



Real-time web-based hyperspectral data viewing and control

Mehrube Mehrubeoglu, Linh Manh Pham, Nagarjuna Paleti

► To cite this version:

Mehrube Mehrubeoglu, Linh Manh Pham, Nagarjuna Paleti. Real-time web-based hyperspectral data viewing and control. Applied Imagery Pattern Recognition Workshop (AIPR), 2011 IEEE, Oct 2011, Washington, United States. 10.1109/AIPR.2011.6176372 . hal-01233258

HAL Id: hal-01233258

<https://hal.science/hal-01233258>

Submitted on 24 Nov 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Real-time Web-based Hyperspectral Data Viewing and Control

Mehrube Mehrubeoglu*, Linh Manh Pham and Nagarjuna Paleti

School of Engineering and Computing Sciences
Texas A&M University-Corpus Christi
Corpus Christi, Texas, USA

*Ruby.Mehrubeoglu@tamucc.edu

Abstract—There is significant interest in hyperspectral image data in multiple disciplines. Laboratory-based hyperspectral imaging systems support research on samples ranging from plants and agricultural products to cell cultures, algae, coral, metals, and other media of interest. In most traditional laboratory settings, the user selects the data acquisition parameters and controls the instrument in the laboratory. In this paper interfacing a lab-based hyperspectral imaging system with the Internet through NI LabVIEW and related programs is described to allow real-time remote access and control of the system as well as viewing of the captured data. Through the designed interface, hyperspectral image cubes can be acquired and stored remotely, and viewed on-line by individuals who are not in the lab but are granted remote access to the system. The functional model for the remote control hyperspectral imaging system and the designed interface for real-time web-based data viewing and control are presented in this paper.

Keywords—hyperspectral imaging, hyperspectral imaging system, real-time viewing, real-time data acquisition, real-time control, Web-based interface, graphical programming

I. INTRODUCTION

Hyperspectral imaging is a technique for acquiring 3D images called hyperspectral image cubes or datacubes. Each hyperspectral image cube provides both conceptual (2D picture-like) and spectral information, and are useful in the visual and optical characterization of materials and media. Unlike a 1D single spectral signal, hyperspectral images provide multi-dimensional spectral information of the tested medium, which can then be analyzed for localized (at each pixel or pixel group) optical properties such as absorption coefficient and scattering coefficient within the hyperspectral image at different wavelengths.

Laboratory-based hyperspectral imaging has found applications in many disciplines not only because of the spectral and spatial information it can provide, but also temporal analysis that can be performed with such systems under controlled conditions. The hyperspectral image cubes are acquired manually where the user adjusts acquisition parameters in the lab. The hyperspectral imaging experiments are usually performed in a dark-room environment, which may not be comfortable for many users for extended periods of time. Moreover, distant users who may not have easy access to

the system cannot provide immediate feedback on the acquired data. The need for allowing users to access the lab remotely and the commitment to outreach activities using the hyperspectral imaging system (HIS) in the Hyperspectral Optical Property Instrumentation (HOPI) Laboratory led to the development of a virtual laboratory for the HIS. The virtual laboratory allows authorized users such as collaborators, research assistants and students interested in STEM education to control the HIS remotely in a real-time manner through a web-based interface, and observe the results on-line.

A. Background and Motivation

The increased access to and ease of use of the Internet has facilitated new approaches and methods for designing and developing virtual and real-time laboratory applications. The concept of remote laboratories has been investigated by various groups as educational tools to support distance education and remote learning [1]-[6]. Remote labs not only support those students who are unable to attend traditional laboratories due to physical distances, but also allow more flexibility in the use of the laboratory equipment.

Among the many Web-based virtual laboratory examples proposed in the literature and implemented in practice, Cheng *et al.* present a power electronics experiment that is conducted in a remotely controlled laboratory [7]. The power system is controlled and monitored via a Web-based tool. Ko *et al.* introduce a Web-based virtual laboratory to carry out frequency modulation experiments in an undergraduate course [8]. Xiaoyan *et al.* developed an electrical and electronics virtual laboratory that describes and demonstrates various aspects of instruments and circuits with LabVIEW programming [9]. Another Web-based distance learning platform is created to benefit an electrical measurements course by Dormido *et al.* [10].

Most of the remote laboratories involve pre-designed experiments with cook-book answers to teach and learn predetermined concepts. In this paper, we describe access to a laboratory-based hyperspectral imaging system set up for research purposes. Typically, a lab-based HIS operates under dark-room conditions to avoid ambient light contaminating the results. Hyperspectral images are complex and require interdisciplinary expertise for analysis and interpretation.

The HIS instrumentation and integration is funded by NSF MRI-R² grant #0960000. This project is sponsored in part by TAMUCC Office of Research and Graduate Studies.

Hyperspectral image cubes vary in size and can occupy well over 3 GB of memory when scanned at full resolution. The sheer size of these datacubes limits the feasibility of fast data transfer for analysis. With the remote access and control, data can be captured and viewed on the host computer without transferring the data on the Internet. For the goals of this project, a model which supports remote acquisition and analysis of hyperspectral images has been adopted with Web-based LabVIEW interface using NI Internet Toolkit and NI LabVIEW Web UI Builder. By this interface, an authenticated user may access and control the HIS system remotely. The experimental results are stored in a FTP server which serves the purposes of querying the analyzed data and viewing the images from the Internet without the need for transferring the data to a remote computer.

