

The design and implementation of a computer interface for a Raman spectrometer using the LabVIEW software package

Dan M. Burger School of Engineering, Vanderbilt University

A user-friendly computer interface was created for the SPEX 1403 Raman spectrometer using the LabVIEW computer package. The interface allows the user to align the spectrometer easily, produce a graph of the Raman spectra, and save the file to a format readable by a spreadsheet program. The program has three screens: the first allows the user to set initial settings, the second allows the user to read from a particular wavelength for proper alignment, and the third allows the user to set scan settings and perform the scan. The interface was successful in reading Raman spectra easily and efficiently.

Introduction

Raman spectroscopy is the scientific analysis of light scattered by a material. When light shines on a particle of matter, most of the light that scatters retains its original wavelength and is characterized as Rayleigh scattered light; a very small amount of the light, however, produces a shift in wavelength and is referred to as Raman scattered light.¹ The shifts in wavelength correspond to the atomic composition and bonds present in the sample material, and by producing a graph of the intensity of the Raman scattered light over certain wavelengths, this information can be extracted from the sample.

Unlike other methods of material analysis, Raman spectroscopy does not require surface contact, destruction, or extensive preparation of the sample. In addition, Raman spectroscopy provides minimal interference with water and glass for easy handling of these samples, and the sample can be in a solid, liquid, gaseous, or solution state. Additionally, results can be obtained relatively quickly.¹

The most common applications for Raman spectroscopy are the production and discovery of pharmaceuticals, semiconductors, and carbon nanotubes, as well as geological and biological analysis. Raman spectrometers have also been used in forensics, polymer research, art analysis, and archaeology.¹

Normally, a Raman spectrometer or any other piece of scientific hardware comes with proprietary software to operate the system. Such software is very limited and does not meet the exact specifications of its customer. The original software for this spectrometer, for instance, was designed for an

MS-DOS system and relies on a less intuitive interface. Most scientific equipment can be operated in software created with the LabVIEW programming interface provided by National Instruments². LabVIEW, originally created in 1986 for the Apple Macintosh, has been used in areas as diverse as oceanography, product testing, robotic automation, and manufacturing: it allows its users to build software interfaces for scientific equipment with an exceptional level of customization.^{2,3}

In this paper, I will describe a software interface created using LabVIEW for a Raman spectrometer that is user-friendly and provides features not available in proprietary software.

Hardware and software specifications

The Raman spectrometer used is a SPEX 1403 0.85m double spectrometer, as shown in Figure 1. It receives an input in wavenumbers, which is the reciprocal of wavelength and is typically measured in cm^{-1} :

$$\text{wavenumber (cm}^{-1}\text{)} = \lambda^{-1} \quad (1)$$

A stepper motor inside the spectrometer rotates the two diffraction gratings inside the spectrometer so that it only collects light at that particular wavenumber. The term “position of the spectrometer” is used herein to describe the wavenumber that the diffraction gratings allow to enter the photomultiplier tube. An analog counter on the spectrometer displays this current position.

The light input of the spectrometer is a Coherent Innova 90 scientific argon-ion laser which produces light at 19436 cm^{-1} (green) and 20492 cm^{-1} (blue). An auxiliary laser in a nearby room can also



Figure 1. A photograph of the SPEX 1403 Raman spectrometer (top of picture) and its accompanying equipment below. The computer on the lower left has the interface installed and is connected to the equipment. The small box on the lower right supplies power to the photomultiplier tube. In front of the spectrometer is the driver for the stepper motor; one of the lasers that supply monochromatic light is behind the spectrometer.

beam light into the spectrometer through a hole in the wall. The spectrometer data is collected by a photomultiplier tube (PMT), which converts the photons into analog pulses. A preamp then converts this data into digital pulses that can be collected by the data acquisition device.

