ELSEVIER

Journal of Neuroscience Methods xxx (2006) xxx–xxx

JOURNAL OF NEUROSCIENCE METHODS

www.elsevier.com/locate/jneumeth

MPScope: A versatile software suite for multiphoton microscopy

Quoc-Thang Nguyen *, Philbert S. Tsai, David Kleinfeld

Physics Department, University of California, San Diego, USA

Received 21 December 2005; received in revised form 22 February 2006; accepted 1 March 2006

Abstract

MPScope is a software suite to control and analyze data from custom-built multiphoton laser scanning fluorescence microscopes. The acquisition program MPScan acquires, displays and stores movies, linescans, image stacks or arbitrary regions from up to four imaging channels and up to two analog inputs, while plotting the intensity of regions of interest in real-time. Bidirectional linescans allow 256 × 256 pixel frames to be acquired at up to 10 fps with typical galvanometric scanners. A fast stack mode combines movie acquisition with continuous z-focus motion and adjustment of laser intensity for constant image brightness. Fast stacks can be automated by custom programs running in an integrated scripting environment, allowing a 1 mm^3 cortical volume to be sampled in 1 billion voxels in approximately 1 h. The analysis program MPView allows viewing of stored frames, projections, automatic detection of cells and plotting of their average intensity across frames, direct frame transfer to Matlab, AVI movie creation and file export to ImageJ. The combination of optimized code, multithreading and COM (Common Object Model) technologies enables MPScope to fully take advantage of custom-built two-photon microscopes and to simplify their realization.

Keywords: Femtosecond laser; Two-photon; Electrophysiology; Software; Scripting; Calcium imaging; Histology

1. Introduction

Multiphoton microscopy (Denk et al., 1990) was first applied to neurobiology more than a decade ago (Denk et al., 1994) and has since acquired a prominent status among neuroscience methods (for a review, see Nguyen et al., in press). The advantages of multiphoton laser scanning microscopes (MPLSMs) include deep penetration (100–1000 μm) into highly scattering neural tissues (Helmchen and Denk, 2005), micron to sub-micron resolution and optical z-sectioning, as well as reduction of fluorophore bleaching and/or phototoxic damages.

Custom-built MPLSMs (e.g. Fan et al., 1999; Mainen et al., 1999; Majewska et al., 2000; Nguyen et al., 2001; Nikolenko et al., 2003; Roorda et al., 2004; Tan et al., 1999; Tsai et al., 2002) are not only considerably cheaper than commercial microscopes (by almost a factor of two) but can also be specifically designed for a particular purpose such as in vivo recordings (Tsai et al., 2002). One limitation of this approach is that programs controlling custom MPLSMs (e.g. Nguyen et al., 2001; Pologruto et al., 2003; Tsai et al., 2002) incorporate only a certain number of features from an ideal "wish list" of requirements for neuroscience.

The most important feature is a high frame rate that will enable the capture of fast functional events such as variations in intracellular calcium concentrations as well as the possibility to provide a near real-time assessment of cellular responses as the experiment unfolds. This can be achieved by first drawing areas of interest encompassing select neurons and then by having the program plot during the acquisition session the average pixel intensity of these areas for each frame as a time series in an oscilloscope-like display. Also, simultaneous frame acquisition, display and disk streaming is of considerable importance and thus precludes the use of a "blind" mode where frames are captured and stored without any form of visual feedback.

Further, the need to coordinate electrophysiological recordings and image acquisition is important for many functional experiments. One solution is to centralize the control of most electrophysiological operations in the scanning program to ensure proper synchronization of the analog and video streams.

Another requirement of MPLSM systems is the ability to generate, magnify and rotate laser scan patterns in order to bring the scanning field into a favorable orientation to image

* Corresponding author at: Urey Hall Room 7108, University of California at San Diego, 9500 Gilman Drive, La Jolla, CA 92093-0374, USA.
Tel.: +1 949 246 8143; fax: +1 858 534 7697.
E-mail address: qnguyen@physics.ucsd.edu (Q.-T. Nguyen).

0165-0270/$ – see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.jneumeth.2006.03.001

particular structures or processes such as dendrites or blood vessels. This feature is critical when measuring cerebral blood flow (Kleinfeld et al., 1998), since the method depends on aligning linescans on top of linear portions of blood vessels. Some multiphoton microscopes (e.g. Tsai et al., 2002) depend on a custom-designed electronic circuit based on an embedded DSP or PIC processor to rotate and magnify the scan pattern. These devices increase the overall complexity of the microscope and are difficult to replicate, maintain and upgrade. The phenomenal computing power now available in virtually all current personal computers should, in principle, be largely sufficient to allow the MPLSM program to undertake the functions of these devices.

Perhaps the most difficult requirement for MPLSM software is the possibility to use the same program in widely different experiments ranging from time-lapse imaging (Ruthazer and Cline, 2002) to histology (Tsai et al., 2003). MPLSM programs can, in theory, be customized by modifying their source code. However, this solution can challenging even when the programs are written in the Matlab (MathWorks, Natick, MA) or Labview (National Instruments, Austin, TX) programming languages. As an alternative, customization and automation of repetitive tasks can be achieved by providing users with the possibility to write small programs called scripts, usually in an easy to learn programming language like Microsoft VBA (Visual Basic for Application) or VBScript (Visual Basic Scripting Edition). This approach is adopted in the commercial programs LaserSharp and LSM that control, respectively, the Radiance2100 MP from Bio-Rad (now Carl Zeiss CellScience) and the LSM 510 NLO (Zeiss) multiphoton systems, and also in the custom-build system described by Nikolenko et al. (2003). Scripting based on the Microsoft ActiveX standard has the advantage of allowing users to include in their scripts some of the many ActiveX software components already written or to tap into functionalities present in existing programs, thus creating script-based "meta-applications" (Nguyen and Miledi, 2003).

In this paper we present MPScope, a suite of turnkey applications that fulfills the requirements outlined above. MPScan, the frame acquisition program, is flexible enough to control any MPLSM based on generic scanning and data acquisition hardware. MPView is designed to analyze the data generated by MPScan. We describe the design principles we adopted for MPScope and present benchmark data based on a novel imaging scheme allowing deep *z*-stacks to be taken almost up to an order of magnitude faster than with a previous program written in LabView.