B. The Interface Programming Tool

In this project, LabVIEW has been used as a high-level graphical programming language to build the interface for the virtual laboratory. This application requires real-time access and control of the HIS hardware; LabVIEW provides tools for building a virtual laboratory with functionality that supports testing, instrument control, data acquisition, data processing and analysis, as well as various modes of viewing. The virtual instrument is designed and controlled via GPIB, RS232 or VXI standard interfaces. The Web-based applications are built using the NI Internet Toolkit and NI Web UI Builder. The LabVIEW program provides both logical and physical visualization of devices as well as their simulations [11].

The built-in Web server that supports regular world-wide Web languages such as HTML and XML is utilized. The Web server also supports LabVIEW modules to simulate the control and viewing of mechanical and digital devices and their input/output. With this toolset, the hyperspectral data can be visualized in real-time. The NI LabVIEW Internet Toolkit supports a variety of network application-layer protocols such as HTTP, FTP, and SMTP. Popular applications providing remote control services including VPN, VNC or Logmein require manual set-up software at the client side (client agent). With LabVIEW, run-time modules integrated into the client browser as plug-ins are installed in the client machine automatically while a HTTP header is being called, hence the choice of this toolset to create the Web-based interface.

The differences between local and remote hyperspectral imaging system are summarized in Section II. The functional model for the local and remote HIS are also presented in this section. In Section III, both the data communication hardware and graphical programs interfacing the system are described. In Section IV, the user interfaces with menus that have been embedded into the Website are described. The conclusions are presented in Section V.

II. REMOTE HYPERSPECTRAL IMAGING SYSTEM

A. Functional Model

The interface for the hyperspectral imaging system described in Section I must have capabilities for collecting

data from various sources. In addition, the user must be able to then analyze, store as well as share the captured data. Similar efforts to accomplish the above are describe by Kerekes and Baum [12], and Learner and Drake [13] for system modeling and application development in locally controlled (local) HIS. A functional model of the local HIS for the system described in this paper is shown in Fig. 1.

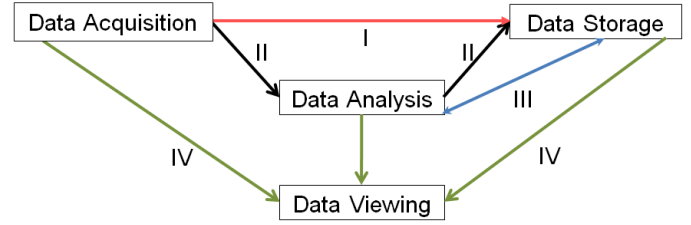


Figure 1. Local HIS Functional Model

In this model, the user accesses the HIS locally in the lab for all acquisition, storage, analysis, and viewing purposes: Hyperspectral images can be obtained in the Data Acquisition module and sent directly to the Data Storage module for storage (I, red). The hyperspectral images are directly acquired from the system through system control functions, and saved on the host computer or server for data storage purposes. The obtained hyperspectral images can be sent to the Data Analysis module for image processing. Hyperspectral images are analyzed through independently-developed image processing functions. The result can be saved through the Data Storage module (II, black). The stored data can be retrieved from the Data Storage module, sent to the Data Analysis module for image processing, then sent back the Data Storage module (III, blue). In addition, data can be viewed from Data Acquisition, Storage or Analysis modules directly (IV, green).

In comparison with the local control hyperspectral imaging systems, research in the remote control mode has not been as widely published. Several remote control models are described by Vladimir *et al.* who developed a tool for Web-based viewing of hyperspectral images [14], and by Krehbiel *et al.* who used LabVIEW interface for remote laboratory access [15]. In the former, the interface is utilized for web-based data processing, and unlike the HOPI HIS, not for remote device control [14]. In the latter, the system is used for educational experiments [15].

For research conducted with the lab-based HIS, local HIS cannot meet the demands for system access and data viewing by multiple users outside the physical laboratory environment. As a solution, a remotely controlled HIS model has been devised to address the remote system and data access and control needs for the HIS. In Figure 2, the devised seven-module Remote HIS functional model is depicted. With this model, similar to the local HIS model, the hyperspectral images can be obtained in the Data Acquisition module and sent directly to the Data Storage module to be saved on the local host computer (I, red). The acquired hyperspectral images can be sent to the Data Analysis module to apply image processing algorithms first and then to the Data Storage module (II, black). The stored data can be retrieved from the

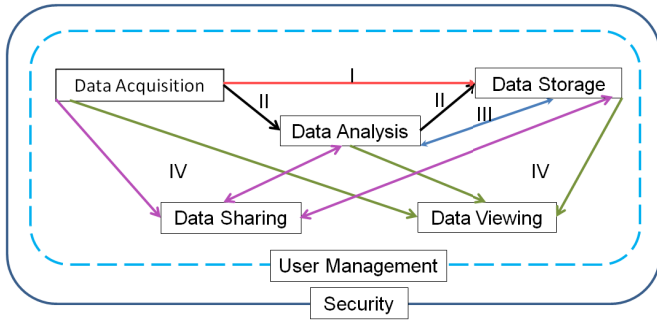


Figure 2. Remote HIS Functional Model

Data Storage module and sent to the Data Analysis module. The result can then be sent back to the Data Storage module (III, blue). The data from the Data Acquisition, Analysis, Storage modules can be sent to the Data Viewing module for online viewing (IV, green arrow). In addition to the above, the remote HIS functional model allows the data from the Data Acquisition, Analysis, Storage modules to be sent to the Data Sharing module to be viewed by multiple users (IV, purple). The user management module keeps track of multiple users who access and control the system. The security module affects every other module to ensure secure access to the system.

