

DATE: May 3, 2001
TO: Hang Li and Dr. Dean Neikirk
FROM: Natthanant Pon Skulkaew and Prem Nainani
SUBJECT: Final report on a PC-based data acquisition and wireless transmission system.

ABSTRACT

The object of our project is to design and realize a system capable of capturing images from an input video stream and then transmitting them via wireless technology to a host server. Our system is PC-based and must allow a user to control image capture settings. Along with captured images, the system transmits its geographical location. To implement this system we used a CCD camera, IMAQ card, wireless LAN, and GPS receiver. We also used LabVIEW 5.0 software to control all hardware and design a simple user interface. Our project is fully functional and meets all design specifications. We hope that in the future our system will be implemented on a ruggedized, portable-PC for field testing purposes.

INTRODUCTION

The goal of our project was to build a portable, PC-based data acquisition and wireless transmission system for use in chemical analysis applications. Our objective was to put together a system which captures images from a video stream and wirelessly transmits them to the Internet, along with a geographical location of where the image was captured. To accomplish this, we used an image capture card to capture images, and a wireless LAN to transmit our data. Though our initial hope was to implement our system on a ruggedized, portable PC, we encountered budget difficulties with the university and instead we constructed a demonstration model for test and evaluation purposes.

MOTIVATION or BACKGROUND

Over the last year, research groups at UT have developed an automated process for collecting image data from chemical sensors known as the “Electric Tongue.” The chemical sensors are micro-machined beads which change color when exposed to certain chemicals [1]. After an image of the sensors is acquired, data is extracted and the image is compared to a database of previously recorded images. By comparison, the identity of an unknown chemical can be deduced. Used in a military setting, these sensors could facilitate early detection of hostile chemicals. Our project has realized a system, by using commercial hardware and software, to acquire sensor data from the “Electric Tongue” and wirelessly transmit the data to a main server.

PROJECT APPROACH

First, we researched the necessary components for our project. We used the internet to explore the availability of products. Next, we evaluated certain specifications of these products such as performance, compatibility, and cost to find the best components for our project. In order to do this, we contacted the product vendors by telephone. Once we chose an optimal system configuration, we purchased the chosen components. After the products arrived, we proceeded to integrate the hardware and install any software required such as an operating system, device drivers, image processing software, and wireless internet programs. Then we began to design a software system that would control image capture, read GPS data from the receiver, automatically transmit images to the Web, and provide the user with a simple-to-use interface. It was also necessary to design a method to display the transmitted data. We designed a process so that a website is updated automatically. Finally, we tested and evaluated the performance of this system. Appendix A contains our project flow diagram which outlines the steps associated with our project.

PROBLEM DESCRIPTION

As previously stated, the goal of this project was to construct a portable, PC-based data acquisition and wireless transmission system. The PC should capture images from an input video stream in order for them to be analyzed. The video will either be in Super-Video (S-Video) format or in the National Television Standards Committee (NTSC) format. After analysis, the PC must transmit the image and its accompanying data, along with a geographical location of where the video/image was taken, via a wireless communications channel. Any image or data processing done before transmission is not part of the project. Our system was to operate in two modes: single capture and continuous capture. In single capture mode, a user can capture images whenever he/she desires by simply clicking a button. In continuous capture mode, the user can specify the capture rate, and the system automatically captures images at the given rate. In both modes the system is to automatically transmit data to a server where images can be viewed on a website. Our system also required a software environment to control the functions of these different devices. Additionally, a Graphical User Interface (GUI) was

required for the user to specify capture and transmission settings. To see a block diagram outlining the algorithm of our system refer to Appendix B.

HARDWARE SOLUTION

We spent approximately three weeks researching and comparing hardware components and vendors. After the initial research phase, we chose to use a Delorme Earthmate GPS [2] receiver and a Novatel Merlin Cellular Digital Packet Data (CDPD) wireless modem [3]. Dr. Neikirk also provided us with a National Instruments IMAQ (Image Acquisition) card. We discussed our component choices with Dr. Neikirk, and he suggested we use a wireless local area network to transmit our data, as CDPD wireless service is quite costly. Keeping this in mind, we decided to use a 3Com Wireless Home Gateway and a 3Com Wireless adapter [4]. We also purchased a PC card reader so we could use the wireless adapter on a desktop or on a portable PC. We tested each piece of hardware individually and found that they worked properly. Figure 1 shows an overall system configuration.