2. Material and methods

2.1. Software development

MPScope was developed in Object Pascal using the Delphi 5 programming environment (Borland Corp., Scotts Valley, CA) on a Dell Inspiron 8000 laptop (Dell Corp., Austin, TX). MPScan ran on a dual 1 GHz processor Dell workstation with 1 GB RAM running Microsoft Windows 2000 Professional Edition (Microsoft Corp., Redmond, WA).

2.2. Software and hardware requirements

MPScope is designed for IBM-compatible personal computers running the Microsoft Windows 2000 or XP operating systems. As stand-alone compiled executables, MPScan and MPView do not depend on a run-time environment. MPScan requires National Instruments PCI-6110E card and the Traditional NI-DAQ library (v. 6.9 or higher), which is automatically installed when setting up the data acquisition board.

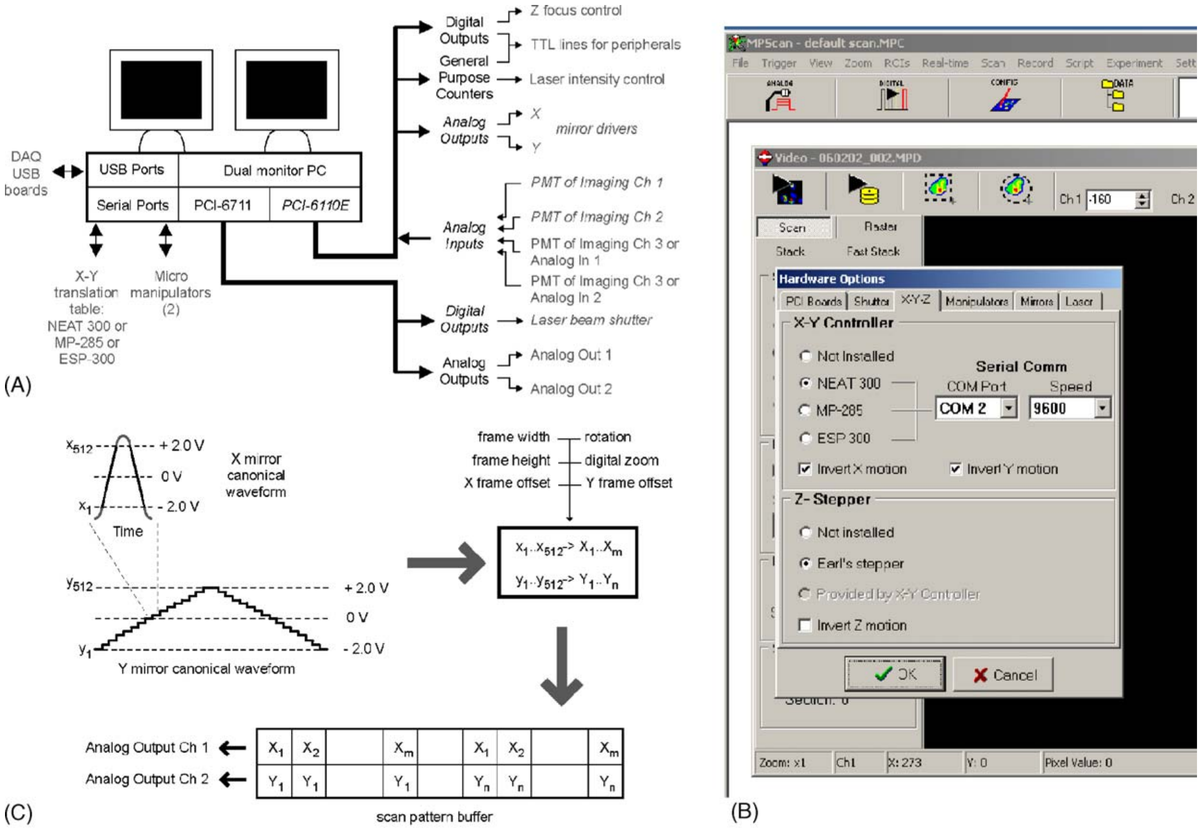
2.3. Multiphoton microscope

MPScan was tested on a femtosecond laser scanning fluorescence microscope instrument consisting of a Mira-Verdi pump-femtosecond laser combination (Coherent, Santa Clara, CA), fluorescence optics and detection electronics (Tsai et al., 2002). Fluorescence photons are detected by up to four imaging channels equipped with photomultipliers (PMTs) (Model R6357, Hamamatsu, Bridgewater, NJ). The PMT output currents are converted into voltages by locally designed feedback resistor-based preamplifiers. The resulting voltages are further amplified and either low-pass filtered or integrated by a custom four-channel pixel integrator. The integration time of this device is set by the analog conversion clock of the data acquisition board.

2.4. MPScan: hardware interfacing

MPScan was designed to be compatible with custom-made multiphoton microscopes such as the Denk/Svoboda design described in Pologruto et al. (2003) and the instrument described in Tsai et al. (2002). The program only requires that the scanning system consist of a pair of orthogonally mounted, servo-controlled galvanometric mirrors. The deflection of the laser beam, set by the angle of each mirror, is directly controlled by the analog outputs of the main data acquisition board (Fig. 1(A)). Our microscope and the one described in Pologruto et al. (2003) use two Model 6210 galvanometric mirrors controlled by MicroMax 670 servos (Cambridge Technologies, Cambridge, MA). The maximal voltage output from the board can be set by the user in MPScan to use galvanometric mirrors from other vendors that have different input voltage requirements (e.g. those from GSI Lumonics).

The PCI-6110E data acquisition board is also used to digitize the signals from up to four imaging channels. In case analog signals (e.g. electrophysiological data) need to be recorded simultaneously, the last two imaging channels can be used instead to sample analog data (Fig. 1(A)). An additional, optional PCI analog output board (National Instruments PCI-6711) can be controlled by MPScan. The board provides two analog output channels which can be used to control additional devices such as a Pockels cell to modulate the laser beam intensity (e.g. Nikolenko et al., 2003) or to generate stimulation waveforms for cellular amplifiers. The laser shutter can be turned on or off either by a digital output line from one of the PCI boards. Additional data acquisition boards can be



X-Y Controller

○ Not Installed
● NEAT 300
○ MP-285
○ ESP 300

Serial Comm: COM Port COM 2, Speed 9600

☑ Invert X motion ☑ Invert Y motion

Z-Stepper

○ Not installed
● Earl's stepper
○ Provided by X-Y Controller
☐ Invert Z motion

Fig. 1. MPScan: principles of operation. (A) Hardware connections. The minimal hardware configuration of MPScan requires the connections and devices in italic. (B) Hardware configuration control panel. (C) Scanning pattern generation. Gray portions of the *X* linescan belong to the turnaround trajectory of the fast mirror.

added to the computer via the USB ports. For instance, we use a USB-based PMD-1608FS board (Measurement Computing Corporation, Middleboro, MA) to acquire up to eight channels of analog data. The board is run during user-written scripts thanks to an OCX software component that controls the board.