B. Web-based Application Interface for Remote HIS

A Web-based application has been developed based on the remote HIS functional model. The application provides a Web-form interface running on LabVIEW Web Server 2010, and supports remote acquisition and analysis of hyperspectral images. Through this interface and the NI Internet Toolkit, authenticated users can access and control the HIS remotely using only a regular computer with a Web browser connected to the Internet. The experimental results are stored in a FTP server or the local host computer which serves the purposes of querying the captured data and viewing the images from the Internet.

In the laboratory, a LabVIEW-based application integrated to the HIS that was packaged with the hardware is used for image acquisition. This software allows users to collect and view hyperspectral data of limited sizes when researchers are physically present in the laboratory and in front of the HIS. The development of the HIS Web-based application hides the underlying complicated hardware and software as well as eliminates the need to install and run the application on the client's own computer. This partly fulfills requirements of Software as a Service (SaaS) in Cloud Computing terms [16].

III. DATA COMMUNICATION INFRASTRUCTURE

A. The Hardware System

The network topology of the real-time Remote HIS is shown in Figure 3. The hardware connections are set up such that the horizontal stage controller, which enables the acquisition of spatial data, is connected to the central server through the RS232 standard cable. An adapter is also used to

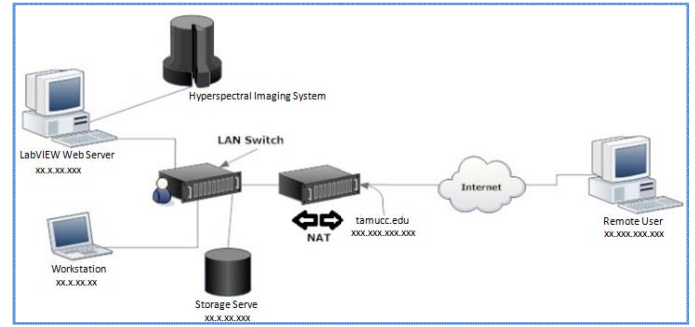


Figure 3. Network Topology of the Real-time Remote HIS

convert RS232 interface to the COM interface supported by the central server. LabVIEW is adopted to implement the local instrument control. On the basis of the RS232 address assigned to the instrument, the serial commands are transferred through the cable to the corresponding RS232 box which executes these commands for the mechanical movement of the stage. The laboratory PC, where the LabVIEW and Ethernet interfaces are implemented, acts as a central server and communicates as a full duplex form with the HIS. The full duplex form allows communication in both directions for faster transmission.

The camera itself is connected directly to the central server through a regular FireWire cable (IEEE 1394). Both the imaging data and control signals are sent back and forth in the same cable. This is a smart CCD camera equipped with functional modules such as camera RAM (camRAM), camera processor (COC processor), and interfaces (FireWire, Camera Link or Gigabit Ethernet), similar to a regular PC but in a miniature scale. The components of the hyperspectral imaging system located in the HOPI Laboratory are shown in Figure 4.

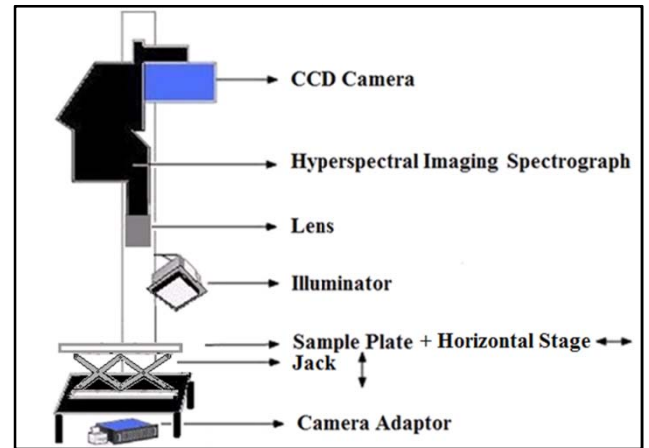


Figure 4. The Components of the Local Control HIS

The central server is connected to a cluster of FTP servers located on the same network. The hyperspectral images are sent to the FTP cluster for storage.

B. Graphical Programs

The graphical programs or block diagrams are created in the graphical programming environment for LabVIEW applications. Connections between the virtual instruments

(VIs) and analog I/O devices are implemented in LabVIEW 2010 using the G programming language. Figure 5 shows the block diagram of the “Start-up Page” for the user interface of Figure 6. Figures 7 and 9 depict the block diagrams for “Application 1” and “Application 2” for the user interface of Figures 8 and 10, respectively.

