NucLear MAC
User's Manual

Table of Contents

Table of Contents

1 Introduction

1.1 General Description

The NucLear MAC gamma camera computer is a high-performance system for acquisition, display and processing of nuclear medicine studies. It provides the performance expected from modern nuclear medicine computers, but with the ease-of-use associated with the MacOS operating system (Apple Computer, Inc.).

The NucLear MAC software follows the standard Macintosh user-interface guidelines. It operates through the familiar menus, windows, cursor controls and dialog boxes. NucLear MAC supports standard image formats of the Macintosh (PICT, TIFF), so images are easily shared with other standard programs. This provides many alternatives for image display and printing.

NucLear MAC comes in two forms: a processing-only station or an acquisition *and* processing station. The NucLear MAC acquisition and processing station consists of the MacOS computer, one or two data-acquisition boards, an interface assembly and operational software. These are installed and tested on your gamma camera by a dealer or one of our engineers. You may wish to have a person at your site observe the installation.

The acquisition system is installed by inserting the board(s) in the computer, attaching the gamma camera output wires to the input connector box, and attaching the cable from the connector box to the board(s). Standard BNC connectors are used for the X, Y, Z, and G cables. The following sections of the Installation chapter provide more detailed installation instructions, as well as signal specifications:

Hardware Installation
Software Installation

The NucLear MAC processing-only station consists of the MacOS computer, a software key and operational software. Installation instructions are in these sections of the Installation chapter:

Software Key Installation
Software Installation

When updating the NucLear MAC software in an existing system, refer to the Installation chapter section titled:

Software Update Installation

1.2 Computer Definitions

You should be familiar with the following terminology to best use the NucLear MAC system.

1.2.1 Bit

A bit is the smallest possible unit of computer memory. A bit has two states, typically called "zero" and "one," or "off" and "on."

1.2.2 Byte

A byte is a set of eight bits. A byte has $2^8 = 256$ possible states. Computer memory is usually measured in bytes or larger units. A simple way to think of the size of a byte is as one character in a text document. A typical gated-blood-pool study might occupy 250,000 bytes.

1.2.2.1 Kilobyte

A kilobyte (KB) is 1,024 bytes.

1.2.2.2 Megabyte

A megabyte (MB) is 1,024 kilobytes, which is a little over a million bytes.

1.2.2.3 Gigabyte

A gigabyte (GB) is 1,024 megabytes, which is a little over a billion bytes.

1.2.3 Random Access Memory (RAM)

Random access memory is a term for the high-speed, volatile memory found in today's computers. In this case, volatile means that the information stored in the memory is lost when power is removed. In addition, RAM is far more expensive than most types of non-volatile memory. The high speed of RAM overcomes these drawbacks for general computer operation, where lots of memory accesses are required. Thus RAM is used for things like data acquisition, processing and display, while non-volatile memory is used for long-term storage and archiving.

RAM for personal computers comes in the form of computer chips mounted on boards. These are called SIMMs or DIMMs. It is usually straightforward to upgrade a computer's memory by adding more or larger SIMMs or DIMMs. Today's personal computers typically have from eight to 128 MB of RAM, although the maximum is constantly increasing.

1.2.4 Disks and other non-volatile memory

Disks and other non-volatile memory are used for long-term storage of computer programs and data. These devices retain their data even when power is removed. The tradeoffs for these devices are among speed, cost and reliability.

1.2.4.1 Hard Disk

Almost all computers now come with a hard disk drive. A hard disk uses magnetism to store information on one or more disks rotating at a high speed. The read/write heads

do not physically touch the surfaces of the disks. Current hard disks have storage capacity ranging from 100 MB through ten GB. Improving technology continues to increase the upper limit.

1.2.4.2 Floppy Disk

Floppy disks are similar to low-capacity hard disks. Magnetism is used to store information on a bendable (floppy) disk within a plastic cartridge. Floppy disk drives are designed to eject and accept different floppies in the course of normal operation. The heads in a floppy disk drive come in contact with the surface of the disk in normal operation. This greatly reduces the reliability and speed of the device compared to hard disk drives. Floppies can store about two megabytes of data. Floppy disks may be on the verge of obsolescence due to the higher-capacity cartridge drives that are now available.

1.2.4.3 Cartridges

Data cartridges are an attempt to merge the best features of hard disks and floppy disks. Namely, cartridge drives try to achieve the speed of hard disk drives while giving the portability of floppy disks. Most of the current cartridge drives (e.g. Zip or Jaz) use hard-disk technology. Some drives use magneto-optical technology. The latter are slower but have better long-term stability for archiving. Capacities of current cartridges range from 100 MB to two GB.

1.2.4.4 Tape

Tape drives use magnetic tape that is similar to high-quality audiotape. Tapes are high capacity, storing hundreds of gigabytes. However, tape drives are quite slow, which limits their uses to archiving and backing up hard disks.

1.2.4.5 CD-ROM

CD-ROM drives use optical technology. The lack of any physical contact with the disc surface improves speed and reliability. In the past, one had to use expensive equipment to place information on CD-ROMs. This limited their use to distribution of pre-recorded music and software. More recently, two variations of compact disc technology have been developed. These are called CD-R and CD-RW. In both cases, relatively inexpensive equipment is used to place information on a compact disc. The discs are then readable by standard CD-ROM drives. In the case of CD-RW, you can rewrite the disc, whereas you can only write once with the CD-R technology. Compact discs have a capacity of 650 MB.

1.2.5 Hybrid memory

1.2.5.1 Virtual memory

Virtual memory amounts to one type of memory mimicking another. Space on a hard disk is used to give the illusion of having extra RAM. This is attractive because hard disk space is less costly than adding more RAM. There is a price in terms of performance. It is much slower to retrieve data from a hard disk than from RAM.

1.2.5.2 RAM Disk

A RAM disk is a portion of computer RAM that is set aside to act like a disk. Disks and free RAM are used differently, but a RAM disk acquires the advantages and disadvantages of RAM: it is much faster than non-volatile disks, but the data it contains is lost when the computer is turned off.

1.2.6 File

A file is a data structure on a disk. A file can contain either executable programs or data. Under MacOS, the Finder represents files by icons. Executable programs are called applications, and most data files are called documents. You can manipulate files in many ways, including copying them, deleting them and opening them. You will encounter some or all of the following file icons when using the NucLear MAC system.

 NucLear MAC or NucLear Power application program

 A study file, which contains images, overlays and other information

 The "NucLear MAC Prefs" file or any protocols file created by NucLear MAC

 A graph file, which contains a graph created by NucLear MAC

 A grouping file, which contains a list of study and graph files along with their screen locations

 A macro file, which contains text instructions that NucLear MAC can follow

 A user function file, which contains an external function that NucLear MAC can run

 A bull's-eye plot overlay file

 A bull's-eye plot database file

1.2.7 Folder

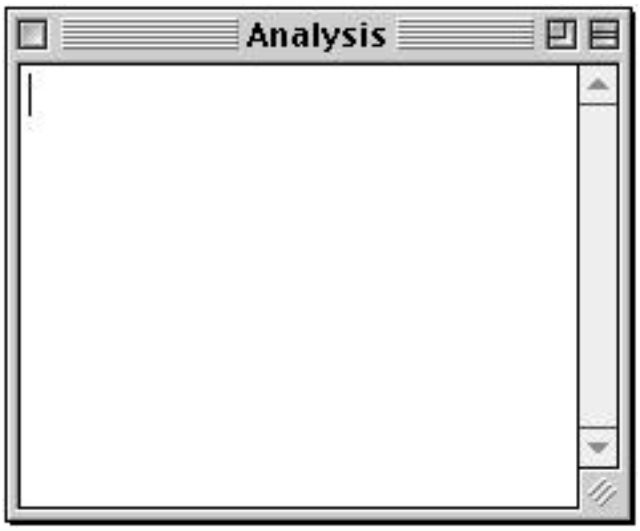
A folder is technically a directory, which is a place where files can be placed. Under MacOS, folders are used to organize files on disks in much the same way that actual folders are used to organize paper documents.

1.2.8 Desktop

Under MacOS, the desktop is the combined area of all the monitors attached to the computer. Technically speaking, the operating system owns the desktop; only the Finder can draw items directly on the desktop. Other applications, such as NucLear MAC, are limited to drawing in their own windows.

1.2.9 Window

Windows are areas overlaying the desktop. They are somewhat analogous to sheets of paper. Application can draw into windows, and applications also use windows to receive information from users. Windows are usually rectangular, as in the case of a word-processing document, which is drawn to look like a printed page. NucLear MAC windows are described in NucLear MAC Basics chapter.



NucLear MAC Analysis Window

1.2.10 Menu

A menu is a window containing a selection of items. Typically a menu appears only when you click in its title and disappears when you release the mouse button. Most MacOS applications have a set of menus whose titles appear in the menu bar when the application is active. The menu bar is the narrow strip at the top of the main monitor. You use menus to issue commands to applications and to provide information.

A sample menu

1.2.11 Cursor

A cursor is the symbol drawn to indicate the current location of the mouse pointer. The cursor symbol can convey what takes place when you click the mouse in a given location. For example, the magnifying glass cursor is used to zoom in on one portion of an image. In other situations the cursor indicates which mouse movement produces a result. For example, the horizontal arrow cursor ↔ means that moving the mouse left or right causes an action.

1.3 Computer Basics

NucLear MAC processing systems can be installed on any Macintosh or MacOS-compatible computer running Color Quickdraw and System Software of at least version 6.0.5. NucLear MAC acquisition and processing systems can be installed on any Macintosh or MacOS-compatible computer having at least two free nuBus slots or one free PCI slot and running Color Quickdraw and System Software of at least version 6.0.5. As explained in the Software Installation section, there are different but functionally identical versions of the NucLear MAC software to run on Macintosh computers with different hardware. These are just a few of the Macintosh models that can be part of a NucLear MAC system: II, IIx, IIcx, IIci, IIsi, IIfx, IIvx, Quadra 700, Quadra 800, Quadra 840, Quadra 900, Quadra 950, Power Mac 7100, Power Mac 8100. The user can decide between the various systems based on desired processing speed and number of available interface slots. In general, your computer should have at least eight megabytes of RAM installed in order to run NucLear MAC. For SPECT, and especially gated-SPECT acquisition and processing, you should have at least 16 MB of RAM installed.