MPScan provides up to three hardware configurations for computer-controlled, motorized positioning of the field of view. In the first option, which is that of our microscope, X-Y translation of the preparation is performed by a NEAT 300 two-axes table (Danaher Precision Systems, Salem, NH) connected to the computer by a RS-232 link, while Z- focus is done by a stepper motor controlled by two digital output lines from the main acquisition board (Fig. 1(A) and (B)). The second configuration, which is identical to that described in Pologruto et al. (2003), relies on a serial port-driven MP-285 microelectrode positioner (Sutton Instruments, Novato, CA) to move the stage in three directions. Finally, MPScan offers the possibility to use an ESP-300 three-axes controller (Newport Corp., Irvine, CA). This industrial device is advantageous in in vivo experiments, since it can control stages capable of translating heavy asymmetric loads.

Users can conveniently set the hardware configuration of their data acquisition boards, *X*–*Y*–*Z* actuators, laser shutter and mirrors by selecting the appropriate options in a dialog box (Fig. 1(B)).

2.5. *Laser intensity control*

A simple and inexpensive software-controlled laser intensity control was added to the microscope by employing a servo mechanism, originally designed as a winch for sailboat models (model HS-225BB; Hitec, www.hitecrd), to rotate the half-wave polarizing plate via a worm drive. The digital input of the servo is connected to the output of one of the two general purpose counters on the main data acquisition board in order to receive a TTL pulse train whose duty cycle determines the rotation angle of the servo. This angle sets the desired attenuation of the laser beam, ranging from 0 to 100%, using an empirical cosine relationship whose coefficients are stored permanently in MPScan.

3. Software description

3.1. *MPScan: principle of operations*

In most imaging modes, MPScan drives the fast mirror using a symmetrical waveform in order to record pixels during the forward and backward linear motion of the mirror. The dual direction linescan pattern, which resembles a sine wave, consists of two linear (e.g. constant speed) segments connected with parabolic turnabout sections. The shape of the linescan was designed to avoid sharp discontinuities that could have resulted in distortions of the fast mirror position due the limited bandwidth of the scanning servomechanism. The linear segments ensure spatial uniformity of the images while maximizing the duty cycle of linescans, which amounts to 80% (i.e. during a scan, 80% of the time is used to collect usable pixels). *XY* scan patterns are generated by combining the fast linescan of the *X* mirror with a slower triangular frame pattern for the *Y* mirror (Fig. 1(C)). The scanning pattern is rotated by a angle chosen by the user using a trigonometric formula. The resulting digital values are then offset and scaled to adjust the position and area of the *XY* scan according to the user's choice. The *X* and *Y* mirror positions are then stored in an output buffer and delivered as analog voltages to the scanner control boards.

Continuous data acquisition is implemented by a "ping-pong" double buffering scheme. In this algorithm, one half of a large circular buffer containing several frames is filled using demand mode, scatter-gather DMA (Direct Memory Access) while the other half is processed. Linescans are demultiplexed, stripped of the pixels belonging to the turnaround portion in the waveforms