In general, each block diagram is divided into modules corresponding to configuration parameters. Only the main modules are shown in the figures. Each module calls appropriate subVIs that serve as sub-functions. For example, the module named “Spectral binning” in Figures 7 and 8, which is used to adjust the spectral averaging parameter of the camera, needs to call GetBinning.vi, SetBinning.vi and other related subVIs from the LabVIEW library named Sensor.Ilb.

In the “Application 1” block diagram shown in Figure 7, the main modules consist of the camera configuration, stage controller configuration, BIL (band interleaved by line) and HDR (header) file creation and storage, displaying real-time camera output screen, and plotting real-time hyperspectral data. Through these modules in Application 1, the camera and stage controller’s parameters are set, and then a sample is scanned to obtain the hyperspectral datacube. The BIL file contains the scanned data. The HDR file contains information on parameters such as number of rows, columns, acquired spectral bins as well as the data format and wavelengths corresponding to the bin numbers (Figure 7).

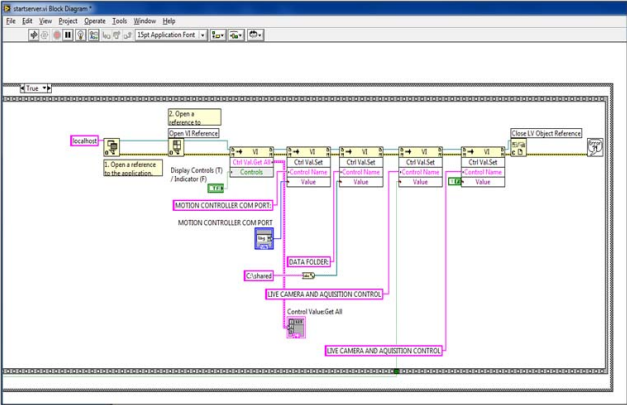


Figure 5. Block Diagram of the Start-up Program

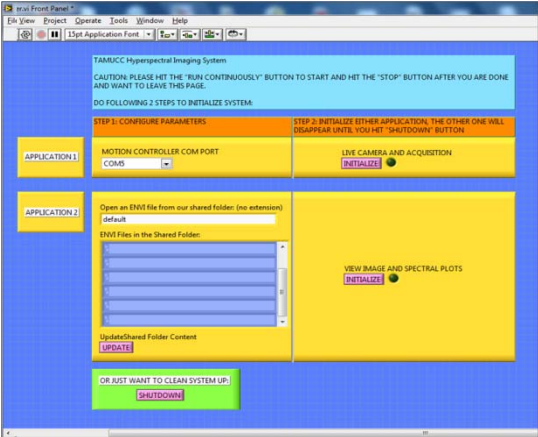


Figure 6. Front Panel of the Start-up Program

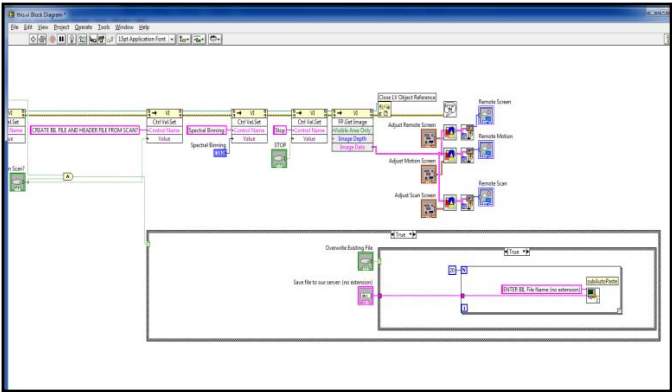


Figure 7. Block Diagram of Application 1

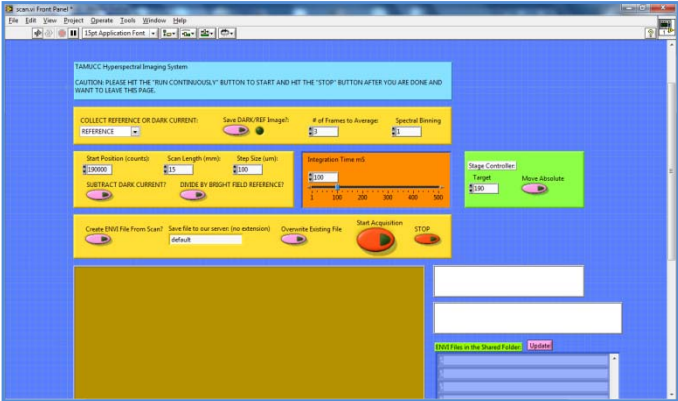


Figure 8. Front Panel of Application 1

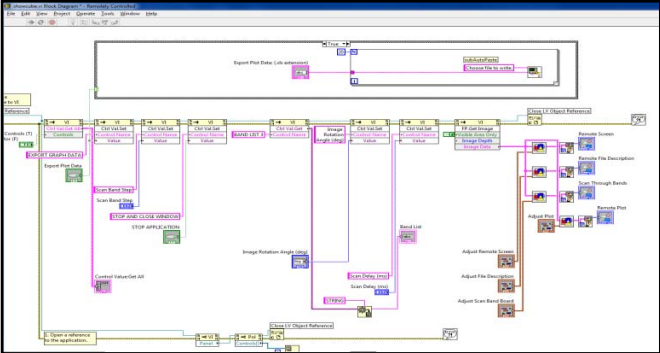


Figure 9. Block Diagram of Application 2

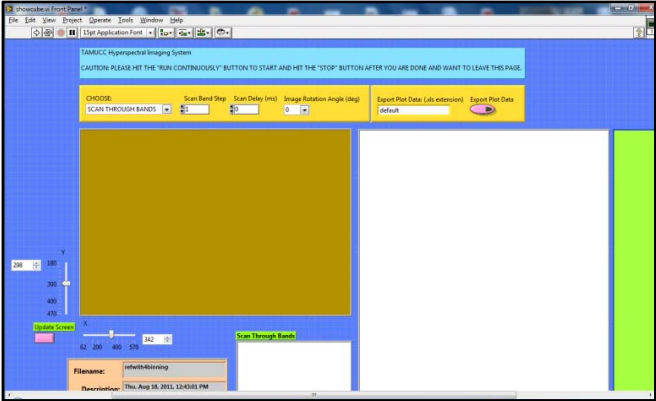


Figure 10. Front Panel of Application 2

Similarly, in the “Application 2” block diagram shown in Figure 9 associated with the user interface of Figure 10, the main modules consist of viewing configuration, opening and showing the contents of a hyperspectral image, plotting stored hyperspectral data, saving the plot to a spreadsheet, and opening and reading the contents of a spreadsheet.

IV. THE USER INTERFACE

A. Front Panels

The front panels for the system itself represent the user interface with the system and provide the necessary user control functionality. For the start-up and application pages, the front panels are divided into two sections: control buttons and output screens. The control buttons are used to configure the set-up parameters of the camera and the horizontal stage controller. The output screens show the results corresponding to input parameters of system. The results can be shown as a waveform or in a real-time signal capture screen. The front panel is arranged in user-friendly top-down design with the control buttons section above the output screen section.