The computer is an IBM compatible machine with an Intel Celeron 1.40GHz CPU and 384MB of RAM. It has the Windows XP Service Pack 2 operating system and LabVIEW version 8.0.2 (20th Anniversary Edition) installed on the hard drive. It is connected to the stepper motor with a PCI-Step-40X motion driver and to the preamp with a PCI-6602 data acquisition device, both of which are supplied by National Instruments.

Two packages of LabVIEW modules are provided for each of the drivers. The ValueMotion modules allow LabVIEW to send commands to the motion driver to move the stepper motor to a certain wavenumber. Upon receiving its command, the stepper motor will accelerate in a specified time interval (in seconds) to a certain velocity (in wavenumbers per second), move toward a specified position, and decelerate using the same acceleration time interval. It is also able to determine its current position, retrieve diagnostic settings, and open and close an internal shutter that blocks intense Rayleigh scattered light from entering the photomultiplier tube.

The DAQmx modules allow the user to access incoming data from the data acquisition device. The data is received in the form of an unsigned 32-bit integer that continuously increases based on how many electrical pulses the data acquisition device receives. The rate of increase of the raw data is the intensity of light entering the photomultiplier tube.

Design and implementation

Design procedures

Prior to constructing the computer interface, specifications for the program were discussed and the core program was built. A set of goals were established for the program: it must be user-friendly, its results should be accurate, the equipment should be protected from harm, and its code should be easy to read. After the core program was built, the program underwent a series of revisions to eliminate errors, protect the equipment from damage, and provide further aid to the user. Some of these revisions came out of necessity; others came as a result of discussions with faculty who are expected to use the program. For example, earlier versions of the program only accepted the position value in absolute coordinates (e.g. move to 20000 cm^{-1}). After testing the program, it became apparent that the program should also accept relative coordinates as well (e.g. move 200 cm^{-1} above the current position).

Design

The program consists of three screens (Figure 2), each of which may be moved back and forth in the standard “wizard” format used in many installation programs.

Initialization screen

The first screen is used to set the initial settings for the spectrometer. There are three input fields on this screen. The first input takes in the analog counter on the spectrometer, which is used to calibrate the spectrometer with the motion drivers. An additional “nudge” button moves the spectrometer slightly for better calibration. The second and third inputs are for the wavenumber of the laser and the safety shutter setting (usually 20 cm^{-1}). If the current position of the spectrometer is within the safety shutter setting of the laser wavenumber, the shutter will close. This prevents the photomultiplier tube from being saturated by the strong intensity of the laser light.