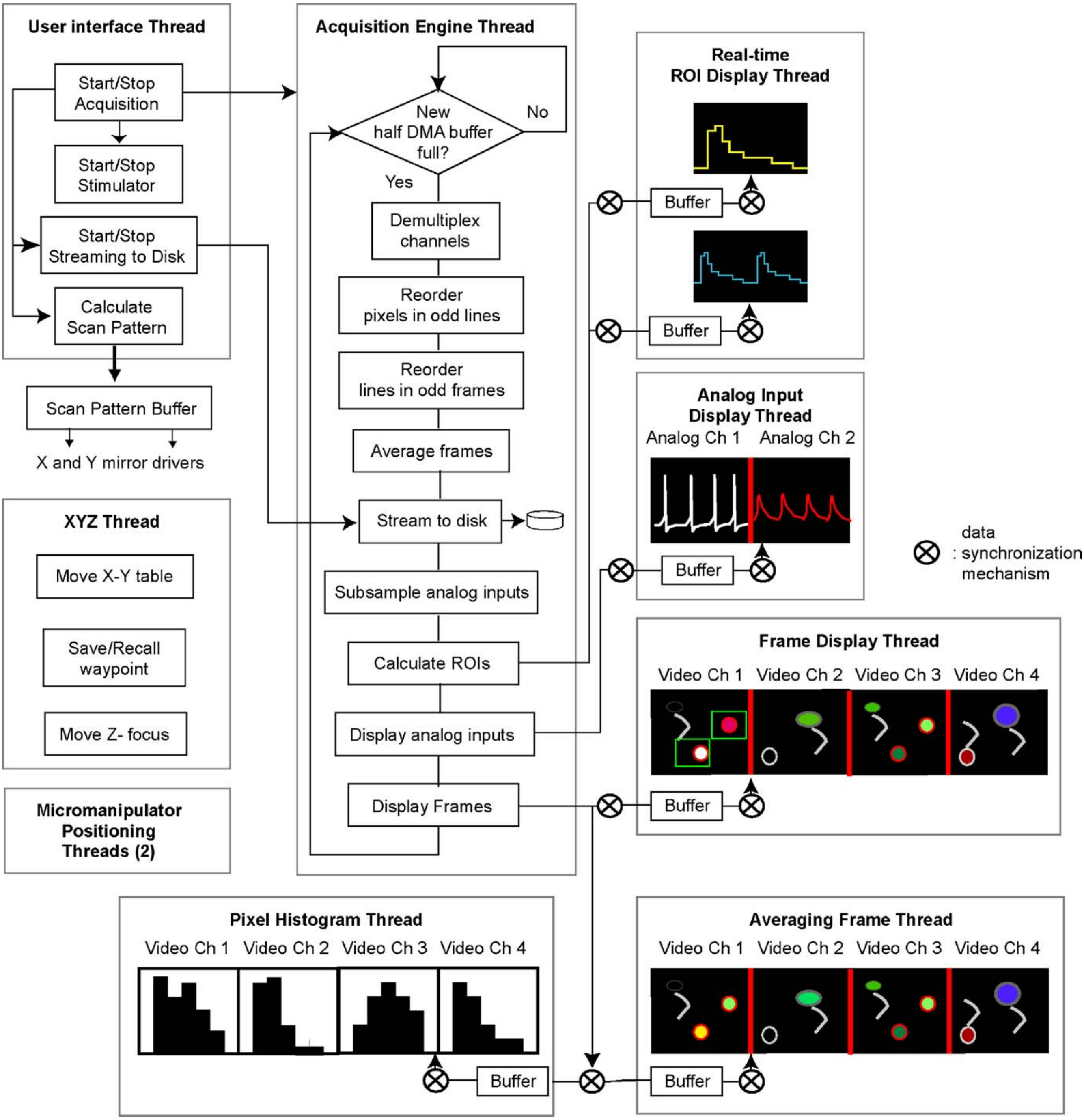


Fig. 2. MPScan: architecture. This diagram describes the functional architecture of MPScan in movie mode. Simultaneously running threads are shown in bold captions. In this example, movie acquisition takes place while two real-time ROIs are plotted. Data from up to four imaging channels or two imaging and two analog channels can be processed simultaneously.

and inverted if they were taken during the backward motion of the mirror.

Simultaneous execution of scan mirror control, frame acquisition and other time-critical operations relies on multithreading (Silberschatz et al., 2002). In this scheme, concurrent activities are assigned to separate lightweight processes called threads. As seen in Fig. 2, MPScan consists of an acquisition engine thread which runs at a high priority, a user-interface thread which allows experimenters to control MPScan during acquisition and several other threads that are executed nearly simultaneously. To avoid synchronization problems, especially when a data-producing thread wants to access a buffer at the same time as a data-consuming thread, MPScan makes extensive use of process synchronization primitives provided by the Windows operating system.

3.2. MPScan: general features

Scanning modes include movie, stack, fast stack, linescan and region of interest (ROI) scan. In all modes, users can turn on or off disk saving at any time.

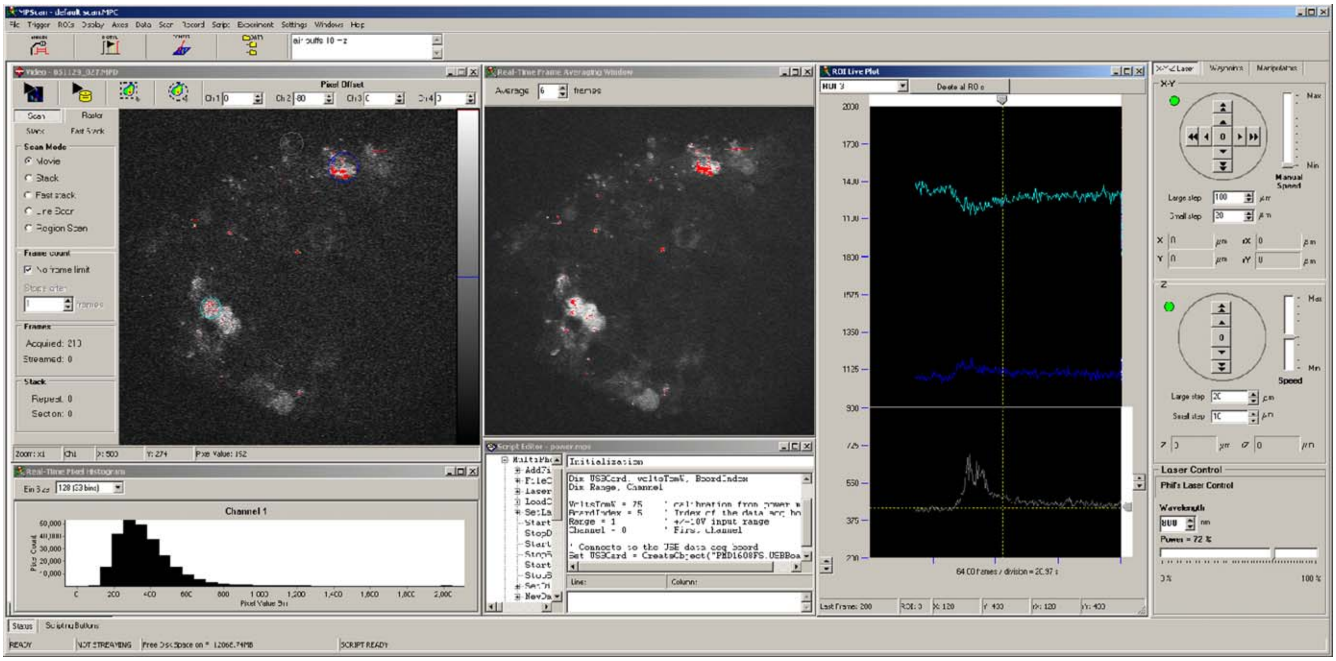
In movie mode, MPScan continuously acquires frames ranging from 100 × 10 to 512 × 512 pixels from up to four imaging channels. Incoming frames can be displayed in full or enlarged in size, in gray scale, false colors or using a custom look-up table (LUT) for each channel. Smaller frame sizes, allowing faster frame rates, can be selected by choosing a rectangular region with the mouse. During acquisition, the scan pattern can be freely rotated and the amplitude of the *XY* scan can be reduced to magnify the field of view by a factor ranging from ×1 to ×10. The nominal frame rate is 3 fps with a 512 × 512 frame size resulting in a 900 μm × 900 μm field of view with a ×20 water immersion objective. MPScan can achieve 10 fps with images as large as 256 × 256 pixels.

MPScan is able to display and continuously update a histogram of pixel values for each of the four video channels. This histogram is useful for adjustment of gain and offset of the photomultipliers in order to take advantage of the full dynamic range of the data acquisition board. In addition, a separate window displays the continuous average of incoming frames as they are collected.