In the front panel of the start-up page (Figure 6) the preliminary configuration interface of two applications is presented. With “LIVE CAMERA AND ACQUISITION,” the user selects the communication port which is the interface between a central server and the stage controller. To set up the system, a sample needs to be placed on motion-controlled stage. The stage movement is controlled by the central server. The purpose of the stage controller is to move the sample in the horizontal direction under the camera and spectrograph. This motion is required for push-broom hyperspectral imaging systems to build the spatial dimension of the hyperspectral images with the required spatial resolution between horizontal pixels. With “VIEW IMAGE AND SPECTRAL PLOTS,” the user at minimum must select the hyperspectral file to be opened. These files have been stored in a shared folder where authenticated users can save their BIL and HDR files. A list of files in the shared folder is also provided to remind the users of the hyperspectral file names (Figure 6).

The purpose of Application 1 is to acquire the hyperspectral images through the CCD line-scan camera. Application 2 is a tool used to view the stored hyperspectral data. Because the two applications cannot run simultaneously, after the users are finished with one application, they must restart and also refresh the system. Then they can initialize the other application. A “Shutdown” button is provided for exiting the applications as well as clearing the system. The front panels for Application 1 and Application 2 shown in Figures 8 and 10 respectively are described below.

In “LIVE CAMERA AND ACQUISITION” page accessed through the startup page of Figure 6 (Application 1), the parameters of camera that the user controls include number of image frames to be averaged, number of spectra to be binned, integration time which affects the amount of photons entering the system, and the collection of reference and dark current signals for normalizing the data (Figure 8, top). These parameters are set by the user in real-time using the

appropriate control buttons; therefore, the users can change the quality of an output hyperspectral image in real-time through the change of these input parameters, viewing the results and readjusting the parameters as necessary. A control panel including control buttons for the stage is also presented in the front panel of Application 1. The users can specify stage’s start position, length of scan, and step size that determines the spatial resolution of the images. The absolute position of the stage controller can also be adjusted. A screen shows the real-time spectral image being captured and recorded by the camera (Figure 8, bottom left). When the user scans the sample, the current position of the stage is updated in the position box. The percentage of scanning is reported real-time. When the user activates the real-time screen, a pointer shows up in this screen. With this pointer, the user can select a pixel to display its spectral content using four buttons: “Up,” “Down,” “Left” and “Right” (Figure 11). The real-time information of the selected pixel is displayed on the main screen.

