each lesson. This has the format of a lab guide, with step-by-step instructions on how to accomplish and reproduce the results shown in the first part of the lesson. In the lab guide the operator names are actual links to Khoros operators, which are activated by the network browser using the `mime` type convention. A special file type for the Khoros form file related to each Cantata operator is defined. Following the lab guide, there is a list of suggested exercises that provide further questions and problems so that users can expand and practice what they have learned. At the end of `Laboratory`, there is a link to a visual program workspace which takes the students directly to Cantata where a proposed solution is loaded and is ready for execution. Now users can run the operators individually or the whole workspace, change parameters, use other images, check and compare the results, etc.

The *traditional* approach to learning combines a textbook, a sequence of lectures and tests, and perhaps a project in which the student writes some code using a standard programming language. By combining interactive text and data with the Khoros 2 system we have taken steps to make the learning process more dynamic. Hypertext and the Khoros 2 system becomes a conduit between the instructors and students cutting across geographical, temporal, and cultural distances.

A great advantage of this course is that students learn

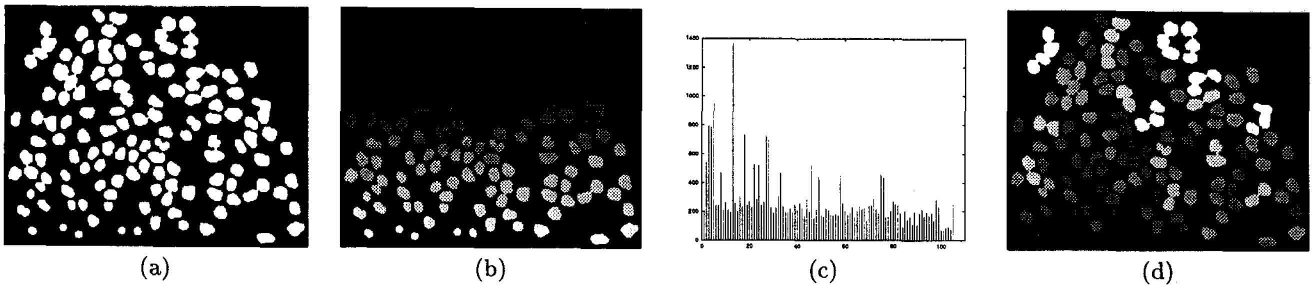

Figure 1. a) original image, b) labeled image, c) histogram of labeled image d) area display image

by programming using a high-level visual language found in an existing data processing tool used to solve practical problems. Furthermore, by allowing the students to modify the input data and operator parameters, they can immediately verify the results, determine any limitations of an algorithm, improve the solution of a particular problem.

In order to teach the fundamentals of those primitive data processing operators, the course sequence follows a non-traditional approach of introducing a subject, that is, a particular problem or a concept. We introduce classes of primitive operators and, for each class, explore its use for many applications. As an example, when the single-operand point operations are introduced, we use them for image segmentation by thresholding, for contrast enhancement, for pseudo-coloring, and to create functional data, i.e. images with pixels displaying feature values. In this manner, students learn a new operator and see the many situations where it can be applied, helping them to create a solid hierarchical concept of image processing tools.

Fig 3. shows the related images from a lesson in chapter 4 which makes use of labeling, histogram and color table primitive operators to create a functional image in which the pixels of the image represent the area of its connected region. This is a very useful tool for image analysis. Fig 1a shows the original binary image, and Fig 1b shows the result of the application of the labeling operator in the original image. In the labeled image each pixel value gives the unique region number to which it belongs. The regions are numbered sequentially starting from 1. The histogram of the labeled image, shown in Fig 1c, gives the number of pixels of each region, i.e. its area. Using the histogram as the mapping function (color table) of the labeled image, each pixel value is replaced by its area (Fig 1d). This lesson is a good example of the creation of a powerful interactive image analysis tool using primitive operators.

A very active research area is the access and retrieval of images which has numerous applications. Some of these applications are the storage of images on disk, the transmission of images via the Internet, and document processing just to mention a few. Toward this end several groups are currently working on developing the necessary software environments to achieve reliable and cost-effective transmission of documents. A promising technology has shown to be wavelets for the compression of large amounts of data [7]. To give a simple example of how the DIP Khorosware can be utilized to teach, learn and report on research progresses is given next.

The current technology to achieve high rates of data compression is the Embedded Zerotree Wavelet (EZW) coding technique introduced by Shapiro [8]. This is a very effective and computationally simple technique for image compression. Here we describe a realization of this technique under the Khoros 2 system following the ideas of Said and Pearlman [9]which proved to be extremely fast and efficient.

Depicted in Fig 3. are six images with different bit rates. The original image is an 8-bit 256x256 monochrome image. We can observe that the difference between the original image and the 0.25bits/pixel image is almost imperceptible. This act is very important in many applications such as progressive transmission, image/document browsing, and multimedia applications.