The movie mode also provides the ability to select multiple rectangular or elliptical regions of interest (ROIs) and to plot the average of pixel intensities in a scrolling display in near real-time. One ROI per channel can be designated as background and all other traces are corrected by removing the intensity of the background ROI. The screenshot in Fig. 3 shows the ROI plot and other windows during a functional calcium imaging experiment.

In stack mode, MPScan acquires one or more frames for averaging and moves the objective in the *z*-axis by a pre-set number of microns before another section is acquired. Laser power can be incremented automatically, via the device described in Section 2, by either a linear or an exponential relationship to compensate for increasing imaging depth in a scattering sample. During transition from one section to another, the laser shutter can be closed to prevent unnecessary exposure of the sample to the laser beam. MPScan can record time-series stacks by repeating stacks multiple times. Scripting code can be automatically executed after a section is acquired.

The fast stack mode is similar to, but much faster than stack mode. In fast stack mode, MPScan continuously acquires frames



air puffs 10 –z

○ Movie ○ Stack ○ Fast stack ○ Line Scan ○ Region Scan

Ch1 0 Ch2 -80 Ch3 0 Ch4 0

☑ No frame limit

Stop after 1 frames

Acquired: 213

Streamed: 0

Repeat: 0

Section: 0

Zoom: x1 Ch1 X: 500 Y: 274 Pixel Value: 152

Bin Size 128 (33 bins)

Average 6 frames

ROI 3

64.00 frames / division = 20.97 s

Last Frame: 200 ROI: 3 X: 120 Y: 400 dX: 120 dY: 400

Large step 100 μm

Small step 20 μm

X 0 μm dX 0 μm

Y 0 μm dY 0 μm

Large step 20 μm

Small step 10 μm

Z 0 μm dZ 0 μm

Wavelength 800 nm

Power = 72 %

Initialization

```
Dim USBCard, voltsToml, BoardIndex
Dim Range, Channel

VoltsToml = 75       ' calibration from power m
BoardIndex = 5       ' Index of the data acq bo
Range = 1            ' +/-10V input range
Channel = 0          ' First channel

' Connects to the USB data acq board
Set USBCard = CreateObject("PMD1608FS.USBBoa
```

READY NOT STREAMING Free Disk Space on * 12066.74MB SCRIPT READY

Fig. 3. MPScan: user interface. Example of two-photon functional calcium imaging of rat trigeminal ganglion neurons (Nguyen et al., in preparation) with MPScan. Cells were previously loaded with the calcium indicator calcium-green dextran and excited at λ = 800 nm. The user interface included the following window: frame viewer, frame pixel intensity histogram, frame averaging window, ROI pixel average intensity live plots, scripting environment and *XYZ* control. Three ROI intensity time-courses were simultaneously plotted during the experiment.

as a movie, displays them and streams data to disk while the z-focus is changed in a continuous manner. The effect of the z-motion is eliminated by choosing a distance between consecutive frames smaller than d_{axial}, the axial resolution of the microscope ($d_{\text{axial}} = 2\,\mu\text{m}$ with a ×40 water objective; Tsai et al., 2002). The z-position is changed by 0.125 μm increments at intervals as short as 1 ms with accurate timing provided by the Windows Multimedia timer. Automatic laser intensity control can also be enabled to compensate for scattering losses with increasing imaging depth. Multithreading allows concurrent execution of these activities and eliminates the overhead incurred during a traditional stack procedure resulting from stopping and restarting operations after each section.

To improve temporal resolution of fast physiological signals, single linescans can be repeated at the same Y location with a rate as fast as 2.6 kHz. Region scan mode allows users to specify multiple rectangular, elliptical or polygonal regions which are merged to create a raster pattern sent to the mirrors.

All scan parameters are collectively known as a configuration. MPScan maintains a list of the 10 previous configurations and allows users to recall a previous configuration. In addition, configurations can be stored in files and reinstated later either from the user interface or programmatically via ActiveX Automation (see later), allowing chaining of configurations into a sequence.

The field of view can be positioned in the horizontal plane or vertical direction by translation stages controlled by MPScan (see Section 2). Users have the possibility to store an arbitrary number of X–Y–Z coordinates along with a useful name and a snapshot of the current frame, and to move back to these waypoints later.

MPScan includes several features specific for cellular electrophysiology. A built-in digital stimulator can deliver trains of digital bit patterns with a precision and resolution of 1 ms to control TTL-compatible devices. The stimulator can be started any time during a scan or can be triggered as the same time as the scan begins. Script-based programmatic control of the on-board general purpose timers can also provide accurate digital pulse trains.

MPScan can control up to two MP-285 computerized micromanipulators to visually place microelectrodes on selected cells during two-photon targeting experiments (Margrie et al., 2003). In this technique, the patch pipette is filled with a fluorophore whose emission spectrum differs from that of the cells of interest. The microelectrode tip and the neurons are imaged using two different video channels. In MPScan, these two channels can be overlaid on the same image with the electrode in one color and the cells of interest in another. Once the microelectrodes have been placed, MPScan can deliver up to two different stimulation patterns via two analog channels. These outputs are usually fed to the command inputs of microelectrode amplifiers. Stimulation patterns include seal test for whole-cell patch clamping experiments, trains of pulses and steps with programmable incremental amplitude.

Signals from up to two analog sources such as microelectrode amplifiers can be acquired continuously (e.g. gap-free) alongside with two video streams by the main data acquisition board. Because these signals are acquired at the same acquisition frequency as the optical frame data, the analog channels are oversampled. To discard the extraneous data points, the analog data streams are subsampled by a factor set by the user. The remaining subsampled data are displayed in an oscilloscope-like display and saved to disk.

3.3. MPScan: scripting features

MPScan is designed to be an Automation server, allowing external programs to control many of its operations via the ActiveX Automation standard (for a description of this technology, see Nguyen and Miledi, 2003). External programs written in various languages (including Matlab or Labview) can load a configuration, start a scan or move the preparation or the focus and control digital output lines. In addition, MPScan is also an Automation Client. MPScan features an ActiveX Scripting integrated development environment (IDE) to allow users to write their own program from within MPScan. The language currently supported is VBScript, an interpreted subset of the widely used Microsoft Visual Basic language (Microsoft, Redmond, WA). The IDE, which is similar to the one described previously (Nguyen and Miledi, 2003) includes a script editor featuring simple debugging facilities and a window to browse the functionalities exposed by other programs or software components. Event-driven code can be called automatically when a particular event occurs during a scan (e.g. when scanning starts or ends) or when users press on one of ten scriptable buttons.