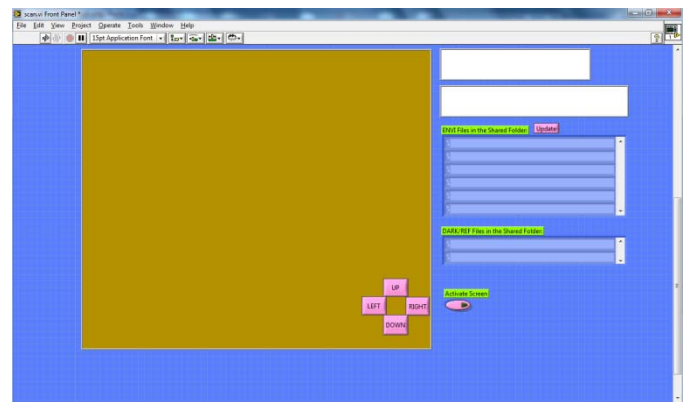


Figure 11. Real-time Output View Control for Pixel Selection

In “VIEW HYPERSPECTRAL IMAGE AND DATA PLOTS,” the users can choose the viewing mode (Figure 10). The viewer can view an image at individual spectral bands, display an image in gray scale, or show an RGB color representation of the image. The speed of displaying the image planes and the sample bands at which the images are to be viewed can also be adjusted. The left screen displays the hyperspectral image plane the BIL-formatted hyperspectral file. The image can be rotated 90, 180, 270 degrees. The right screen shows the plot of amplitude versus spectral bands corresponding to a single pixel selected in the displayed image. This plot can be exported to a spreadsheet file. Both a list of pixel intensities and the corresponding wavelength are stored in the spreadsheet representing a spectrum from a pixel. The content of this file can be accessed later using Application 2 itself or using spreadsheet software. Again, the users can activate the left screen and select a pixel from the image using pick-up tool at the bottom-left corner of this screen. The pixel’s spectral information is then displayed as a waveform on the screen.

The experiment can be stopped immediately by clicking the “Stop” button located at the top left of all three front panels. To start these applications, the control panels and graphs must be connected in the proper form in the block

diagrams presented earlier in Figures 5, 7 and 9. The list of configuration parameters in all three LabVIEW front panels is summarized in Table 1.

TABLE I. SYSTEM CONFIGURATION PARAMETERS

Parameter	Description
Integration Time (μ s)	Camera integration time in microseconds
Spectral Binning	Spectral pixels to be Averaged
Move Absolute	Move the stage to the desired position
% Scan Complete	Scanning progress indicator
STOP	Stop scan (Hold down for ~ 1 second to complete STOP)
Start Position (counts)	Start position of scan 2000 counts = 1 mm
Scan Length (mm)	Total scan length in mm
Step Size (μ m)	Distance stage travels between spatial scan lines (row/column of pixels)
Create ENVI File From Scan	Saves data to user-specified (BIL) data file. No extension added to file name. An ENVI compatible header file is created with the same name and a .txt extension automatically.
Start Acquisition	Start scanning
IMAGE	8-bit grayscale or 24-bit color (RGB) image of data. Data plotted in original bit depth.
GRAY	Generates 8-bit grayscale image of the scanned object at user selected band (wavelength). Bands loaded from header file.
RGB	Used only for visible spectrum scans. Generates color image of the scanned object. Bands loaded from header file. Select RED, GREEN and BLUE band to create color image. Bands closest to the following are selected: RED: 660 nm GREEN: 567 nm BLUE: 460 nm
Scan Through Bands	Scan through spatial grayscale images at each band scanned
Scan Band Step	Set "SCAN THROUGH BAND" step. Speeds up scan for large files
Scan Delay	Set "SCAN THROUGH BAND" time delay (ms). Slows scan for smaller files from 0 to 10,000 ms

B. The Web Site

The Website is designed to perform experiments with the Hyperspectral Imaging System (HIS) remotely. The Web-based HIS applications have a total of four user interface pages listed as a user authentication page, a start-up page and two separate application pages (Figures 12-15). A LabVIEW Web server 2010 is built to support the Start-up page and two application pages. This Web server operates in a port dedicated to LabVIEW HTTP requests. The user authentication page is developed by PHP and runs on Apache Web server (Figure 12). The Apache Web server is installed in the same physical server, operating in port 80. The Web server serves the regular HTTP request coming from the Internet.

The Web site is designed in accordance with Microsoft Internet Explorer 7.0 or higher and Mozilla Firefox 5.0 or higher. In the real-time application, the Web site allows only one user to control the system at one time since there is only one set of physical devices; however, multiple users are accepted to view the system simultaneously without changing control parameters. When the user logs into the system, he or she requests control to obtain exclusive access to the system. If the system is available, the central server gives the user the requested control. The control is granted to user for a maximum of five minutes in case another user is waiting to use the system. If no one else is waiting for the HIS control, the current user can have unlimited time on the HIS. If the system is not available at the time of user request (another user is using the system), this request is put into a queue. During this time, the candidate user can only see the activities which are being performed by the controlling user on his or her computer screen. The control will be granted to the next user in line either when the user in control of the HIS closes the Web browser, or the maximum use time of five minutes pass. The database of the users and related logging information is stored in a MySQL database.

Through the National Instruments Dynamic Link Library (DLL) that is part of the NI Internet Toolkit and NI Web Builder the virtual instruments developed in LabVIEW can be integrated into .NET. Thus LabVIEW components such as the control panels, graphs or plots can be converted into the corresponding components in .NET when we want to publish the LabVIEW front panel to the Web. Many functions and components presented in DLL have been used in our designed Web site.