1.4 Software Basics

The NucLear MAC user interface generally follows the standard guidelines recommended by Apple. The same menu, dialog, file sharing, etc. operations that are used in other standard Macintosh programs are also used in the NucLear MAC. This feature makes learning the basic operations very easy for anyone who is familiar with at least one other Macintosh program.

Note: It is **strongly** recommended that users familiarize themselves with basic Macintosh operation before attempting to use NucLear MAC. Users can do this by

reading the Macintosh manual and/or taking the “Guided Tour” which comes with most Macintoshes and MacOS computers. In particular, be sure to gain familiarity with the Macintosh concept of “Folders” being used to store files (as well as other folders); fully understanding this concept is critical in avoiding difficulties associated with inadvertently misplaced files.

1.5 Program Files

The NucLear MAC software has two essential files:

1) The **main application program**, e.g. “NucLear Power 5.0”

2) The **preferences** file, “NucLear MAC Prefs”

The **application program** file has a name such as “NucLear MAC 3.1” where the last group of numbers is used to indicate the software version. 3.1 is more recent than 3.0, for example. The first number indicates the major release number. The second number is the minor release number. When there is a third number, the version has small changes or bug fixes. To load software onto a computer that has not had a previous version of the NucLear MAC software on it, follow the instructions in the Software Installation section.

When a NucLear MAC program is started for the first time on a computer, it automatically generates a file within the Macintosh “System” folder called “NucLear MAC Prefs”. The NucLear MAC program uses this **preferences** file to save its internal parameters and user protocol settings. The “NucLear MAC Prefs” file must be left within the Macintosh “System” folder. Note: when using System 7.0 or later, the “NucLear MAC Prefs” file is placed in the “Preferences” folder that is inside the “System” folder.

When the NucLear MAC program is started subsequently, it uses the existing “NucLear MAC Prefs” file. If you make any permanent changes to the protocols or program parameters, then the changes are stored automatically in the “NucLear MAC Prefs” file. Permanent changes are made by adding and deleting protocols, as well as by clicking on the “Set Default” buttons in the various dialog boxes containing program settings. Scientific Imaging recommends that you make backup copies of the preferences file periodically. This way you can avoid re-entering all your protocols and settings should the preferences file ever be damaged.

2 Installation

2.1 Hardware Installation

This section describes how to install NucLear MAC hardware into a Macintosh or MacOS-compatible computer. If you are installing two NucLear MAC nuBus boards, then proceed with section 2.1.1. If you are installing a single NucLear MAC PCI board, then skip to section 2.1.2.

2.1.1 Board Installation and Connection, nuBus

Your computer must have at least two free nuBus slots. The only tool required is a flat-bladed screwdriver.

Scientific Imaging provides the following hardware:

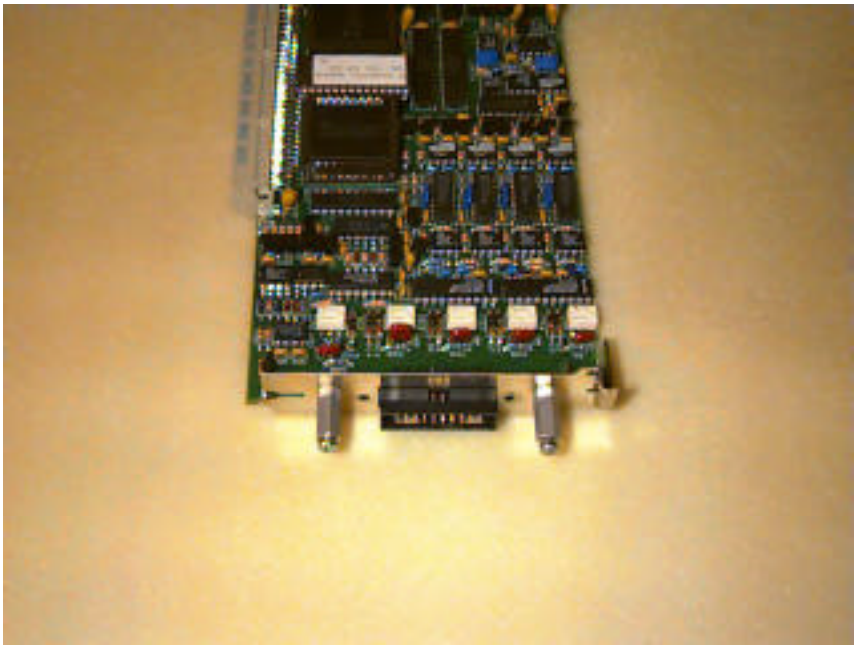
1. NucLear MAC analog board (long, upper left)
2. NucLear MAC digital board (short, lower left)
3. Cable, DB25M to DB25M, 1-3 ft. (upper right)
4. Interface adapter, DB25F to 50-pin and 20-pin connectors (lower middle)
5. 0.625” standoffs and mounting hardware, (2) (lower middle)
6. Interface box, DB25F to three or more BNC connectors (lower right)

Step 1—Set up the computer

Set up the Macintosh computer according to the manufacturer’s instructions. Verify that it is operating correctly before proceeding.

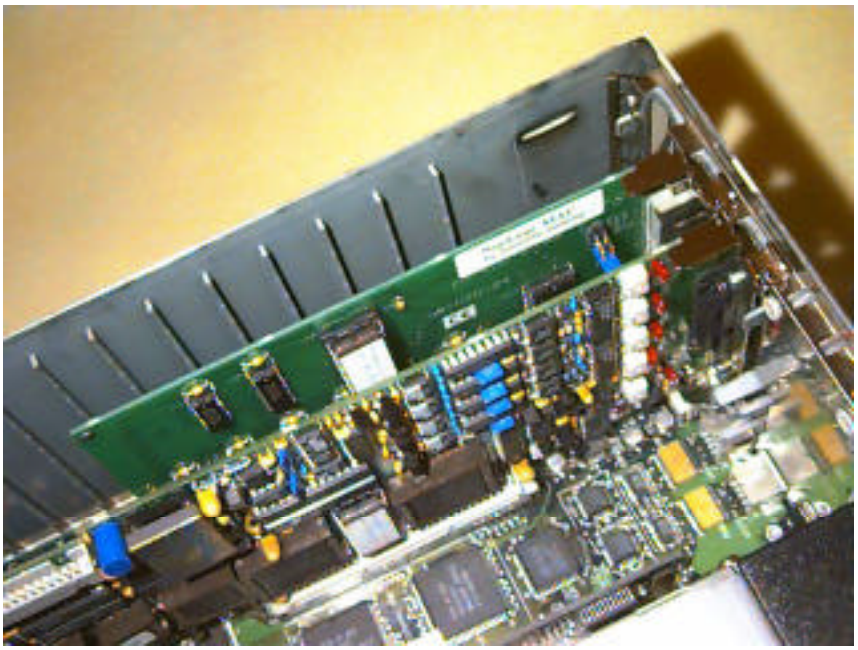
Step 2—Install the board standoffs

Install the two standoffs into the NucLear MAC analog (long) board using the screws and washers provided. Use the two outer holes. The board should look like this:



Step 3—Install the boards

Turn off the computer. Install the two NucLear MAC boards into adjacent nuBus slots in the computer. To avoid bending any connector pins, make sure the alignment is good before applying force. The longer analog board must be installed to the right of the shorter digital board, as in the photograph:

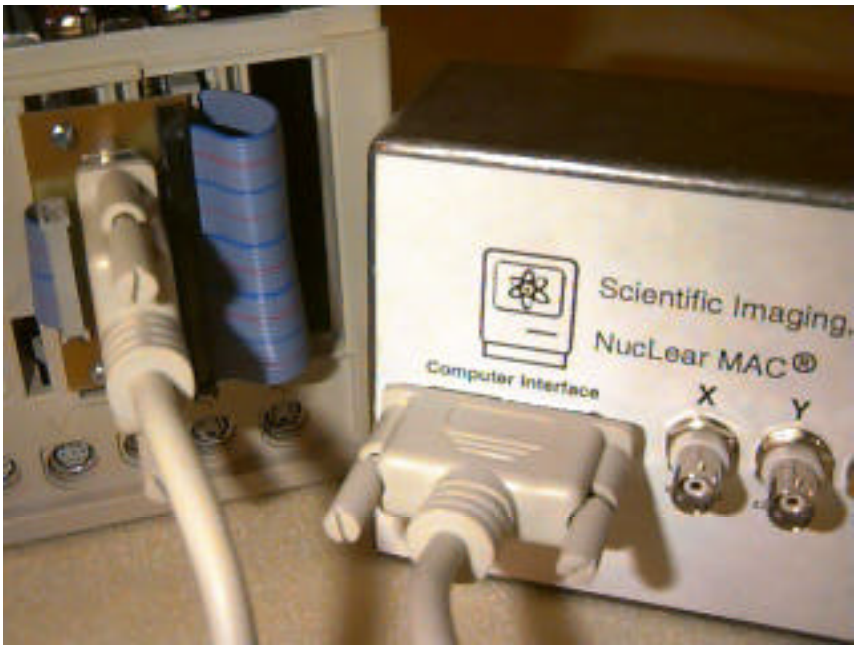


Step 4—Install the interface adapter

Close the computer. Remove the two screws and two washers from the standoff ends protruding from the analog board. Install the interface adapter by first plugging the 20-pin connector into the 20-pin male receptacle on the analog board. Then plug the 50-pin connector into the 50-pin male receptacle on the digital board. Finally, use the screws and washers to firmly attach the interface adapter board to the protruding standoff ends:

Step 5—Connect the interface box

Plug one end of the one-, two- or three-foot DB25M-DB25M cable into the DB25F receptacle on the interface adapter board. Use the screws on the cable to create a secure attachment. Plug the other end of the cable into the DB25F receptacle on the interface box. Again, use the screws on the cable to create a secure attachment:



Continue with section 2.1.3, Interface Box Connection.

2.1.2 Board Installation and Connection, PCI

Your computer must have at least one free PCI slot. The only tool required is a Phillips screwdriver.