3.4. Large-scale imaging of cell body distribution in the mouse somatosensory cortex

We demonstrate the ability of MPScan to automatically image large volumes of cortical tissue by combining the Fast Stack mode and scripting. The goal was to obtain a three-dimensional representation of cell bodies in a 1 mm × 1 mm × 1 mm volume of mouse cerebral cortex with micron resolution amounting to a final data set of 10^9 voxels. Imaging was done on one fluorescence channel at ∼9 frames/s. Each frame consisted of 256 lines of 256 pixels covering an area 240 μm × 240 μm. Automatic laser power control was enabled and used an exponential relationship to increase laser power as imaging progressed deeper in the tissue (Fig. 4(A)). To cover a 1 mm^2 area, 25 overlapping stacks spaced every 200 μm were tiled in a 5 × 5 mosaic (Fig. 4(B)). Automatic positioning of the field of view following completion of each stack was done by a script that moved the X–Y stage. The whole procedure, once started, did not involve any action from the user and lasted 68 min. This duration included the delay to move the objective back to its original z-position after each stack.

3.5. MPScope files

Data files used in MPScope are based on a proprietary format that relies on the Microsoft Structured Storage technology. This feature of the Windows NTFS file system allows each acquisition channel to be stored in its own stream, therefore avoiding the need to demultiplex pixels and analog values when reading the

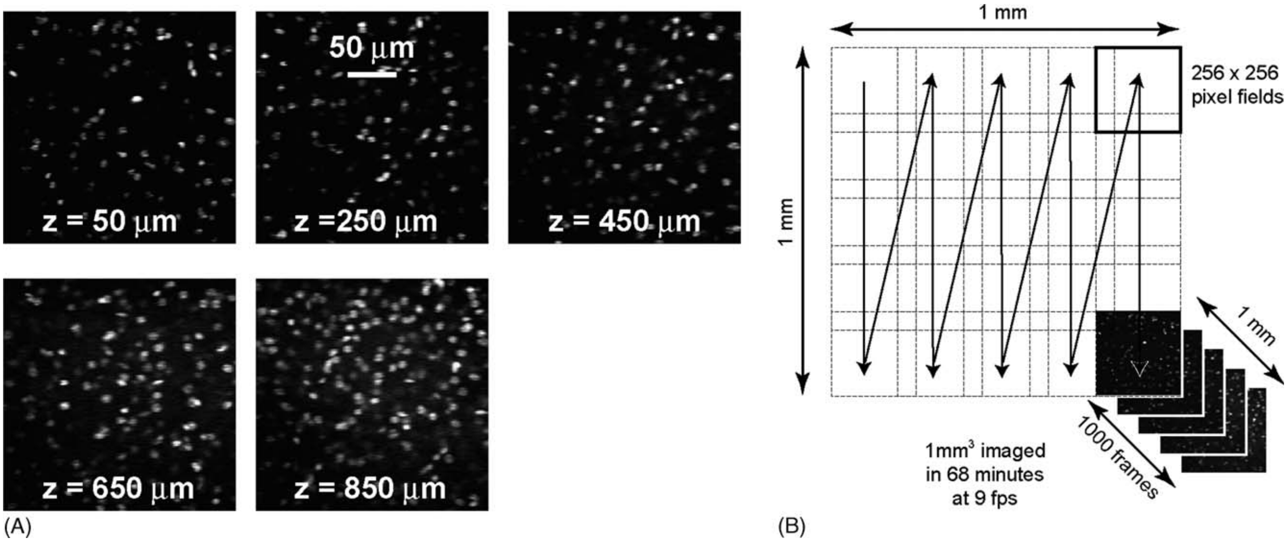


Fig. 4. Volumetric imaging using fast stack and scripting. Fast stacks were taken in fixed, sucrose-cleared mouse somatosensory cortex stained with Hoechst 33342, a dye that labels cell bodies (molecular probes). The preparation was excited at $\lambda = 760$ nm. (A) Example of images in the same fast stack taken at different depths. Laser power was automatically adjusted using an exponential relationship between illumination power and depth. Each image is the average of five consecutive frames. (B) Tiling scheme to sample a 1 mm^3 cortical volume.

file. Up to 51 data acquisition parameters can be written alongside video and analog data in a distinct stream called Summary Information. These parameters include for instance file comments, the *X*–*Y*–*Z* position of the scan, the number of frames in the file etc. The main reason to store these parameters in a Summary Information stream rather than in a more traditional file header is the ability to make MP files self-documenting and extensible. Users can display the name of all the stored parameters as well as their corresponding value in a “Properties” window anywhere in Windows simply by right-clicking on the file icon, without having to open the file in any program. This feature is similar to the one used in Microsoft Office files created by Word or Excel. Users have the possibility to add their own custom parameters to data files by using the AddFileParameter Automation method, which can be used either by programs written in VBScript in the MPScan integrated development environment or from any ActiveX Automation-enabled programming language. In addition, we provide an ActiveX control (MPFILE.OCX) allowing programs written in other languages (e.g. Visual Basic, Delphi, Matlab, C/C++) to open MPScope files, retrieve the values of parameters and read frame data without having to deal with the internal layout of the data files.

3.6. MPView

MPView is an analysis program designed to view, process and export frames generated by MPScan. MPView can open several MP files simultaneously and create virtual data files called workspaces designed to hold results from frame operations. From the user’s standpoint, MP files and workspaces are practically identical as they both contain a collection of frames, stored either on disk for data files or in memory for workspaces. Workspaces can be converted into regular data files by saving workspaces on the hard drive. We will thereafter refer MP files and workspaces simply as data files.

MPView display frames either in grayscale, false or custom colors in viewer windows, which allow browsing data files with usual functions (forward, backward, fast forward, fast reverse, first frame, last frame, jumping to a particular frame, zooming) associated with accelerator keys. Several independent viewer windows can be attached to the same data file. Pixel profiles can be obtained by drawing a line on a frame with the mouse. The profile is then displayed in an independent window and can be measured using cursors. Pixel statistics for a frame region can be generated by selecting the region with the mouse.