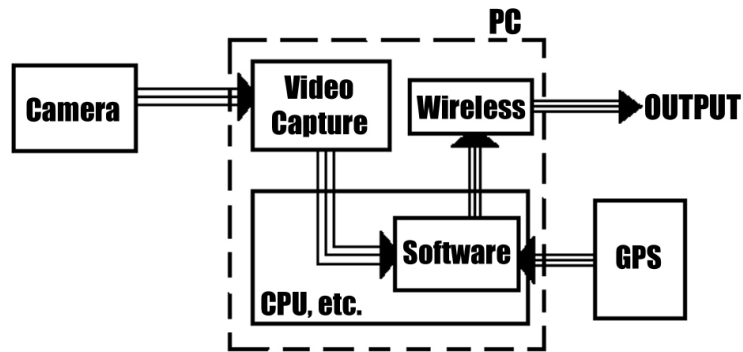


Figure 1. System Overview

Installation of the IMAQ card and the wireless components proved to be relatively simple. We then connected a camera to the S-Video input of the IMAQ card and began receiving images. The GPS receiver, however, proved to be more of a problem. We plugged the receiver into the RS232 (serial) port of the PC, and we used the Windows 2000 Hyperterminal application to interface with the port. We found that the incoming data was encoded and therefore unreadable. We proceeded to contact the manufacturer and were unable to get a decoder. To solve this problem, we ordered the GPS-12, from Garmin [5], which outputs data in the standard National Marine Educators Association (NMEA) format [6]. Once we obtained the receiver, we tested it

and were able to view and identify the appropriate data strings which would give us position and time.

SOFTWARE SOLUTION

Once our hardware was properly installed and configured, we began working on the software which would interface all the hardware components. We decided to use LabVIEW 5.0 G programming language to design the software. Neither of us had much LabVIEW experience, so we used the reference manual [7] to help us. Appendix B is a flow chart of the software algorithms we designed.

Our software package had to capture an image, transmit the image, and append GPS data to the image. The GPS data desired was the time and location. We found it would be most efficient to name our image files based on the GPS data. The first Virtual Instrument (VI) we coded in LabVIEW was the image capture function. This VI simply captures an image from the incoming video stream and saves it as an image file. For the GPS VI, we used a simple serial-port read command to read 900 bytes from the GPS receiver, which was connected to the serial port. Within these 900 bytes is a string which contains all the information we needed. We devised an algorithm which searches for the appropriate tag and adds the proper characters to make the data understandable. We found a NASA website [6] which showed how to decode NMEA data strings, and we developed our algorithm based on the method shown on this NASA website.

The file transmission VI we developed used the File Transfer Protocol (FTP) to send the image to a server. HTML code on the server displayed the most current image on a webpage and also archived the sent image. Once we completed all these VIs, we integrated them into a control structure which allows the user to choose between single or continuous capture. The GUI was the last item we designed. Through the GUI, a user would be able to specify the capture method and the server to where the images are to be transmitted.

TEST, EVALUATION, AND RESULTS

In order to test and evaluate our system, we came up with a simple plan to test functionality. We verified that the single capture function worked by running the software and capturing at random

intervals. Clicking the capture button once should result in one image capture. To test continuous capture, we set the capture interval at 1 image per minute, and let the system run for 3 minutes. We ran this test several times to ensure consistency. The only aspect of the system we wished to examine was the capture rate. The specifications required that the system be able to capture at a rate of at most 1 image per minute.

The results obtained from the testing and evaluation phase were encouraging. We achieved a 95% success rate for single captures and a 98% success rate for continuous capture. The reason why the system did not perform at 100% was due to poor GPS coverage. We experienced poor GPS coverage during some tests and found that this creates an error condition. We quickly remedied this by adding an error handler in the GPS VI. The system is now capable of capture and transmission independent of GPS coverage. We were also pleased that the capture rate exceeds specifications. Each capture and transmit sequence takes roughly 15 seconds, therefore the system can successfully capture and transmit an image every 25 seconds.

CONCLUSION

Our system met all of the specifications. The capture rate is faster than required, and the system outputs the GPS data as part of the image file name. We were also able to account for any GPS coverage issues. We feel that the software is robust enough to withstand the demands of data collection in the field. Although the system was designed according to the demonstration unit specifications, our software will function on any PC with LabVIEW and an IMAQ card, so ruggedized implementation should not be a problem.