Scientific Imaging provides the following hardware:

1. NucLear MAC analog/digital PCI board (upper left)
2. Cable, HDP68F to DB25M, 3 ft. (lower right)
3. Interface box, DB25F to three or more BNC connectors (lower left)

Step 1—Set up the computer

Set up the Macintosh computer according to the manufacturer's instructions. Verify that it is operating correctly before proceeding.

Step 2—Install the board

Turn off the computer. Install the NucLear MAC PCI board into a free PCI slot in the computer. Follow the computer manufacturer's instructions for this procedure.

Step 3—Connect the interface box

Plug the 68 pin female connector of the three-foot cable into the 68 pin male connector of the NucLear MAC PCI board. Push gently until you hear the latches click. Plug the DB25M end of the cable into the DB25F receptacle on the NucLear MAC interface box. Use the screws on the cable to create a secure attachment.

NOTE! The connectors on the three-foot cable may fit certain SCSI or serial connectors on your computer or other devices. To avoid damage to your computer or NucLear MAC hardware, be sure to plug the NucLear MAC cable only into the NucLear MAC PCI board and the NucLear MAC interface box.

2.1.3 Interface Box Connection

Connect male-terminated BNC cables to the interface box. The connectors are labeled to indicate which signals are required from your camera controller or ECG unit:

X: Input. The voltage indicating horizontal position of a detected count. This signal must be in the range of -2.5 volts to +2.5 volts.

Y: Input. The voltage indicating vertical position of a detected count. This signal must be in the range of -2.5 volts to +2.5 volts.

Z: Input. The trigger signal for a detected count. This should be a TTL-level signal. In default operation, the Z signal should remain at a high level until a count is detected, at which point it goes low for a short duration. The X and Y signals must be valid when Z goes low. Alternatively, you can reverse the polarity of the Z detector by switching a jumper in the interface box. In this case, a rising edge indicates a detected count.

Z2: Input (optional). In dual-trigger systems, this TTL-level signal indicates count detection in a second head or energy window. The polarities of Z and Z2 are identical. In dual-Z systems you cannot change the polarity of the interface box by switching the internal jumper.

Gate: Input (optional). A TTL-level signal reflecting the electrocardiogram of a patient. A rising edge on this input should correspond to end-diastole.

Status: Input (optional). Used to monitor gantry movement during acquisition of SPECT studies. In versions 3.2 and older of the software, Status is assumed to rest at a high level (greater than three volts). While the gantry moves, the signal should be pulled to a low logic level.

Advance: Output (optional). Used to control gantry movement during acquisition of SPECT studies. This is a TTL-level signal. Operation may vary depending on the camera controller. Typically Advance rests at a high logic level, and a short low-logic-level pulse tells the camera gantry to move to the next position.

SPECT Camera Controllers that use Status and Advance:

GE 400T. Uses TTL-level controls. "Advance" on the interface box connects to the "Advance" on the Rotation Control Module (part of camera control rack). "Status" on the interface box connects to the "Status" on the Rotation Control Module. The Rotation Control Module must be set to send and receive active-low TTL signals.

Siemens Orbiter (Without DOT). Uses TTL level controls. "Advance" on the interface box connects to the "Camera Step" pin on the bottom of the gantry (Pin 4) (use pin 1 for ground). "Status" on the interface box connects to the "Motor Run" pin on the bottom of the gantry (Pin 5) (Use pin 7 for ground).

2.1.4 Serial Port Connection (RS-232-driven cameras only)

For the following SPECT cameras, rotation control is achieved via serial communication:

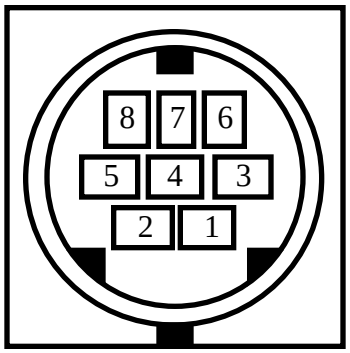
Siemens Diacam
Siemens Orbiter (With DOT)
Technicare Omega

Philips ARC 3000 or ADAC equivalent
Picker SX-300

Connect a standard Macintosh modem cable (mini-DIN8 to DB25) to the modem port of the Macintosh. Then connect the DB25 end of the cable to the camera controller, as per the manufacturer's instructions. In some instances, gender changers and/or null modem adapters may be required to mate the transmit line of the computer to the receive line of the camera and vice versa. Note that NucLear MAC does not currently use hardware or software handshaking.

The following information may be helpful. It is taken from the Apple Macintosh Family Hardware Reference.

Pinout of the Macintosh serial connector:



Signal assignments:

Pin	Signal	Description
1	HSKo	Output Handshake
2	HSKi	Input Handshake or external clock
3	TxD-	Transmit Data -
4	GND	Signal ground
5	RxD-	Receive Data -
6	TxD+	Transmit Data +
7	GPi	General-purpose input
8	RxD+	Receive Data +

“The transmit-data and receive-data lines of the Macintosh II serial interface conform to the EIA standard RS422, which differs from the more commonly used RS-232-C standard in that, whereas an RS-232-C transmitter modulates a signal with respect to a common ground, an RS-422 transmitter modulates the signal against an inverted copy of the same signal (to generate a differential signal). The RS-232-C receiver senses whether the received signal is sufficiently negative with respect to ground to be a logical 1, whereas the RS422 receiver simply senses which line is more negative than the other. An RS422 signal is therefore more immune to noise and interference, and degrades less over distance, than an RS-232 signal.

“If you ground the positive side of each RS422 receiver and leave unconnected the positive side of each transmitter, you’ve converted to EIA standard RS-423,

which can be used to communicate with most R-232-C devices over distances up to approximately 50 feet."

2.2 Software Key Installation

This section describes how to install a NucLear MAC software key into a Macintosh or MacOS-compatible computer. If your software key is for a USB port, then see USB Key Installation below. If your software key is for an ADB port, then see ADB Key Installation below.

2.2.1 USB Key Installation

Your computer must have at least one Universal Serial Bus (USB) port. No tools are required.

Scientific Imaging provides a software key that looks like the following:

Step 1—Set up the computer

Set up the Macintosh computer according to the manufacturer's instructions. Verify that it is operating correctly before proceeding.

Step 2—Install USB support files

Locate the files on your installation disk named "USBSentinel" and "RainbowUSBShim". These should be in the folder named "System Folder Items". Drag these two files to the active System Folder on your hard drive. When you are prompted, allow the system to place the files in the Extensions folder. Restart the computer.

Step 3—Locate a free USB port

Locate a free USB port on the back of the computer. Alternatively, there may be a free port on your keyboard, or you may have a USB hub that provides additional ports.

Step 4—Connect the key

Plug the USB software key into the free USB port:

2.2.2 ADB Key Installation

Your computer must have at least one Apple Desktop Bus (ADB) port. No tools are required.

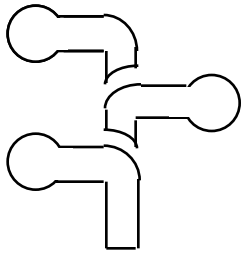
Scientific Imaging provides a software key that looks like the following:

Step 1—Set up the computer

Set up the Macintosh computer according to the manufacturer's instructions. Verify that it is operating correctly before proceeding.

Step 2—Locate a free ADB port

Turn off the computer. Locate an ADB port on the back of the computer. ADB ports are identified by the following symbol:



Step 3—Connect the key

If there are two ADB ports on your computer and one is free, then simply plug the cable end of the software key into the port. Otherwise, unplug an existing ADB device (probably a keyboard or mouse) from the ADB port on the computer. Then plug the device into the ADB port located on the software key. Finally, plug the cable end of the software key into the ADB port on the computer:

2.3 Software Installation

This section describes how to install NucLear MAC or NucLear Power software onto a Macintosh or MacOS-compatible computer. If your computer already has an older version of NucLear MAC or NucLear Power, then please refer to the section titled Software Update Installation for instructions.

Scientific Imaging provides NucLear MAC software on standard 3.5" DS/HD Macintosh-formatted floppy disks. For any NucLear MAC software revision, there are three functionally identical program versions. These are:

NucLear MAC
NucLear MAC SANE
NucLear Power

You have been provided with one or more of these releases depending on the hardware in your computer. All versions of NucLear MAC require a system software version of at least 6.0.5 and Color Quickdraw.

NucLear MAC runs on Macintosh computers based on the MC680x0 family of microprocessors. In addition, NucLear MAC requires the presence of a hardware-based floating-point unit (FPU). Examples of these computers include the Mac II, IIci, IIfx, Quadra, and others.

NucLear MAC SANE runs on any Macintosh or Power Macintosh computer that has the appropriate system software installed. NucLear MAC SANE is intended for MC680x0-based Macintosh computers that lack hardware-based FPUs.

NucLear Power runs on Power Macintosh computers in native mode. This version of NucLear MAC provides the best performance, and is typically several times faster than the other versions for operations involving a lot of processing.

In the following installation steps, the name "NucLear MAC" may be used to refer to any of the three software versions.

Step 1—Set up the computer

Set up the Macintosh computer according to the manufacturer's instructions. Verify that it is operating correctly before proceeding.

Step 2—Set up the hardware

If you are setting up an image acquisition system that will connect to a gamma camera, then follow the instructions in the section titled Hardware Installation. Otherwise, if you are setting up a processing station, then follow the instructions in the section titled Software Key Installation.

Step 3—Check the memory manager

Turn on the computer. Activate the Memory Control Panel. If "Modern Memory Manager" is turned off, then you should turn it on and restart the computer. With older system software, this option may be labeled "32 Bit Addressing." This option may be absent in newer systems. Note: Turning on this option may be incompatible with some older software. If so, you can leave it off. NucLear MAC will run properly, but you may not be able to use all of the memory installed in your computer.

Scientific Imaging recommends that you disable Virtual Memory if you have sufficient random-access memory (RAM) installed in your computer. Virtual Memory uses the computer's hard disk to make it appear that the computer has additional RAM. This can make the computer behave sluggishly. Also, in NucLear MAC data-acquisition systems where the simultaneous method is used, the entire program must be loaded into the existing RAM. With Virtual Memory active, you can get an error of type -620 when starting an acquisition, which means that there is not enough RAM remaining to load the whole program. To prevent this error, disable Virtual Memory and/or decrease the partition size for the program (Step 6 below).