Operations on frames include gamma correction based on an histogram of pixel intensities with user defined thresholds for minimal and maximal pixel values, copy, averaging, frame subtraction and linear combination. Multiple frames can also be stacked by maximally projecting them on the *X*, *Y* or *Z* direction. More advanced frame processing can be done by exporting frames to the Matlab environment (Mathworks, Natick, MA). MPView automatically starts Matlab, creates Matlab arrays and fills them with pixel values from single or consecutive frames without involving any file operation thanks to the ActiveX Automation interface of Matlab. Matlab can reside either on the same computer as MPView or on a different machine connected via a local network. A Matlab console window allows further control over Matlab from within MPView.

The main purpose of MPView is to measure changes in optical signal intensities in subcellular regions or in whole cells across frames. Users can define regions of interest (ROI) either as a rectangular or elliptical area drawn with the mouse on a viewer or by using automatic object detection. An object, which typically represents a neuron, is a ensemble of adjacent pixels with values above a user-defined threshold and an area larger than a pre-set number of pixels. Objects are detected using a recursive eight-connected flood-fill algorithm (Foley et al., 1990), which takes 12 s to find 66 objects in a 512 × 60 pixels frame stack. The threshold for object detection can be determined beforehand by using the pixel profile tool. Once ROIs have been determined, double-clicking on a region of interest calculates the average value of all pixels in the ROI across frames. This generates a plot which is displayed in a separate window. Plot values can be exported as a text-based, tab-delimited array via the clipboard.

Exporting images to more specialized software packages can be done by saving frames either as bitmaps or as multipage TIFF files. This allows MPScope to be interoperable with other image analysis softwares such as ImageJ, which is available free of charge from the National Institute of Health. It is also often desirable to convert microscope data files into a video clip for presentation purposes. As such, MPView is able to create AVI (Audio Video Interleaved) video files from user-selected frames, which can subsequently be displayed either in a gray scale or in false colors at a frame rate selectable by the user.

4. Discussion

4.1. MPScan: design goal

The main design objective of MPScan was to maximize the frame rate of our custom multiphoton system, which was not limited by the scanning hardware but the performance of the LabView program that originally controlled the microscope. This program and also the ScanImage application developed in Matlab by the Svoboda lab (Pologruto et al., 2003) can achieve only two frames per second with 256 × 256 frames (Timothy O'Connor, Svoboda lab, Personal communication). In theory, the scanners allow frames to be acquired at a speed nearly five times faster.

The fastest possible frame rate is essential for at least two types of applications. The first one is the mapping of large neuronal areas with sufficient spatial and time resolution to detect calcium changes in well-defined cells, along with the ability to plot in near real-time the fluorescence intensity in regions of interest defined by the user (e.g. Fig. 3). The second one deals with acquisition of large neuronal volumes for morphological reconstructions (Fig. 4). MPScan achieves maximal performance and provides user-friendly, neurobiology-oriented features by implementing the design choices described below.

4.2. MPScan: implementation

First, writing MPScan in compiled code results in a program inherently faster than comparable applications developed in interpreted languages such as Matlab or Labview. A compiled program can also be optimized by taking advantage of APIs native to the Windows operating system. For instance, MPScan makes extensive use of the DrawDIB functions which allow direct access to the video hardware, the Multimedia timer for millisecond timing and the Structured Storage API for efficient storage of frame data.

Second, adoption of a sinewave-like, bidirectional linescan pattern with high duty cycle for the scan mirrors maximizes the frame rate, which is comparable to or higher than that of commercial MPLSMs. The fastest frame rate is limited only by the maximal frequency $f_{\max}$ at which the fast mirror can be moved. Currently, our galvanometric mirrors have an $f_{\max}$ of ∼1.3 kHz but future servo-controlled mirrors may be even faster. Bidirectional linescans require reordering pixels in alternate linescans during acquisition in real-time. This time-consuming operation further justifies the use of compiled code.

Third, the systematic use of multithreading in MPScan optimizes the computing load of the processor(s) and increases the responsiveness of the user interface. For instance, we routinely use MPScan to simultaneously acquire 256 × 256 frames at 10 Hz, display the incoming frames and stream them to disk, calculate and display averaged frames, stimulate the preparation using the built-in digital stimulator and plot the average pixel intensity of up to four regions of interest. Further, because scan patterns can be completely recalculated almost instantly in the user thread during frame acquisition, multithreading eliminates the need for the custom-built, PIC-based rotating hardware and for two analog output boards originally required by the Labview microscope software to output the scan pattern. This simplification makes the microscope design easier and less costly to replicate.

4.3. Automation of experiments and analysis

Many types of experiments require repeating a series of operations which can become time-critical, tiresome or error prone. Inclusion of a scripting language environment in MPScan solves these problems and allows easy customization by end-users, especially since many pre-written software components to control hardware via USB or serial ports are already available. ActiveX Scripting allows MPScan users to undertake challenging optical and electrophysiological experiments controlled by "meta-applications" involving many software components, programs and computers running simultaneously.

4.4. Application of MPScope to deep-tissue two-photon imaging

A major advantage of MPLSMs is their ability to image deep inside the brain (Helmchen and Denk, 2005). In principle, this feature allows the three-dimensional reconstruction of large volumes of neural tissue. However, this kind of experiment is, in practice, limited by the huge number of frames to take and by the speed at which these frames are acquired. The original Labview program was not capable to sample frames at more than 1 fps during stacks, due to the overhead of moving the *Z*-focus after taking each frame. In contrast, MPScan provides a nine-fold increase in frame acquisition speed when using the fast stack imaging mode. This feature, in combination with a script that automates tiling, allows a drastic reduction in the time needed to sample a 1 mm^3 of cerebral cortex from a full day to approximately 1 h.

5. Conclusion

All MPLSMs, including those custom-built, are very expensive instruments that require extensive care in the design of their optics and electronics to fully exploit the potential of the microscopes. Similarly, programs controlling custom-built MPLSMs should be written to maximize the capabilities of the microscope. To achieve the highest performances, advanced features of modern operating systems can be used to develop powerful control and analysis software comparable to much more expensive commercial programs.

MPScope is currently operational in two microscope setups in our laboratory. MPScope and its User's Guide are available free of charge for academic users at http://www.physics.ucsd.edu/~qnguyen/mpscope.htm/.