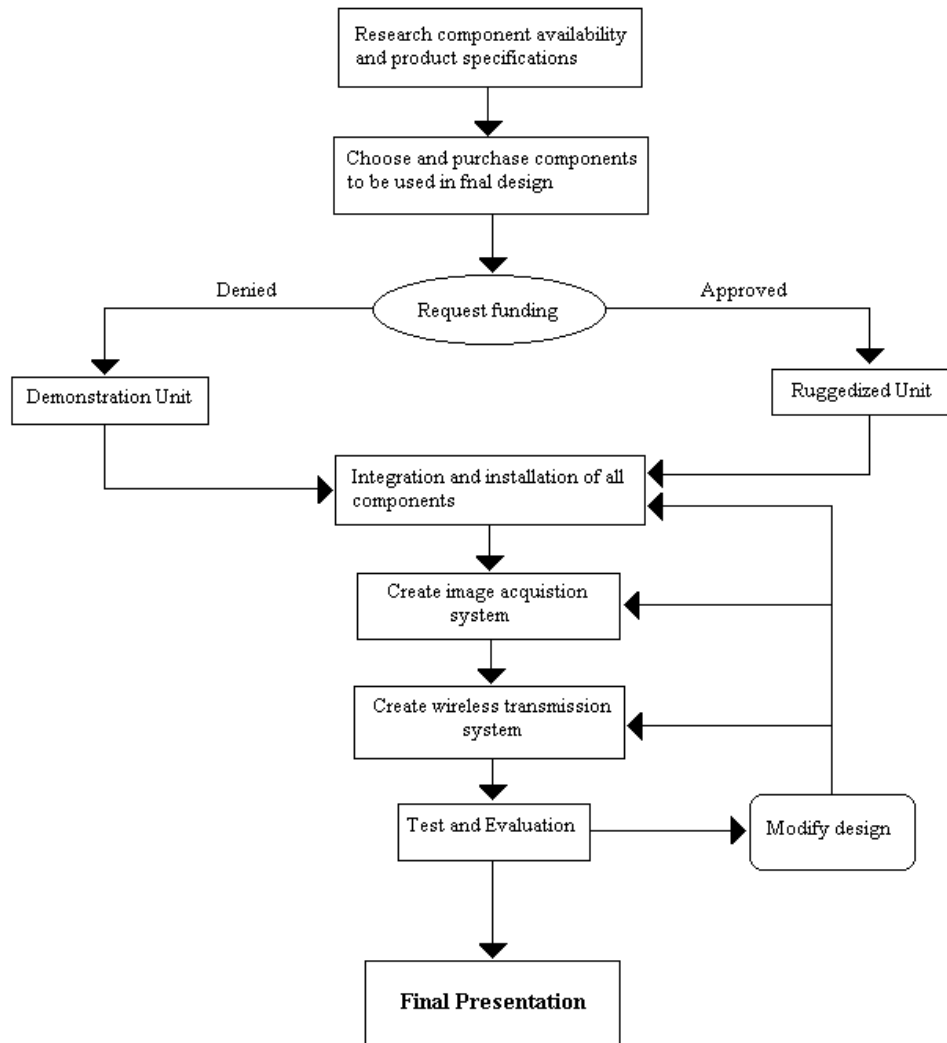
When we first began work on this project, we did not realize that the LabVIEW G programming language was so powerful. We learned how to code in the language over the course of the project, and we found that it is an excellent tool for interfacing hardware. The approach we took towards the project proved successful, as we did not have any setbacks. Focusing on one part of the design at a time helped simplify the coding, and also made the overall course of the project much smoother. We are pleased with our project, as it meets and exceeds specifications. By using our system in a field setting, researchers in the lab can view and analyze data collected in the field seconds after it is gathered. This may eliminate the need

for costly field setups and equipment. All in all, this project was a wonderful learning experience as well as a good engineering challenge.

REFERENCES

- [1] “AutoImage Summary,”
<http://weewave.mer.utexas.edu/DPN_files/courses/464H/imaging_Sp99/AutoImage%20Summary/Summary.htm> Accessed Jan. 28, 2001.
- [2] “Delorme: Earthmate GPS Receiver Overview,” <www.delorme.com/earthmate/>
Accessed February 5, 2001.
- [3] “Novatel Wireless PC Cards, Modems for Laptops,”
<<http://www.novatelwireless.com/pcproducts/index.html>> Accessed February 1, 2001.
- [4] “3Com Wireless LANs – Wireless for the Home,”
<http://www.3com.com/mobile/wireless/wlan_forhome.html> Accessed February 25, 2001.
- [5] “Garmin: GPS 12,” <<http://www.garmin.com/products/gps12/>> Accessed March 3, 2001.
- [6] “MAEL-GPS NMEA Table,”
<<http://www.grc.nasa.gov/WWW/MAEL/ag/agnmea1.htm>> Accessed March 22, 2001.
- [7] *LabVIEW: Function and VI Reference Manual*. National Instruments, Austin, Texas, 1998.

APPENDIX A: Project Flow Diagram



APPENDIX B: Flow Chart of Software Algorithms