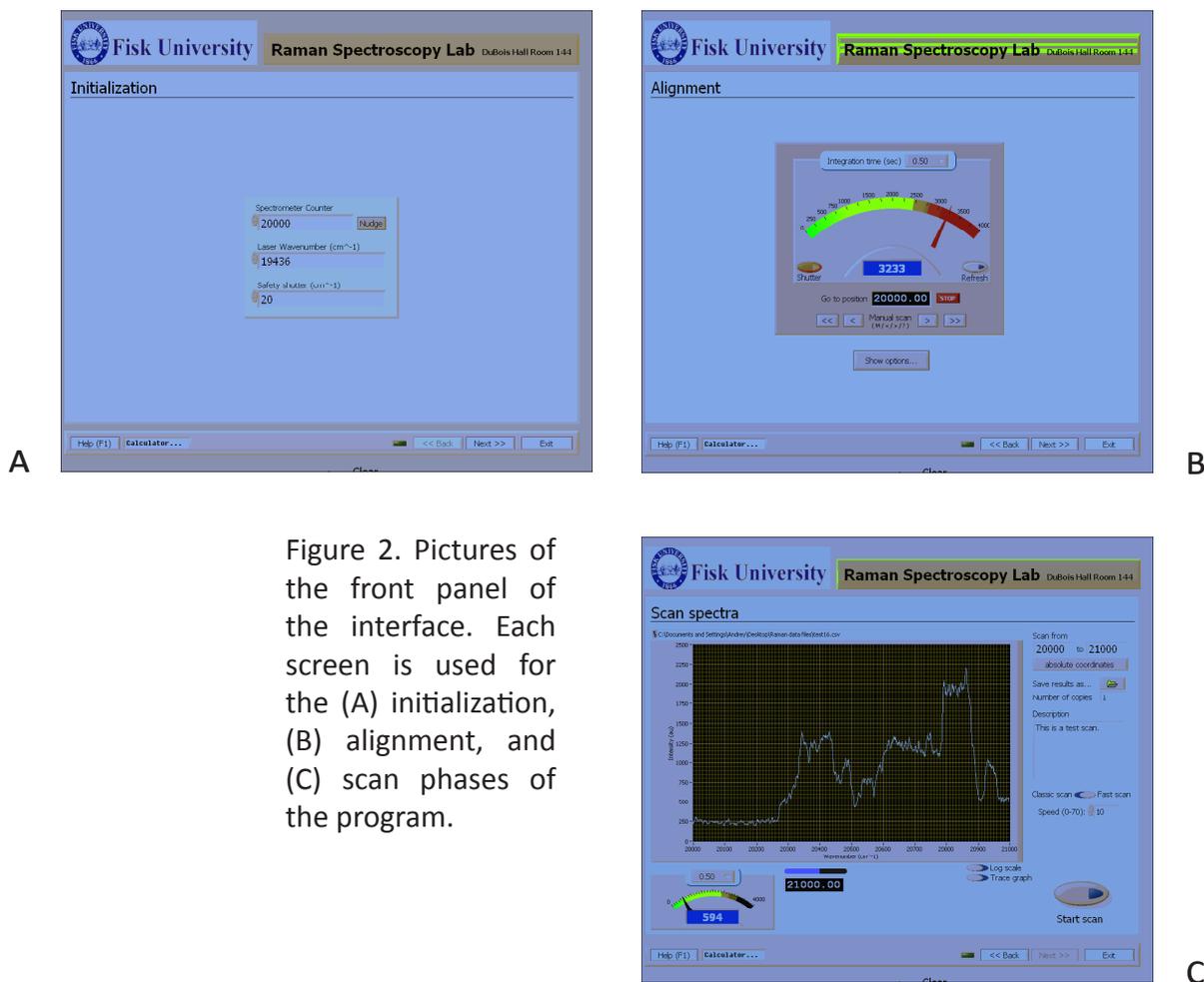


Figure 2. Pictures of the front panel of the interface. Each screen is used for the (A) initialization, (B) alignment, and (C) scan phases of the program.

Alignment screen

The second screen is for the user to align the spectrometer to ensure optimal accuracy. The intensity of the light entering the photomultiplier tube is calculated by subtracting the raw data from the DAQmx module at the current time from what the raw data was at a previously specified time interval. This time interval is referred to as the integration time, which can be set anywhere from 0.05 to 5.00 seconds. It is recommended that darker spectra use greater integration times. An interval that is too low is less accurate and more erratic; an interval that is too high is slower and not as responsive. The default integration time is 0.50 seconds. The result is referred to as the number of counts, which is the intensity of light entering the photomultiplier tube in arbitrary units. This value is usually calculated continuously; the user may prefer instead to use an alternative method to calculate this value only once per integration time.

The current position may also be modified on this screen in one of two ways. One way is to type the position into the position field and press Enter; the position will move automatically to that posi-

tion using the ValueMotion module. Alternatively, the user may press the four manual scan buttons that will increment and decrement the current position in two speeds. The scan buttons can be operated with the mouse or the keyboard.

Usually, the Raman spectroscopist moves to a position where a high intensity of Raman scattered light is expected. The manual scan is used to maximize the number of counts. Afterwards, the spectrometer is adjusted so that the number of counts is further maximized. A more accurate scan is produced with a high number of counts.

The internal shutter may also be opened or closed in this screen. In addition, the alignment screen has an options panel that allows the user to set the velocity and acceleration of the stepper motor and switch to the alternate method of integration.