Step 4—Check the monitor settings

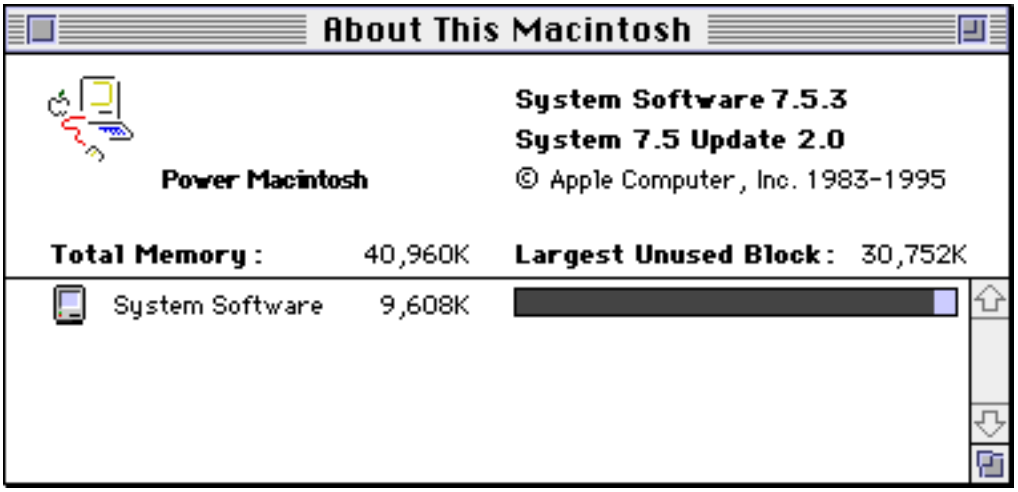
Your Macintosh computer has one or more monitors connected to it. You can use the Monitors Control Panel to adjust the settings for each monitor. Depending on the monitor and its associated video card, various numbers of colors can be displayed. Although NucLear MAC can run regardless of how the monitors are set up, it is optimized for the case of 256 colors. So, for best performance, use the Monitors Control Panel to adjust each monitor to display 256 colors.

Step 5—Load the program

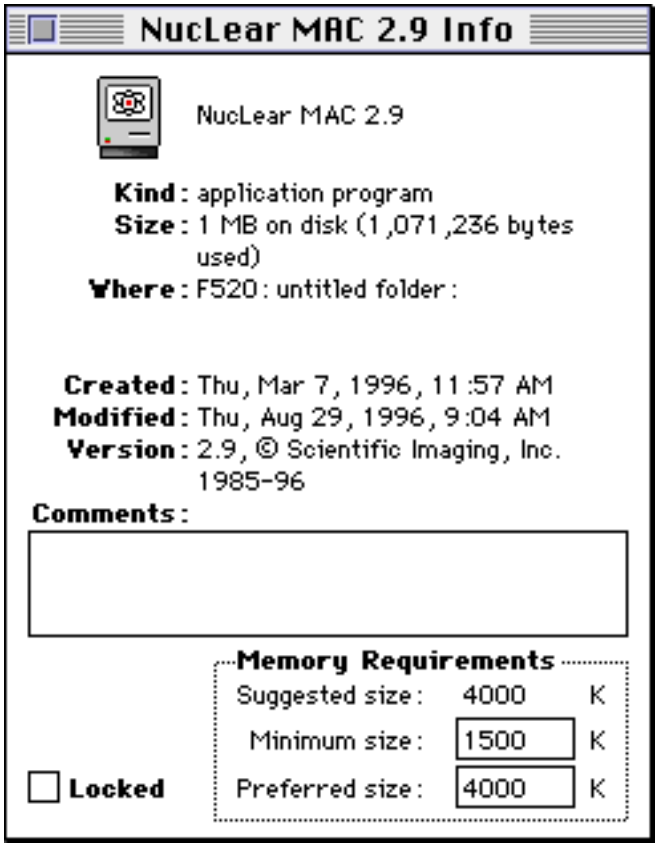
Locate the disk containing the appropriate NucLear MAC program version for your computer. Load the program onto your computer's hard disk. The program may be compressed into a self-extracting archive, in which case a ".sea" extension is appended to the name. To load the archive, simply double-click on it and select your hard drive as the destination for the decompressed program.

Step 6—Check the program memory

Adjust the memory allocation of NucLear MAC to suit your needs. First, from the Finder, choose the menu command titled **About This Macintosh...** It should be the first item under the **Apple** menu. There should be only one entry in the list: System Software (With older system software, the Finder may appear in this list also).



If any other applications are running, then quit them. Make a note of the size of the "Largest Unused Block." This is the maximum amount of memory that the NucLear MAC program can use. A "K" is 1024 bytes. A megabyte or "M" is 1024K. Select the NucLear MAC icon on your hard drive by clicking on it once. Choose the **Get Info...** command from the **File** menu of the Finder. You should see a window similar to the following:



The "Preferred size" in the lower right is the amount of random-access memory that is allocated for NucLear MAC, provided that much is available. This is the number you

can adjust. There is no specific memory requirement for the program; optimal values depend on the type of images that are acquired and the type of processing. The larger the image size, the greater the number of frames, and the more complex the processing, the more memory will be needed. Some general suggested minimums for various uses are:

Acquire Only: 4000K
Basic Planar: 5000K
SPECT: 8000K
Gated SPECT: 12000K

If the minimum size for your intended use is greater than 500K plus the Largest Unused Block size, then you should probably install more memory in your computer. If, after setting the Preferred size for NucLear MAC, you frequently get “Out of Memory” errors while processing, then you can always go back and increase the Preferred size (when the program is not running). Note that other application programs can be run at the same time as NucLear MAC, but they require memory from the unused block, possibly not leaving enough for NucLear MAC.

Important Note: Scientific Imaging has found a possible problem with the system software memory allocation under Systems 7.0 and 7.0.1 with or without the memory tune-up. The problem can cause the Finder to run out of memory and cause error messages that say certain Finder actions cannot be completed. Apple has fixed this bug in operating systems 7.1 and beyond. Therefore, Scientific Imaging strongly recommends that users upgrade their system software to at least 7.1, and preferably, to 7.5.

Step 7—Make the NucLear MAC folder

If you are running System 7 or later, then you may wish to put the NucLear MAC program and related files in a separate folder. This reduces clutter on your hard disk, and you can make a NucLear MAC alias file to activate the program. Make a new folder titled “NucLear MAC Folder” or another name of your choosing. Drag the NucLear Mac application into the folder. Open the folder and select the NucLear MAC application. Choose **Make Alias** from the **File** menu. Drag the alias file onto your hard disk icon or to the desktop. Close the NucLear MAC folder. From now on you can activate the program by double-clicking on the alias.

Step 8—Load the user functions and macro examples

If desired, load the “User Functions” and “Macro Examples” folders onto your hard disk, preferably in the same location as the NucLear MAC program. These folders may be in a self-extracting archive on one of the installation disks.

Step 9—Load the provided filters

Activate the NucLear MAC program by double-clicking on its icon or that of its alias. You should receive a query asking if it is OK to create a new preferences file. If the program does not run, then make sure that either the acquisition boards or the software

key are correctly installed. If the program still doesn't run, then contact your dealer for assistance.

Locate the icon for the NucLear MAC document titled "NucLear MAC filters." It should be on one of the installation disks. Double-click on the icon. The NucLear MAC program should come to the front. It loads the filters automatically. These filters are kernels suggested by Scientific Imaging. You can add your own filters at any time. There are three types of filters: spatial, temporal, and SPECT.

Step 10—Enable the user functions and macros

If you loaded the user function and macro folders in Step 8, then you should now show the NucLear MAC program where they are located. Under the **Analyze** menu in NucLear MAC, go down to the **User Functions** submenu. Select the **Set Directory...** item. Locate the User Functions folder you copied to your hard disk. Select the folder, then click on the "Set Directory" button. Next, under the **Macros** menu in NucLear MAC, go down to the **Create** submenu and select the **Set Directory...** item. Locate the Macro Examples folder you copied to your hard disk. Select the folder, then click on the "Set Directory" button. There should now be a list of user functions under the **User Functions** submenu, and a list of macros under the Macros menu.

Step 11—Load the optional color tables

The NucLear MAC program comes with five color tables installed. In addition, Scientific Imaging provides sets of color tables from a number of different vendors. These can be found in the Color Tables folder on one of the installation disks. This folder may be in a self-extracting archive. You can load any of these color table sets by double-clicking on the file, as you did when loading the filters. To inspect the installed color tables, choose **Colors** from the **Windows** menu. Click in the color icon at the top of the color bar window. This creates a popup menu listing all the installed color tables. Select any name from the menu to show that color table in the color bar.

You may want to delete color tables you don't use. To do this, choose **Custom Colors** from the **Windows** menu. The **Name** popup menu lists all the installed color tables. From the popup menu, select the color table you want to delete. Next, from the bottom of the same popup menu, choose **Delete** and confirm that that is what you want to do.

Step 12—Set the important program parameters

If this is a data-acquisition station, then set these image-acquisition parameters:

a) Enter the station location information. Bring up the Location dialog by selecting **Location...** from the **Edit>Preferences** submenu. Enter the listed information, including institution, department, camera type, and manufacturer. When finished, click on the "Set Default" button to permanently retain the location information.

b) Enter the desired acquisition protocols. Bring up the particular acquisition dialog by selecting its name from the **Acquire** menu (**SPECT Series…** for example). Consult your manual for additional information on the various acquisition parameters. See the appendix titled “Clinical Acquisition Protocols” for examples. Change the parameters to create a particular protocol, then choose **Add…** from the bottom of the **Type** popup menu. Enter an appropriate name for the protocol. The protocol is retained, and can be accessed by selecting its name from the **Type** popup menu.

c) Set the auto-scale voltage ranges and image orientation. Perform a single-image flood acquisition with a zoom of one, and only enough counts to make a fairly uniform image. If your camera is putting out X and Y voltages in the correct ranges, then the flood image, whether circular or rectangular, should be centered in the image window and surrounded by a border containing no counts. Estimate the maximum ratio of the flood dimensions to the window dimensions. For example, if the window is three inches in the vertical direction and the border containing no counts is half an inch at the top and bottom, then the ratio is (3-.5-.5)/3, which is 2/3. Multiply the ratio by 2.5, and round up. In this example the result is 1.7. Bring up the Acquisition Settings dialog by selecting **Settings…** from the **Acquire** menu. Call your calculated value cv. You should enter -cv to cv for both the X and Y auto-scale voltage ranges. In the example, you would enter -1.7 and 1.7 respectively.