In Figure 12, the user authentication page is shown. Similarly, the corresponding pages of the LabVIEW front panels are represented in Figures 13, 14 and 15, respectively. The remote user can login to the hyperspectral imaging system as long as the system in the lab is turned on. The local user has precedence over remote users on the use of the system.

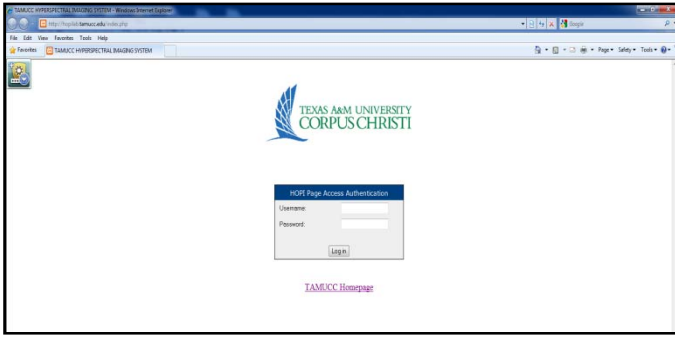


Figure 12. Web-based User Interface for User Authentication and the Homepage

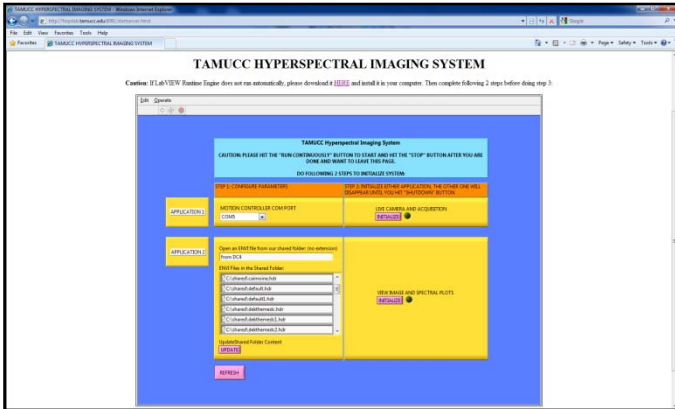


Figure 13. Web-based User Interface of the Start-up Page

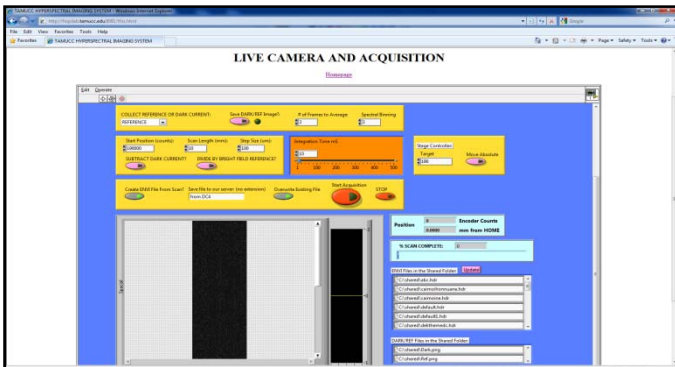


Figure 14. Web-based User Interface of Application 1

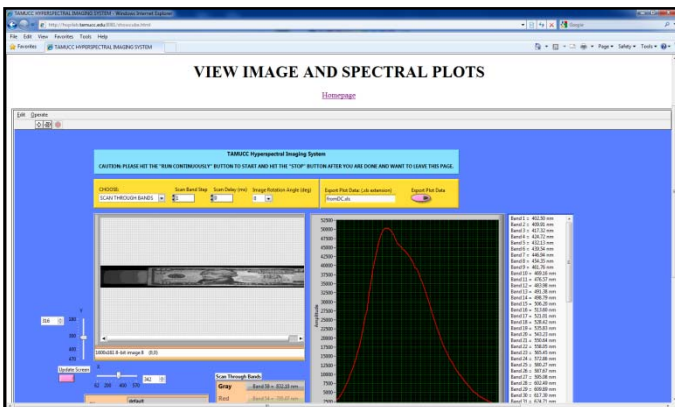


Figure 15. Web-based User Interface of Application 2

The necessary steps for conducting and performing the experiment from the Web interface are summarized below:

- First, username and password for authentication is requested from the user in the Homepage (Figure 12).
- After a successful log on to the session, the user interface for starting the applications of the HIS appears in the Start-up Page (Figure 13).
- The user clicks on the right-mouse button and chooses "Request control" to be granted system control for the maximum amount of time for a single user. If the control is granted, the user can select either of the two applications based on what is to be accomplished in a particular working session. Once an application is chosen, the other application is automatically disabled; the user cannot start the two applications at the same time.
- In the selected application, the user selects the experimental parameters such as "Start position" of the stage, "Scan length" of the image or "Integration time" of the camera. (Figures 14 and 15)
- Since there is only one physical HIS in the laboratory, the Web site allows access to only one user at a time. While one user is performing the experiment, other users may only observe the ongoing experiment simultaneously. During the session, if the user with the active control changes the parameters of the experiment, the other users can watch these changes from their screen.
- Each user is allotted five minutes and the session can be terminated by the user any time or after the five-minute default time limit. The message "The computer [] is controlling the system." is shown on the Web page of users who are on the waiting list. The [] contains the IP address of the computer which is controlling the HIS. When the user finishes the experiment or the default time limit is over, the control is switched to a user who is on top of the waiting list. Priority is assigned on a first-come first-served basis.