Acknowledgements

We would like to thank Ms. Nozomi Nishimura and Dr. Chris B. Schaffer (Cornell University) for suggestions to enhance MPScope. Q.-T.N. is funded by the National Institute of Mental Health (MH071566). P.S.T. and D.K. are supported by the National Institute of Health (RR021907 and EB003832) and the National Science Foundation (DBI0455027).

References

Denk W, Strickler JH, Webb WW. Two-photon laser scanning microscopy. Science 1990;248:73–6.

Denk W, Delaney KR, Gelperin A, Kleinfeld D, Strowbridge BW, Tank DW, et al. Anatomical and functional imaging of neurons using 2-photon laser scanning microscopy. J Neurosci Methods 1994;54:151–62.

Fan GY, Fujisaki H, Miyawaki A, Tsa RK, Tsien RY, Ellisman MH. Video-rate scanning two-photon excitation fluorescence microscopy and ratio imaging with cameleons. Biophys J 1999;76:2412–20.

Foley J, van Dam A, Feiner S, Hughes J. Computer graphics. Principles and practice. 2nd ed. Addison-Wesley: Reading; 1990.

Helmchen F, Denk W. Deep tissue two-photon microscopy. Nat Methods 2005;2:932–40.

Kleinfeld D, Mitra PP, Helmchen F, Denk W. Fluctuations and stimulus-induced changes in blood flow observed in individual capillaries in layers 2 through 4 of rat neocortex. Proc Natl Acad Sci USA 1998;95:15741–6.

Mainen ZF, Maletic-Savatic M, Shi SH, Hayashi Y, Malinow R, Svoboda K. Two-photon imaging in living brain slices. Methods 1999;18:231–9.

Majewska A, Liu G, Yuste R. A custom-made two-photon microscope and deconvolution system. Pflugers Arch Eur J Physiol 2000;441:398–408.

Margrie TW, Meyer AH, Caputi A, Monyer H, Hasan MT, Schaefer AT, et al. Targeted whole-cell recordings in the mammalian brain in vivo. Neuron 2003;39:911–8.

Nguyen Q-T, Callamaras NC, Hsieh C, Parker I. Construction of a 2-photon microscope for real-time Ca2+ imaging. Cell Calcium 2001;30:383–93.

Nguyen Q-T, Miledi R. e-Phys: a suite of electrophysiology programs integrating COM (component object model) technologies. J Neurosci Methods 2003;128:21–31.

Nguyen Q-T, Clay GO, Nishimura N, Tsai PS, Schaffer CB, Schroeder LF, et al. Pioneering applications of two-photon microscopy to mammalian neurophysiology: seven case studies. In: Masters B, So P, editors. Nonlinear microscopy. Oxford University Press, in press.

Nikolenko V, Nemet B, Yuste R. A two-photon and second harmonic microscope. Methods 2003;30:3–15.

Pologruto TA, Sabatini BL, Svoboda K. ScanImage: flexible software for operating laser scanning microscopes. Biomed Eng 2003, Online, http://www.biomedical-engineering-online.com/content/2/1/13.

Roorda RD, Hohl TM, Toledo-Crow R, Miesenbock G. Video-rate nonlinear microscopy of neuronal membrane dynamics with genetically encoded probes. J Neurophys 2004;92:609–21.

Ruthazer ES, Cline HT. Multiphoton imaging of neurons in living tissue: acquisition and analysis of time-lapse morphological data. Real-Time Imaging 2002;8:175–88.

Silberschatz A, Galvin PB, Gagne G. Operating systems concepts. 6th ed. New York: John Wiley and Sons; 2002.

Tan YP, Llano I, Hopt A, Wurriehausen F, Neher E. Fast scanning and efficient photodetection in a simple two-photon microscope. J Neurosci Methods 1999;92:123–35.

Tsai PS, Nishimura N, Yoder EJ, Dolnick E, White GA, Kleinfeld D. Principles, design and construction of a two-photon laser-scanning microscope for in vitro and in vivo brain imaging. In: Frostig RD, editor. In vivo optical imaging of brain function. Boca Raton: CRC Press; 2002. p. 113–71.

Tsai PS, Friedman B, Ifarraguerri AI, Thompson BD, Lev-Ram V, Schaffer CB, et al. All-optical histology using ultrashort laser pulses. Neuron 2003;39:27–41.

Glossary

ActiveX Automation: A COM standard to programmatically control local or remote applications or libraries.

API: Application Programming Interface. A set of programmatic procedures or functions specific to a particular task.

AVI: Audio Video Interleaved. A multimedia file format for movies or sound clips.

COM: Common Object Model. A set of specifications to make objects and applications interoperable in the Microsoft Windows operating system.

Demand Mode: An enhancement to DMA circuits allowing them to transfer data out of/to memory in short bursts.

DSP: Digital Signal Processor. A microprocessor optimized to process digitized analog signals.

DMA: Direct Memory Access. A method used by peripherals such as data acquisition boards to transfer data out of/to computer memory without intervention of the computer processor.

IDE: Integrated Development Environment. A comprehensive set of utilities, such as text editor, compiler, interpreter or debugger integrated in the same environment to develop programs.

LUT: Look-Up Table. A table that assigns a color to a particular pixel value for display purposes.

NI-DAQ: The API to control National Instruments hardware using conventional programming language.

NTFS: New Technology File System. The advanced file system native to Microsoft Windows 2000 and XP.

OCX: A software component, equivalent to a library of functions, compatible with the ActiveX Automation standard.

PCI: Peripheral Connect Interface. A standard connector design to plug peripheral cards inside recent IBM-PC personal computers.

PIC: A simple, low cost 8-bit processor for embedded applications.

ROI: Region of Interest.

Scatter-Gather: An enhancement to DMA circuits allowing them to transfer data out of/to memory to ("gather")/from ("scatter") a peripheral, such as a data acquisition board from discontinuous blocks of physical memory.

Serial port: A set of input/output lines in a standardized connector to communicate with peripherals using the RS-232 communication protocol. Also called a COM (communications) port (no relation with COM described above).

TIFF: Tagged Information File Format. An extensible file format specification for images, originally controlled by the Adobe Corporation.

USB: Universal Serial Bus. A standardized connector design and associated software and hardware specifications to connect peripherals to personal computers.

VBScript: Microsoft Visual Basic Scripting Edition. A subset of the Visual Basic programming language. VBScript is available in Windows as a component of the Explorer web browser.