Perform another flood acquisition. If the image is clipped, then you should increase the magnitudes of the auto-scale voltages. If the border is still too large, then you should decrease the magnitudes of the auto-scale voltages. You should not use auto-scale voltages with magnitudes less than one. This would indicate that your camera is putting out X and Y signals that are too small.

Enter the computer’s persistence-scope mode and create a non-symmetric image. This can be done by partially blocking the source. Compare the computer’s P-scope to the camera’s P-scope. If the computer’s image is reversed in the X direction, then you should enable the “Reverse X” check box in the Acquisition Settings dialog. If the computer’s image is reversed in the Y direction, then you should enable the “Reverse Y” check box.

Click on the “Set Default” button in the Acquisition Settings dialog to make the above changes permanent.

d) Perform a pixel size calibration. Acquire a single image of two point sources separated by a known distance in the X direction. Do not let the image saturate. Bring up the Calibrate Pixel Sizes dialog by selecting **Pixel Sizes…** from the **Acquire** menu. Click on the “X Script…” button and follow the instructions. Repeat the above, except separate the point sources in the Y direction and use the “Y Script…” button. Click on the “Save Values” button in the Calibrate Pixel Sizes dialog to save the values. If you are concerned about variations in pixel size for different collimators, then you can perform the calibration for each collimator you are using. You must then do your acquisitions using the Acquire Any Study… dialog. This dialog allows you to define records for each collimator.

e) For SPECT cameras, calibrate the center of rotation. Acquire a 360° SPECT study of a point or line source. The study must have an even number of equally spaced views. Do not let any frame saturate. Select the **Auto-Scale Script...** command from the **Analyze>COR** submenu and follow the directions. Click on the "Set Default" button in the resulting dialog to save the COR value permanently.

f) For dual-Z (energy or head) systems, adjust the dual-energy threshold voltage so that either two energies or two heads are detected. Bring up the Hardware Settings dialog by selecting **Hardware...** from the **Acquire** menu. Type in thresholds in the range of one to two volts until a single-image dual-Z acquisition works properly: that is, with one head/energy displayed in the top of the image window, and the other head/energy displayed in the bottom. Click on the "Set Default" button in the Hardware Settings dialog to permanently save the threshold voltage.

If this is a processing station, then set these processing parameters:

In cases involving SPECT processing, the default processing parameters do not produce useful results. For these you must enter the settings you desire.

You should start by defining one or more protocols for SPECT reconstruction. Choose **SPECT...** from the **Analyze** menu to access the SPECT Reconstruction dialog. At a minimum, you should set the Source prefilter to one of the Metz spatial filters and set the Reconstruction kernel to a ramp. If you loaded the filters in Step 9, then the Metz and ramp filters are available under the respective popup menus. It is recommended that you leave the "Override saved angles" option unchecked except for special cases. SPECT studies acquired by NucLear MAC should have the correct angles saved in the file, provided the acquisition protocol was correct. Add the SPECT reconstruction protocol by selecting **Add...** from the bottom of the **Type** popup menu. Enter an appropriate name for the protocol when prompted. You may also want to click the "Set Default" button in the SPECT Reconstruction dialog to make the protocol permanently available as the default reconstruction method.

You may find that the supplied Metz filters are sub-optimal for the data generated by your camera. If this is the case, then you should experiment with the interactive SPECT reconstruction in order to design a filter more suited to your system. The following equation-based filters are included in the NucLear MAC filters file: Butterworth, Lear, and Step Lowpass. You can adjust the parameters of these filters to obtain the best results for your system. Then you can save the optimal filter as a regular spatial filter to be used automatically in your SPECT reconstructions.

If you will be generating bull's-eye plots, then you should inspect the Bull's-eye Plot Settings dialog. Choose **Settings...** from the **Analyze>Bull's-eye Plotting** submenu. At a minimum, you should select a SPECT Reconstruction protocol from the **Script/SPECT** popup menu. You may also wish to specify X-Y and Z filtering. You might start with Nema filtering for both and experiment with more aggressive filters later. Click on the "Set Default" button to permanently save your changes.

If you will be processing gated-SPECT data, then you should view the Gated-SPECT Script Settings dialog by selecting **Settings...** from the **Analyze>Gated SPECT**

submenu. At a minimum, you should select a SPECT Reconstruction protocol from the **SPECT Reconstruction** popup menu. Click on the “Set Default” button to permanently save your changes.

Step 13—Back up your preferences file

Quit the NucLear MAC program. Locate the NucLear MAC preferences file. For system software prior to version 7, the preferences file is located in the System Folder. For System 7 and beyond, the file is located in the Preferences folder, which is in the System Folder. The file is named “NucLear MAC Prefs”. Copy the file to a floppy disk, but do not remove it from its current location. As you add more protocols to your NucLear MAC system, you should repeat the back-up procedure. This protects you in the event the file becomes damaged.

2.4 Software Update Installation

This document describes how to update existing NucLear MAC software on a Macintosh or MacOS-compatible computer. These instructions also apply to updates involving the NucLear Power application.

The name "NucLear MAC" in this document is used to refer to either NucLear MAC or NucLear Power software. If you are currently running a version of NucLear MAC, then you should install the new NucLear MAC program. If you are currently running NucLear Power, then you should install the new NucLear Power program.

NucLear MAC software is intended to be backward compatible. In other words, a new version of NucLear MAC software can use the parameters of a previous version. HOWEVER, SOFTWARE IS NOT FORWARD COMPATIBLE—AN OLD VERSION OF SOFTWARE MAY NOT BE ABLE TO USE THE PARAMETERS OF A NEW VERSION!

The above fact is critical in the process of loading a new version of the NucLear MAC program. When the new version is loaded and activated, it uses parameters in the old "NucLear MAC Prefs" file as a starting point. It then adds its new features to the "NucLear MAC Prefs" file. These new features are **not** necessarily compatible with the older NucLear MAC program. **Therefore, once you use a new version of the NucLear MAC software, do not use an older version on the same computer!** Otherwise, the "NucLear MAC Prefs" file can be damaged resulting in loss of stored parameters.

The following sequence should be used to safely load new software.

1) Make a backup copy of the existing NucLear MAC program on a floppy disk.

2) Remove the existing NucLear MAC program from the computer's hard disk by putting it in the trash and emptying the trash.

3) Copy the "NucLear MAC Prefs" file onto the backup floppy disk. This file is located in the Macintosh "System" folder (or with System 7.0 or later the "Preferences" folder within the "System" folder). Do NOT remove the "NucLear MAC Prefs" file from the "System" or "Preferences" folder—just copy it to the floppy.

4) You now should have a floppy disk containing copies of the old NucLear MAC program and the current "NucLear MAC Prefs" file. PUT IT IN A SAFE PLACE.

5) The "System" folder on the computer's hard disk should still have the old "NucLear MAC Prefs" file in it. However, there should be no old versions of the NucLear MAC program on the hard disk. Note: It is important that only one version of the NucLear MAC program be kept on the main hard disk to prevent accidentally running an old version after a new version has been run and to prevent two versions from accidentally being run at the same time.

6) Load the new NucLear MAC program onto the hard disk from its floppy disk. The program may be compressed into a self-extracting archive, in which case a ".sea" extension is appended to the name. To load the archive, simply double-click on it and select your hard drive as the destination for the decompressed program. Be sure to load the new version of the same type of software that you had been using (NucLear MAC, NucLear MAC SANE, or NucLear Power). If your old NucLear MAC program was in a folder then put the new NucLear MAC program in the same folder.

7) Adjust the memory usage of the newly installed NucLear MAC program. Use the "Get Info" command of the Finder to set the preferred size for the NucLear MAC program to the same value used by the older software. If these instructions are not clear, then refer to the chapter titled "Software Installation."

8) Activate the new NucLear MAC program.

9) Most or all of your previous settings should now have been transferred to the new program automatically. With some updates, not all settings can be retained, so you should check them.

10) Select **About NucLear MAC...** from the **Apple** Menu. Click on the "What's New" button in the resulting dialog. Read the on-screen documentation describing changes in and additions to NucLear MAC. Quit the program.

11) SAVE THE UPDATED "NucLear MAC Prefs" FILE ONTO A NEW BACKUP FLOPPY DISK AS A SAFETY PRECAUTION. Periodically, as you add protocols, backup the "NucLear MAC Prefs" file onto floppy disks. Remember not to remove this file from the "System" (or "Preferences") folder, however.

3 NucLear MAC Basics

3.1 NucLear MAC Menus

To do useful work with the NucLear MAC application, you select commands from the various NucLear MAC menus. When the NucLear MAC application is active, the menu bar should contain these menus:

If these menus are not present, then some application other than NucLear MAC is active. If you are ever unsure about the active application, then click on the small icon in the far right of the menu bar. This displays the Applications menu. The items below the gray line are the currently running application programs. The item with a check mark by it is the active application. When you select an inactive application from the Applications menu, that application is brought to the front; its windows, if any, are made visible, and the menu bar changes to show the menus offered by that application.

3.1.1 Submenus

NucLear MAC uses hierarchical menus to keep individual menus from becoming too long or complex. This is accomplished by having some menu items select submenus. Submenu items are identified by the small black triangle to the right of the item name. For example, in NucLear MAC the **Preferences** item is a submenu of the **Edit** menu. In this text, submenus are indicated by the use of the greater-than symbol. For example: **Edit>Preferences**.