4. CONCLUSIONS AND FUTURE WORK

The nucleus for this project grew out of a transcontinental collaboration between the University of New Mexico, USA, and the Universidade Estadual de Campinas, Brazil. It is hypothesized that this course will grow with the active participation of the user community. This is an ongoing project and the issues of development and maintainability must be considered. Lessons and experiments will be continuously extended and enhanced, and the tools to facilitate this development need to be continuously upgraded. These tools enable the generation of interactive electronic pages and book-like hardcopy materials. We see SGML as a solution to the above issues.

With the help of new Internet tools, teaching is being facilitated and encouraged. Furthermore, this new format of interactive courses not only provides the necessary material to learn a new topic in a timely manner, but it provides a new mechanism for the introduction of research results into the classroom. In the future, multimedia mechanisms, as well as collaboration technology, will enhance our ability to learn and work together across geographical, temporal, and cultural distances. The future belongs to collaboration and education across international borders, mediated by computers and high-speed links.

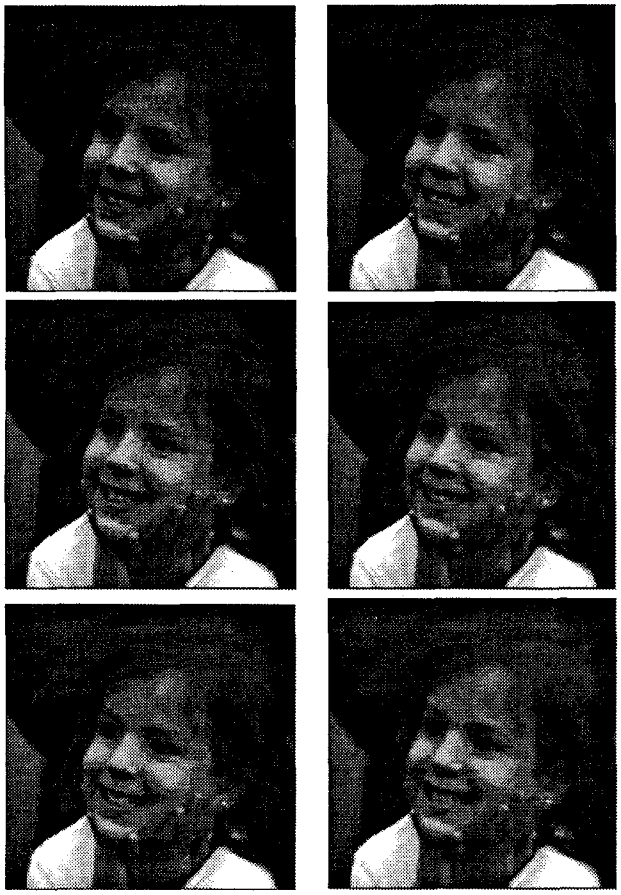

Figure 2. *First row:* **Original and 2 bits/pixel image;** *Second row:* **1 bits/pixel image and 0.5 bits/pixel image** *Third row:* **0.25 bits/pixel image and 0.125 bits/pixel image**

5. ACKNOWLEDGMENTS

This work was supported by the Ibero-American Science and Technology Education Consortium (ISTEC) through the generous donation of Motorola Inc., Nortel, Khoral Research, Inc., Fluke Corporation, IBM-Brazil, the Organization of American States (OAS), Conselho Nacional de Dese nvolvimento Cientifico e Tecnológico (CNPq - Brazil), Fundacao de Amparo a Pesquisa do Estado de Sao Paulo (FAPESP - Brazil), Colciencias (Colombia), CICYT (Spain), Xunta de Galicia (Spain), and the State of New Mexico.

REFERENCES

[1] "Session on image processing in education," in *Proc. IEEE International Conference on Image Processing*, vol. I, (Austin, TX), pp. 501–529, November 13-16 1994.

[2] "Session on image processing in education," in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, (Atlanta, GA), 1996.

[3] "Session on curriculum advances in digital image systems engineering," in *Proc. IEEE International Conference on Image Processing*, vol. I, (Lausanne, Switzerland), pp. 431–460, September 16-19 1996.

[4] J. Rasure, R. Jordán, and R. Lotufo, "Teaching image processing with khoros," in *Proc. IEEE International Conference on Image Processing*, vol. I, (Austin, TX), pp. 506–510, November 13-16 1994.

[5] R. Jordán and R. Lotufo, "Interactive digital image processing course on the world wide web," in *Proc. IEEE International Conference on Image Processing*, vol. II, (Lausanne, Switzerland), pp. 433–436, September 16-19 1996.

[6] R. Jordán and R. Lotufo, "Digital image processing with khoros 2," 1995.

[7] A. Bruce, D. Donoho, and H.-Y. Gao, "Wavelet analysis," *IEEE Spectrum*, pp. 26–35, october 1996.

[8] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3345–3462, December 1993.

[9] A. Said and W. Pearlman, "A new fast and efficient image codec based on set partitioning in hierarchical trees," in *Proc. IEEE International Symposium on Circuits and Systems*, (Chicago IL), May 1993.