Scan spectra screen

The third screen is used to create a waveform graph of the Raman spectrum. During a scan, the computer automatically moves 50 cm^{-1} beyond the starting position before moving to the starting

position itself. This serves to prevent backlash, a problem caused by reversing the spin direction of a motor that reduces the accuracy of the spectrometer position until it rotates a specified amount in the new direction.⁴ Afterwards, the spectrometer moves towards its ending position, collecting data along the way and plotting the data on the graph. When the ending position is reached, the program saves the data to a spreadsheet file and resumes its normal state.

There are two types of scans. The “fast scan” collects the data as the stepper motor is running. This mode is useful when a quick survey is needed and usually lasts around five minutes. The “classic scan” moves the motor in user-specified incremental positions, stops the motor, and then collects the data for the integration time. This process is repeated until the scan is complete. Since the classic scan provides the best accuracy, it is used in most cases. A classic scan usually takes between 30-60 minutes.

The scan spectra screen displays the graph on the screen in either linear or semi-logarithmic format, in addition to showing the number of counts and the current position.

The user is able to set the starting position, ending position, type of scan, and a brief description of the scan that is appended to the spreadsheet file. The starting and ending positions may be provided either in absolute coordinates or relative to the laser wavenumber, and the computer will recognize this automatically. A filename for the spreadsheet of the scan is optional at this point; if no filename is specified right away, the computer will ask for one once the scan is finished. The user can also specify the wavenumber increment for the classic scan or the speed of the fast scan. Finally, the user presses a button on the screen to start the scan.

During the scan, a display shows the amount of progress made and the estimated time remaining in the scan. The user may abort the

scan at any time.

After the scan has completed, the data in the waveform graph is saved to a file in a comma-separated value (CSV) format, a universal format that is readable in Microsoft Excel, OpenOffice.org Calc, and virtually all spreadsheet programs. The scan data can then be plotted to a graph, as shown in Figure 3. The CSV file contains the date and time of the scan, the integration time, and the description of the scan.

At this point, the user may wish to find the coordinates of the peaks of the Raman spectra. This is possible using the trace feature, which displays a set of crosshairs on the graph that the user may drag to the peaks. The wavenumber and intensity of that point are then displayed.

Safety features

The safety shutter feature is only one of many safety features designed to protect the equipment. Even with the safety shutter feature it is possible for the photomultiplier tube to be bombarded with photons; for instance, sunlight may leak into the spectrometer. If the intensity of light exceeds one million counts per second, the shutter will close and a dialog box will pop up.

A few safety features exist for the stepper motor as well. The stepper motor can only function at a range of 10000 to 30000 cm^{-1} , beyond which the equipment may be damaged. To prevent this from occurring, a pair of software limits is set at 10500 and 29500 cm^{-1} . If the user goes beyond these limits, the motor will automatically stop and inform the user to move away from these limits. In addition, there is a pair of limits provided by the hardware at