3.1.2 Command Key Equivalents

Many menu items have command-key equivalents. You can access these items without using the mouse. Instead, you hold down the command (⌘) key while typing the appropriate character. The command characters are displayed to the right of the menu items. Many of these commands are the same for all MacOS applications. For example, typing “Q” while holding down the command key is always used to stop running the active application (Quit).

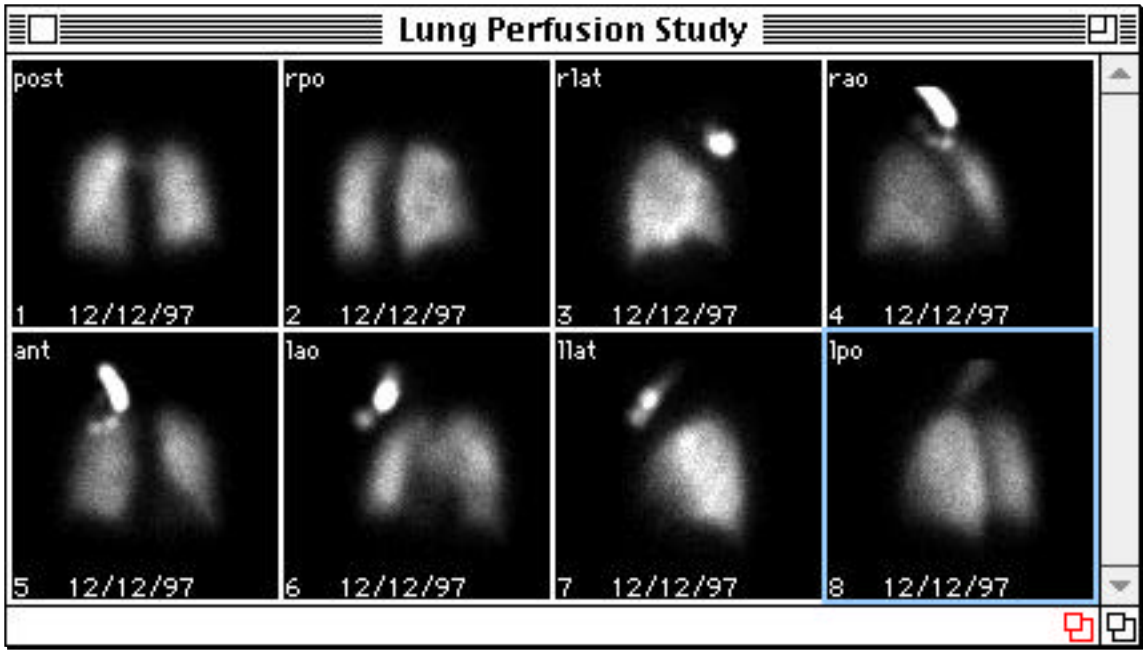
3.2 NucLear MAC Windows

The NucLear MAC application has several types of windows that you use in the course of acquiring, processing and displaying nuclear medicine data. The following subsections describe the various NucLear MAC windows.

3.2.1 Study Windows

Study windows are used to display nuclear medicine images, regions of interest, and other overlays. The contents of study windows can be saved to a disk file having the same name as the study window. When you later open a saved study file, NucLear

MAC opens a corresponding study window which displays the study images and overlays.



NucLear MAC Study Window in All-Frames Format

3.2.1.1 Images

Computerized images for nuclear medicine typically consist of arrays of numbers. These arrays are arranged into rows and columns of elements. When we speak of a 64 by 64 image, we mean there are 64 rows of 64 elements each, for a total of 4096 elements. For acquired nuclear medicine data, each element in an image matrix corresponds to the number of high-energy photons received by a small area on the face of the gamma camera detector.

In order for us to view computerized nuclear medicine image data, the elements the images contain are mapped onto a computer video monitor. A computer monitor consists of a matrix of pixels. A typical 13" monitor with video card displays 480 rows of 640 pixels. The brightness and color of each pixel can be set independently. The individual elements in an image matrix are often called pixels, but pixels in a nuclear medicine image do not correspond directly to video monitor pixels. To distinguish the two types of pixels, they are referred to in this manual as image pixels and screen pixels, respectively.

NucLear MAC saves the elements of nuclear medicine image data as either bytes or words. This "depth" determines the range of values possible in each element of an image. A byte-sized element can hold values ranging from zero through 255, while a word-sized element can hold values ranging from zero through 65535. The **only** time NucLear MAC changes the values of elements in an existing image is during data acquisition of that image. Adjustments affecting how an image is displayed (screen pixels) do not change the underlying data (image pixels).

3.2.1.2 Studies

A NucLear MAC study consists of one or more images, as well as extra information. Single images in a study may also be referred to as frames. All images in a single study have the same number of elements and all their elements have the same depth. In addition, the *resolutions* of all frames in a single study must match. That is, they all have the same number of rows and columns.

NucLear MAC studies are displayed in separate windows on your monitor. If a study has been saved to disk, then the name of the window containing the study is the same as the name of the disk file.

Every study currently open in the NucLear MAC application is listed under the **Windows** menu. Selecting the name of a study from the menu activates the study; the study window is brought to the front, allowing you to apply other commands to the study.

3.2.1.3 Display Formats

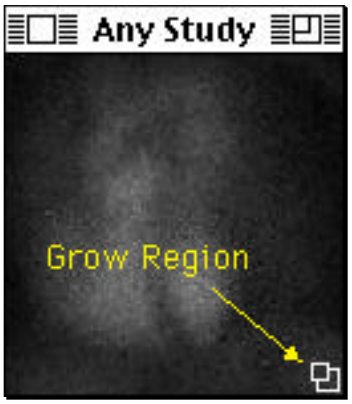
Every study containing more than one image frame can be displayed in one of two formats. In **Movie Format**, only one frame of the study is visible at a time. In **All-Frames Format**, as many frames as will fit your monitor are displayed. You can switch between display formats for an active study window by choosing the item in the **Display** menu with the format name: **Movie Format** or **All-Frames Format**.

Movie Format

Typically the movie format is used for studies whose frames have the same view imaged at different times, such at gated and dynamic studies. The movie format is also used to save on screen space, as windows in movie format are smaller.

Window Size

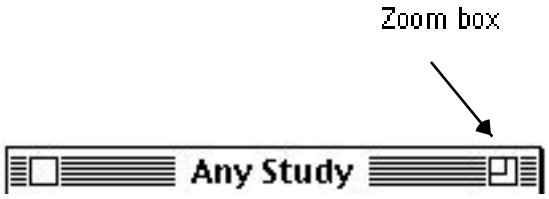
For a study window in movie format, the size of the displayed image frame is directly proportional to the size of the window. You can adjust the window size by dragging the window grow region, located in the bottom, right of the study window. By changing the window size, you are changing how many screen pixels are used to display each image pixel. For example, if the study image matrix is 64 by 64, but the display window on your monitor is 128 by 128 pixels, then four screen pixels are being used to display each image pixel.



When you resize a study window, the window dimensions are constrained to the same aspect ratio as the underlying image matrix. That is, if the image matrix is square, then the window is constrained to be square as well. You can override this behavior by holding down the option key while resizing.

You can constrain the size of the study window to integer multiples of the underlying image matrix by holding down the shift key while resizing. In the case of an image matrix of 64 by 64 elements, the shift-key constraint would limit the window size to N times (64 by 64) screen pixels, where N is an integer greater than zero.

You can also change the size of an active study window by clicking in its zoom box, located in the right of the window's title bar. Clicking the window's zoom box toggles the window size between two states. The standard states for study windows are the base size, which is the size of the image matrix, and twice the base size. You can change the zoom states in the Image Defaults dialog, which is accessed through the **Edit>Preferences>Image Defaults...** menu command.



Clicking in the zoom box causes a different result when either the option or the command (⌘) key is held down. These modifier keys cause the window size to be halved or doubled, respectively.

Active/Visible Frame

Sometimes you will perform operations that apply to only one frame of a study—Paste, for example. When a study window is in movie format, these operations apply to the visible frame. When the study window is active, the Controls window displays the number of the active frame in a box.

Controlling the active/visible frame

There are several commands in the **Display** menu that operate only on active, multi-frame study windows in movie format. These commands all change the image frame currently visible in the study window. You can also use the Controls window to change the current image frame.

Step Forward

The **Step Forward** menu command causes the next image frame to be displayed in the active study window. For example: if the fifth frame of a study with ten frames is visible, then the **Step Forward** command causes the sixth frame to be displayed. If the last study frame is the visible frame when **Step Forward** is applied, then the first study frame is displayed.

Step Backward

The **Step Backward** menu command causes the previous image frame to be displayed in the active study window. For example: if the fifth frame of a study with ten frames is visible, then the **Step Backward** command causes the fourth frame to be displayed. If the first study frame is the visible frame when **Step Backward** is applied, then the last study frame is displayed.

Run/Halt

The **Run** menu command causes the visible image frame of a study window in movie format to continuously advance to the next frame, as if you were constantly applying the **Step Forward** command. You can adjust the speed of the cinegram with the "Slow/Fast" slider control in the Controls window. When the active study window is in the "run" mode, the menu command changes to **Halt**. Choosing **Halt** stops the study from automatically advancing the visible image frame.

Mouse Track

The **Mouse Track** menu command enters a mode where moving the mouse left or right changes the current visible frame of the active study window. Moving the mouse to the left decreases the frame number, as if you constantly applied the **Step Backward** command. Moving the mouse to the right increases the frame number, as if you constantly applied the **Step Forward** command. If while mouse-tracking your cursor reaches the edge of the screen, you can hold down the shift key on the keyboard to move the cursor back to the center of the screen without changing the current image frame.

All-Frames Format

The all-frames window format is used to review many or all frames of a study at once. When you switch a study window to all-frames format, the individual frames stay the same size, but the window is enlarged so that multiple frames are visible. In addition, the individual frames are numbered in their bottom, left corners. If the window isn't

large enough to display all the frames, then a scroll bar is visible along the right side of the study window. You can use this control to scroll up or down to view all the study frames. If you have an extended keyboard, then you can use the "home," "end," "page up" and "page down" keys to navigate instead of the mouse.