In order to switch between the applications or disable/enable the system, the user must go back the Homepage and hit the "Shutdown" ("Refresh" in newer versions) button. When the system is deactivated by the system manager using the central server in the laboratory, nobody can access the experiments. The central server now has the system control. When the system is deactivated, the other users can only observe the experiment which is performed by the system manager.

V. CONCLUSIONS

In this document, a six-module remote control HIS model that covers the hyperspectral imaging process from hyperspectral data acquisition to user management is presented. An application interfacing a lab-based

hyperspectral imaging system with the Internet is implemented based on this model in the HOPI Laboratory at TAMUCC. The developed application allows the system hardware to be controlled remotely. The data can be viewed real-time and on-line by individuals who are part of the research team but unable to access the lab physically, and by other authorized individuals. The development of the LabVIEW interface to achieve real-time remote control of the HIS, the management of the acquired data as well as efforts to secure the HIS system have been explained in detail. The burden of learning the system put on new researchers is reduced significantly when the system is used remotely, as the details of the hardware system is hidden from the remote user. The successful implementation of the remote HIS will not only benefit students and collaborators who want to work with the system remotely, but also STEM outreach activities.

ACKNOWLEDGMENT

The authors thank EnviroBio for its contribution to the optical instrumentation. The HOPI team is grateful to the TAMUCC network security personnel for their help with the logistics of allowing remote access to the HIS through the University's network security protocols.

REFERENCES

- [1] H. A. Basher, and S. A. Isa, "On-campus and Online Virtual Laboratory Experiments with LabVIEW," Southeast Conference, pp. 325-330, 2006.
- [2] Ch. Salzmann, D. Gillet, and P. Huguenin, "Introduction to Realtime Control using LabVIEW with an Application to Distance Learning," International Journal of Engineering Education, pp. 255-272, 2000.
- [3] M. Straatsma, D. Cox, C. Ctistis and R. Bartz, "Development and Enhancement of RLab, A Remote Laboratory System," IEEE, 2009 4th International Conference on Systems and Network Communications, pp. 159-164, 2009.
- [4] E. Fabregas, N. Duro, R. Dormido, S. Dormido-Canto, H. Vargas, S. Dormido, "Virtual and remote experimentation with the Ball and Hoop system," IEEE Conference on Emerging Technologies & Factory Automation (ETFA 2009), pp. 1-8, 2009.
- [5] D. Hercog, B. Gergič, S. Uran, and K. Jazernik, "A DSP-Based Remote Control Laboratory," IEEE Transactions on Industrial Electronics, Vol. 54, No. 6, pp. 3057-3068, Dec. 2007.
- [6] L. Pengfei and N. Luhua, "Remote Control Laboratory Based On LabVIEW," IEEE, 2009 2nd International Conference on Intelligent Computation Technology and Automation, pp. 84-87, 2009.
- [7] K. W. E. Cheng, C. L. Chan, N.C. Cheung, and D. Sutanto, "Virtual Laboratory Development for Teaching Power Electronics," Institute of Electrical and Electronics Engineers, pp. 461-466, 2002.
- [8] C. C. Ko, B. M. Chen, J. Han, S. Hu, V. Ramakrishnan, C.D. Cheng, Y. Zhuang, and J. A. Chen, "Web-Based Virtual Laboratory on a Frequency Modulation Experiment," IEEE Transactions on Systems Man, And Cybernetics, pp. 295-303, 2001.
- [9] C. Xiaoyan, Z. Xiaodong, and C. Xi, "A Virtual Laboratory for Electrical and Electronics Teaching," IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications Proceedings, pp. 491-495, 2005.
- [10] S. Dormido, H. Vargas, J. Sánchez, N. Duro, R. Dormido, S. D. Canto, and F. Esquembre, "Using Web-based laboratories for Control Engineering Education," International Conference on Engineering Education, 2007.
- [11] *LabVIEW User Manual*, National Instruments, Austin, TX, 2010.
- [12] J. P. Kerekes and J. E. Baum, "Hyperspectral Imaging System Modeling," *Lincoln Laboratories Journal*, Vol. 14, No 1, pp. 117-130, 2003.
- [13] J. M. Lerner and L. A. Drake, "Hyperspectral imaging in a LabVIEW environment," *Proc. SPIE*, 3605, 264, 1999; doi:10.1117/12.347571.
- [14] J. A. Vladimir, V. D. Z. John and M. Robert, "Estimation of Estuary Phytoplankton using a Web-based Tool for Visualization of Hyperspectral Images," *aipr*, pp. 25, 35th Applied Imagery and Pattern Recognition Workshop (AIPR'06), 2006.
- [15] D. Krehbiel, R. Zerger and J. K. Piper, "A remote-access LabVIEW-Based laboratory for environmental and ecological science," *Int. J. Eng. Education* 19, 3, 495—502, 2003.
- [16] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "Above the clouds: A Berkeley view of cloud computing," Tech. Rep. UCB/EECS-2009-28, EECS Department, U.C. Berkeley, Feb 2009.