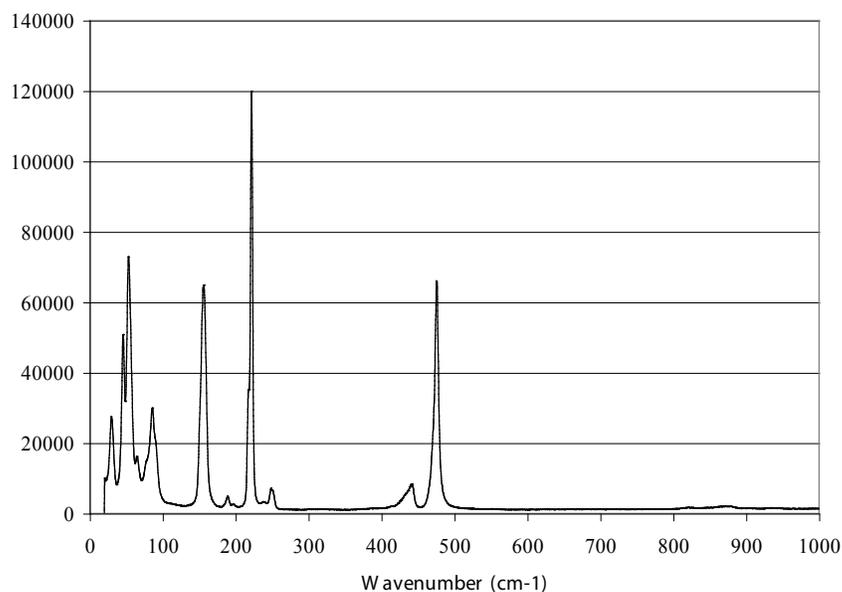


Figure 3. Raman spectrum of solid sulfur created with the new software interface. The scan was performed in classic scan mode with an integration time of 0.500 sec and a wavenumber increment of 0.5 cm^{-1} .

approximately 10240 and 29760 cm^{-1} that will lock up the motor entirely. This can be later restored using the diagnostic tool that is provided with the software interface.

Other features

The software informs the user if the equipment is turned off at the start of the program so that the user can turn the equipment on.

A simple calculator is provided at the bottom of the screen, where the user may type in a calculation and press Enter to view the solution. The calculator accepts arithmetic, logarithmic, exponential, and trigonometric functions as well as some constants.

A help feature is provided to aid users in the Raman scanning process. In addition to providing instructions, the user is able to point at features on the screen to learn about them.

The software interface also comes with a diagnostic tool. This is a collection of ValueMotion and DAQmx utilities that communicate with the motion driver and data acquisition device allowing the user to diagnose problems with the equipment.

Coding

A LabVIEW program (also known as a virtual instrument) consists of two parts. The front panel consists of all the components that the user will see. These components are simply placed onto the screen. The other part of the program, the block diagram, is the code that the program will run. Unlike most other programming interfaces, the block diagram uses graphics to describe the flow of data. It consists of hundreds of types of functions, each of which accepts data on the left side and transmits data on the other side. These functions are linked together with wires that are color-coded to indicate the type of data being transmitted (integer, floating

point number, boolean, string, etc.). Each control on the front panel receives a corresponding function on the block diagram, which allows the program to receive data from the controls and send data to the indicators. A sample of the block diagram used for this program is shown in Figure 4.

The most difficult challenge was making the program simple and descriptive enough such that modifications could easily be made and a person with LabVIEW experience could understand the program. Eventually, as more features were added, the program inevitably became increasingly complex. Nevertheless, the block diagram for the program is very descriptive and relatively easy to modify; therefore it has met its goal.

The top of the program describes events that will happen at the start of the program, such as initialization of the controller and loading of settings from the disk. Just below are the mechanisms that calculate the number of counts and plot the data to the graph during a fast scan. Since the timing of these tasks is different, they are separate from the rest of the code.

The rest of the code is placed in a loop that repeats itself every 0.05 seconds until the program is finished. This makes all of the features available while the program is running. The top of the loop allows the user to switch back and forth between pages and sets up certain elementary tasks (open shutter, close shutter, move to position, etc.) that can be executed anywhere within the program. Further down is the code for calculating the current position and for operating the alignment screen, including the manual scan buttons. This code is followed by the safety features and the code for plotting the Raman spectra. Miscellaneous code has been placed at the end.

The program uses several methods to keep the program organized. One is to access the data using property nodes, which are special functions that can change the value of any object. The other is the enumerator data type, which can hold multiple sets of text, one of which is activated at a time. Since the enumerator can be set up to perform actions based on its value, it is particularly useful in building tasks. For example, the manual scan enumerator has five sets of text: “fast increment”, “slow increment”, “slow decrement”, “fast decrement”, and “stop