Window Size

Study windows in all-frames format have two grow regions in the bottom, right corner. Clicking and dragging in the right-most grow region simply controls the size of the window. The size of the image frames does not change. When the window is resized, the image frames are rearranged to fit as many per row as possible at their current size.

Dragging in the left grow region controls both the window size and the size of the image frames. The image frames are resized in proportion to the size change of the window. With this type of resizing the window dimensions are constrained to the same aspect ratio as the original window. You can override this behavior by holding down the option key while resizing. You can constrain the size of the study window to integer multiples of the original window size by holding down the shift key while resizing.

Active Frame

Because multiple frames can be visible in an all-frame format window, highlighting indicates the active frame of an active study window. The narrow band around the active image frame is drawn in the current system highlight color. Commands requiring a single image frame, such as Paste, are applied to the active frame. When the study window is active, the Controls window displays the number of the active frame in a box.

Controlling the active frame

You can change the active frame of an active study window in all-frames format by simply clicking in an inactive frame. The inactive frame is activated, while the previously active frame is deactivated. You can also change the active frame with the Controls window.

3.2.1.4 Color and Brightness Mapping

NucLear MAC provides two methods for controlling the brightness of an image pixel when it is displayed as screen pixels in a study window. The two methods, Data Mapping and Color Table Mapping, are combined to produce the study window display.

Data Mapping

Data mapping refers to the process of converting image pixel values to a range of 256 indexes used for screen display. The data mapping for each study window can be adjusted separately. Because the mapping is linear, the equation involved is simple. Given an image pixel value p, the index is computed as follows:

tmp = scale * (k1 * p - k2)

index = 0 for tmp < 0

index = tmp for 0 ≤ tmp ≤ 255 (linear range)

index = 255 for tmp > 255

The user can adjust the scale (slope) and k2 (linear range start) values for optimal viewing. These adjustments are made using the Controls window, as described later. The scale value is allowed to range from zero through 255. The value of k1 is one for byte-depth studies and 255/(maximum study pixel value) for word-depth studies.

When a study is mapped in inverse mode, the first mapping equation is changed to:

tmp = 255 - scale * (k1 * p - k2)

When a study window is active, its current mapping equation is graphed in the Controls window.

Color Table Mapping

After the image pixel data are mapped to 256 indexes, these indexes are mapped to one of the current NucLear MAC color tables. Each color table is a list of 256 colors or gray-scale values, and each color corresponds directly to one of the indexes. You can activate up to four color tables at one time. You can assign to each study window any one of the active color tables. A floating color bar window represents each color table. By repeatedly selecting **Colors** from the **Windows** menu, you can make visible and select each of the color bar windows in turn.

You can define groups of color tables in the Color Tables dialog, which is accessed through the **Edit>Color Tables>Settings...** menu command. Although you can load previously defined color table sets with the Color Tables dialog, it is usually faster to click in one of the color bar windows. See Color Bar Windows below.

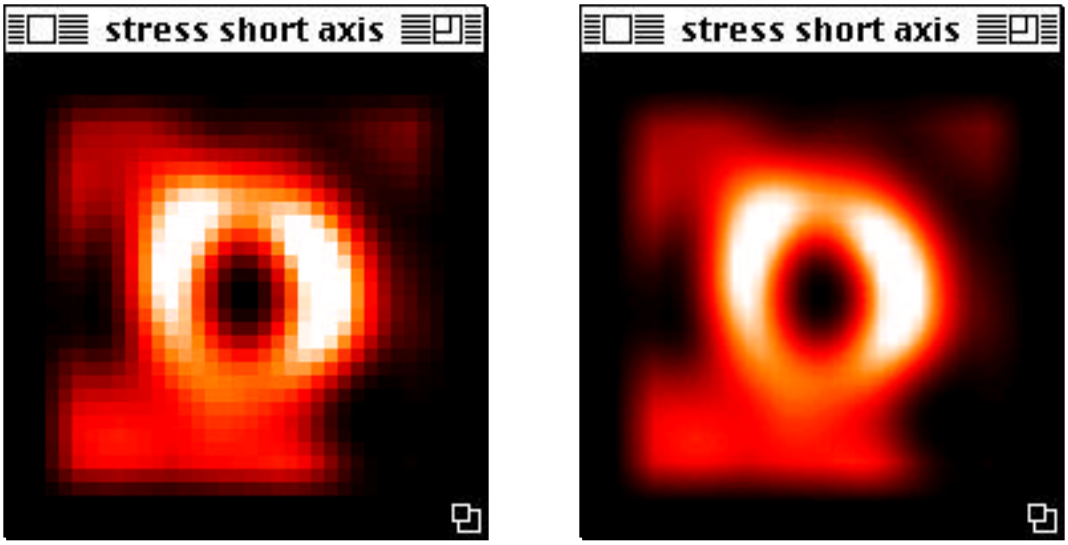
You can use a color bar window to change the color table it displays. These changes also affect the appearance of study windows currently using that color bar. With a color bar window, you can remap any color table in the same way that image data are mapped.

To display a study window with a particular active color table, select the study window, then choose the number of the color table from the end of the **Edit>Color Tables** submenu. The number of the color table currently used by the active study window is checked. Color table numbers correspond to their order in the list in the Color Tables dialog.

3.2.1.5 Interpolation

Interpolation is a method of smoothing an image when it is drawn in a study window. Interpolation is used only when the matrix of the study window is larger than the underlying image matrix. That is, interpolation can be used when there are more screen pixels than image pixels. Generally you will always keep interpolation enabled except

when you need to differentiate individual image pixels. As this example shows, an interpolated image display is visually more pleasing.



Non-interpolated (left) versus interpolated (right) image display

Each study window can be displayed with or without interpolation. You can set whether interpolation is used with the **Interpolation** item in the **Display** menu. This menu item applies to the active study window. If the item has a check mark by it, then interpolation is enabled for the active study window, and selecting the item disables interpolation. If the item does not have a check mark by it, then interpolation is disabled for the active study window, and selecting the item enables interpolation. You can set the default interpolation state for new studies in the Image Defaults dialog, which is accessed through the **Edit>Preferences>Image Defaults...** menu item.

3.2.1.6 Zooming

Sometimes the visible image frames in a study window show only a portion of the underlying image matrixes. This happens when you have zoomed in on a portion of an image with the magnifying tool. This is similar to simply resizing the study window, except that with window resizing you are limited to the size of the monitor, and the entire image remains visible. Use the hand tool to expose hidden portions of a zoomed image frame.

3.2.1.7 Overlay Objects

Overlay objects are items drawn over images in study windows. These items change only the visual display; they do not affect the underlying image data. Objects are saved with a study when you save that study as a disk file. When a study window is resized, any objects it contains are redrawn in proportion to the window. This changes the object sizes as measured in screen pixels, but not the sizes as measured in image pixels.

For information on placing objects in study windows, please refer to the Tools window section. Overlay objects respond to the standard edit commands, which are described in the Editing chapter.

Overlay object characteristics

Overlay objects have several common characteristics. You can change these characteristics as follows:

Object selection

Objects can be selected or not selected. You need to select objects in order to apply certain commands to them, such as **Copy**. You can simultaneously select any number of objects provided the following conditions hold:

1. The objects are located in the active frame of the active study window.
2. The objects are visible.
3. The objects are not locked.

You can tell what objects are selected because selected objects have *resize areas* that continuously scroll a black and white pattern.

To deselect all objects in the active frame of the active study window, click the arrow cursor in the frame, but outside of any object.

To select a single object, click on the object with the arrow cursor. Alternatively, click the arrow cursor outside of any object, then drag out the *selection rectangle* so that it intersects the object you want to select, then release the mouse button. The cursor changes to  when you are dragging out a selection rectangle.

To select additional objects, hold down the shift key while selecting the additional objects, as above.

To select multiple objects, you can select them consecutively, as above. Alternatively, you may be able to click the arrow cursor outside of all objects, then drag the selection rectangle to intersect those objects you want to select, then release the mouse button.

To select all objects in the active frame of the active study window, choose **Select All** from the **Edit** menu.

To deselect certain selected objects, while leaving others selected, hold down the shift key while clicking the arrow cursor on the objects.

Object visibility

Each object can be visible or invisible. To make selected objects invisible, choose **Hide** from the **Edit>Objects** submenu. To make all objects in a study window invisible, chose **Hide All** from the **Edit>Objects** submenu. To make all objects in a study window visible, chose **Show All** from the **Edit>Objects** submenu.

Object shown in what frame(s)

Each object has a home frame. This is the study window frame in which the object was created. Therefore each object can be in one of two states that affect which frames it appears in:

1. In Home Only. The object is drawn only in its home frame.
2. In All Frames. The object is drawn in every frame of the study window.

You can change the state of selected objects by choosing the new state from the **Edit>Objects** submenu. Choose **In Home Only** to make the selected objects visible only in their home frames. Choose **In All Frames** to make the selected objects visible in every frame of the study.

Object position

You can move visible, unlocked objects within their image frames. To move all selected objects, click the arrow cursor on one of the objects (not in the resize area), then drag all the objects to the new location, then release the mouse button. Choose **Undo Move** from the **Edit** menu if you make a mistake.

To move a single object that is not selected, simply click the arrow cursor on the object, drag the object to the new location and release the mouse button. If you need to move one selected object and other objects are also selected, then first deselect all objects (click in the frame outside all objects) before moving the object.

Object size

You can resize objects that are selected. Place the arrow cursor over the resize area of one of the objects. The cursor changes to  or  . Click and drag the cursor in one of the indicated directions to make the selected object(s) larger or smaller. To limit the bounding rectangle of the object(s) to a square, hold down the shift key while resizing.