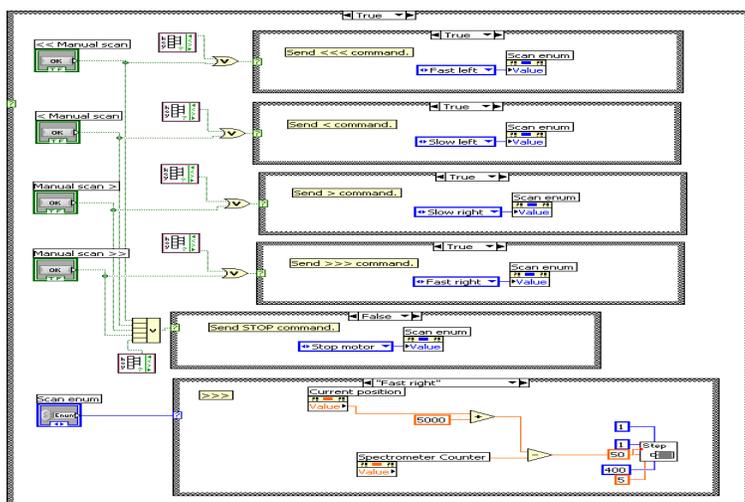


Figure 4. A portion of the LabVIEW block diagram for the software interface that handles keyboard operation of the manual scan buttons.

motor". Setting the active set of text will perform the specified result.

Applications of the program

This program will be available to many undergraduate and graduate researchers as well as faculty at Fisk University to conduct various experiments for many years to come. Since Raman spectroscopy is very useful in finding the chemical composition and structure of a sample, plenty of demand of the program is expected for use in experiments.

The most common use for the interface will be to investigate the properties of special glass compositions. The interface has already been used to investigate random lasing action in glass ceramics⁵. It was expected that stimulating erbium-doped glass ceramics with a random laser would produce monochromatic light in all directions. The Raman spectroscopy interface was able to produce reliable scans that were saved to disk and later analyzed using a spreadsheet program. Although the experiment was unsuccessful in finding such scattering, it was an invaluable aid in investigating this type of scan.

Conclusion

The construction of the program was a success and it is fully functional. Based on the result of the project, it is possible for a computer user to produce a software interface for a Raman spectrometer using the LabVIEW software package. Some of the advantages of this software interface include an easy-to-learn interface, customizable settings for alignment and scanning, continuous calculation of the number of counts, fast scanning, spectra analysis, and multiple scans, as well as exporting data in the CSV spreadsheet format with descriptive information. This software interface supports the fact that LabVIEW can be used to produce software for scientific equipment that exceeds the boundaries of proprietary software and is customized with the user's needs in mind.

Program availability

A copy of this software interface is available as a free download on the World Wide Web at the following address: <<http://www.fisk.edu/~aburger/042608ri0725.zip>>. This ZIP file contains the LabVIEW program as well as the diagnostic tool and any dependent files. Information on software requirements are provided in the file.

Acknowledgements

This paper has been submitted as a partial requirement for the Fisk University Research Experience for Undergraduates / Summer Research Program (REU/SRP) in the summer of 2007. This project has been funded by NSF-CREST grant #HRD-0420516 and NSF-REU grant #DMR0453562. The author wishes to thank Dr. Steven Morgan, Dr. Zhengda Pan, Michael Northington V, Kemar James, Dr. Kent Wallace, Tiffany Crenshaw, and Dr. Arnold Burger.

References

- ¹"Horiba Jobin Yvon." <http://www.jobinyvon.com/usadivisions/Raman/index.htm> (accessed 3 Jun 2007).
- ²"LabVIEW: 20 Years of Innovation." http://www.ni.com/labview/presskit_awards.htm (accessed 13 Jun 2007).
- ³"LabVIEW Customer Solutions." http://sine.ni.com/csol/cds/list/vw/p/qry/prd/id/28/srt/dt_d/nid/124100 (accessed 13 Jun 2007).
- ⁴Adler, Michael. "Meccano Frontlash Mechanism." <http://web4.actcom.co.il/meccano/frontlash.html> (accessed 22 Apr 2008).
- ⁵Northington V, Michael. "Random Lasing Properties of Oxyfluoride Glass-Ceramics with Increased Er³⁺ and Different Heat Treatments." (2007)