Object color

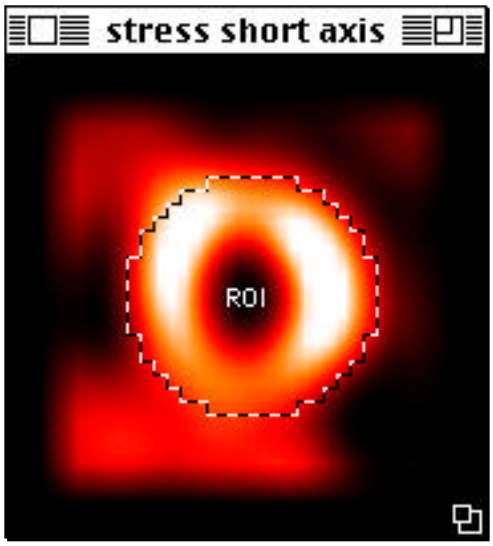
To change the color of selected objects, choose the name of the new color from the **Edit>Objects>Set Color** submenu. Read the description for individual object types to learn the exact effect of changing an object's color. You can set default colors for the different types of objects. Use the Default Colors Dialog, which is accessed through the **Edit>Preferences>Object Colors...** menu command.

Object locked or unlocked

Objects can be locked, which is a state where you cannot edit them. You can only unlock or change the visibility of locked objects. To lock objects, select them, then choose **Lock** from the **Edit>Objects** submenu. To unlock objects, activate the frame containing the objects, make the objects visible if they are not already, then choose **Unlock Frame** from the **Edit>Objects** submenu.

Regions of Interest (ROIs)

Regions of interest define specific groups of image pixels in an image matrix. You can define ROIs of any shape, and they do not need to be contiguous. In study windows, an ROI is outlined by narrow black and white lines that enclose the pixels of the ROI.



Study window with one region of interest

Each ROI has a name. To change the name of an ROI, select the ROI and choose **Change...** from the **Regions>Names** submenu. Type the new name into the Region Name dialog and click the "Set Name" button.

When applied to an ROI, selecting a different color from the **Edit>Objects>Set Color** submenu affects only the ROI name. The ROI boundary is always drawn in black and white. Set the default color for region names in the Default Object Colors dialog, which is accessed through the **Edit>Preferences>Object Colors...** menu command.

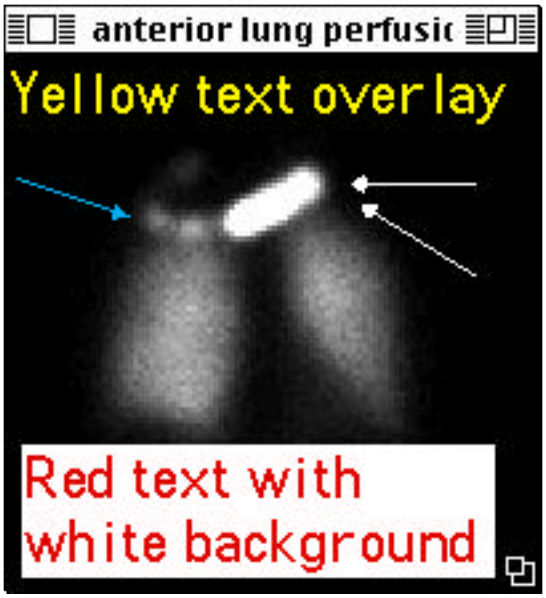
There are three submenus under the **Regions** menu that apply specifically to ROI objects in the active study window. These are **Select**, **Visibility** and **Names**. The **Select** submenu lists all ROIs present in the current study. Choosing the name of an ROI from the menu selects the ROI, making it visible if necessary, and changing to the ROI's home image frame, if necessary. The **Visibility** submenu offers one function not found in the **Edit>Objects** submenu: **Delete All**. This item permanently removes all ROIs from the active study window. Choose **Edit>Undo Clear** if you did not actually want to delete all ROIs. Use the **Names** submenu to manipulate the names of selected ROIs; you can show, hide, and edit the names.

Text

Text overlays are used to print any information in study windows. This can include study acquisition date, view name, hospital name, etc. It is common to lock text overlays so they do not interfere with operations involving ROIs.

To edit an existing (unlocked) text overlay, choose the Text tool from the Controls window, then click in the overlay and edit by typing your changes. If you are using the arrow cursor, then you can double-click in a text overlay. This is a shortcut for changing to the Text tool and clicking in the text.

To change the color of selected text overlays, use the **Edit>Objects>Set Color** submenu. Set the default color for text objects in the Default Object Colors dialog, which is accessed through the **Edit>Preferences>Object Colors...** menu command. You can also enable a background rectangle for new text overlays, as well as set the color of the background.



Examples of text and arrow overlays

Arrows

Arrows are used for annotation and measuring. To change the color of selected arrow overlays, use the **Edit>Objects>Set Color** submenu. Set the default color for arrows in the Default Object Colors dialog, which is accessed through the **Edit>Preferences>Object Colors...** menu command.

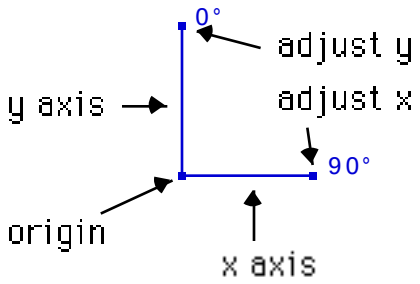
Pictures

Pictures are a standard way of conveying graphical information in MacOS. You can use the **Paste** command in the **Edit** menu to place pictures on an image. There is no restriction on the contents of the pictures, but they should be small enough that they do not obscure the image display. Some examples of pictures you may wish to overlay are hospital logos, other images, annotation, etc.

The **Edit>Objects>Set Color** submenu has no effect on picture objects.

3.2.1.8 The Axes Overlay

The axes overlay is similar to other overlay objects, but its functions are more specialized. There is only one set of axes per study window. The axes, when visible, are displayed in all frames of the study window. The axes are used primarily in processing scripts, where the user is asked to define various locations, lengths and angles.



The axes overlay

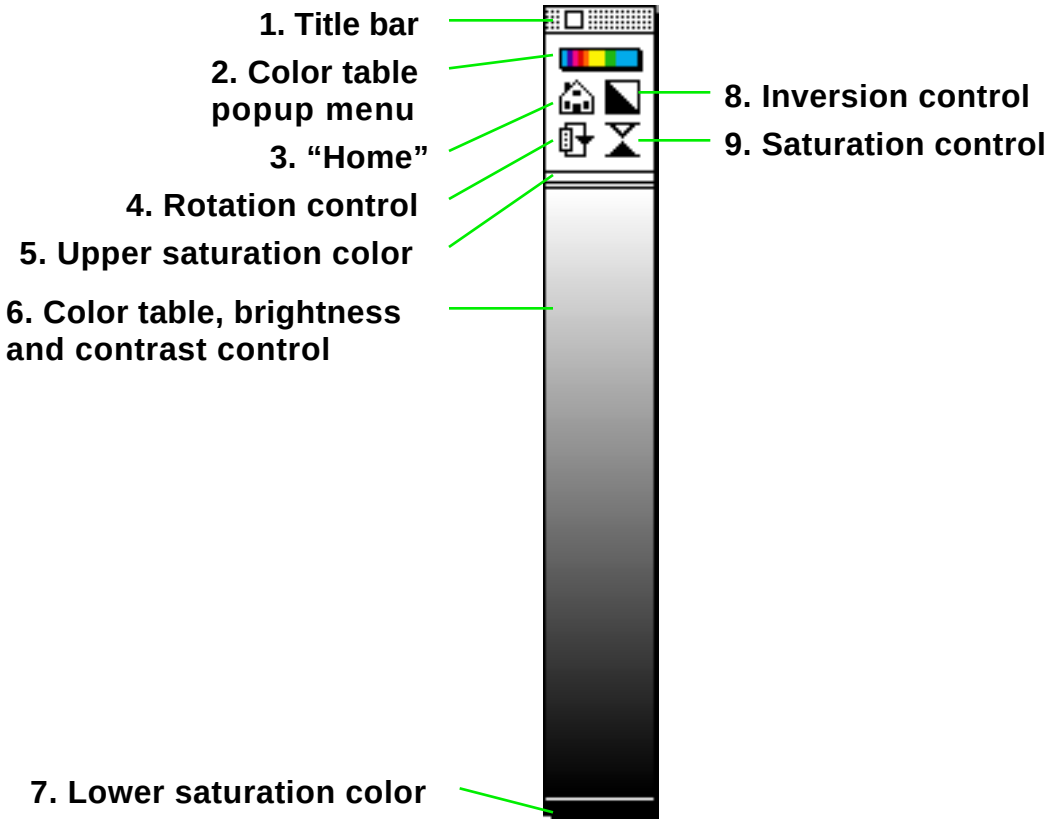
Set the color for axes overlays in the Default Object Colors dialog, which is accessed through the **Edit>Preferences>Object Colors...** menu command. This affects all axes overlays when they are next redrawn.

3.2.2 Floating Palette Windows

Floating palettes are small windows that appear to float in a layer above any active study window. These windows provide access to frequently used functions. To hide all floating palette windows at once, hold down the option key and select **Close All** from the **File** menu.

3.2.2.1 Color Bar Windows

Color bar windows display the color tables currently in use for study windows. You can use color bar windows to change the active color table and adjust brightness and contrast.



Floating palette color window

1. Title bar. Click in and drag the title bar to reposition the Color bar window. Click in the close box to hide the Color bar window. Select **Colors** from the **Windows** menu to show the Color bar window(s).

2. Color table popup menu. Click to bring up a menu containing all color tables installed in NucLear MAC. Select the desired color table to make it active in this Color bar window. Hold down the option key to select from a menu of color table sets. This can change both the number of Color bar windows and the color tables they contain. Use the **Edit>Color Tables>Settings...** menu command to define color table sets.

3. "Home." Click to revert to the default color table. The default color table is determined by the current color table set, or by the current color table in the Custom Colors dialog.

4. Rotation control. Click and drag up or down to rotate the current color table.

5. Upper saturation color. Click to determine whether the upper saturation color comes from the top (white) or bottom (black) of the color table.

6. Color table. This area displays the current color table. Click and drag up, down, left and right to change brightness and contrast. The up and down directions control the size of the linear portion of the color table. Areas outside the linear region are set to the respective saturation colors. The left and right directions control the location of the

linear portion of the color table. If your monitor is set to display 256 colors, then color table changes occur instantly. Otherwise changes occur when you release the mouse button. As a shortcut to make study windows use this color table (when there are multiple color tables), click in the color table while holding down the command (⌘) key. Then click on the desired study windows. Click twice on the last study window to end the process.

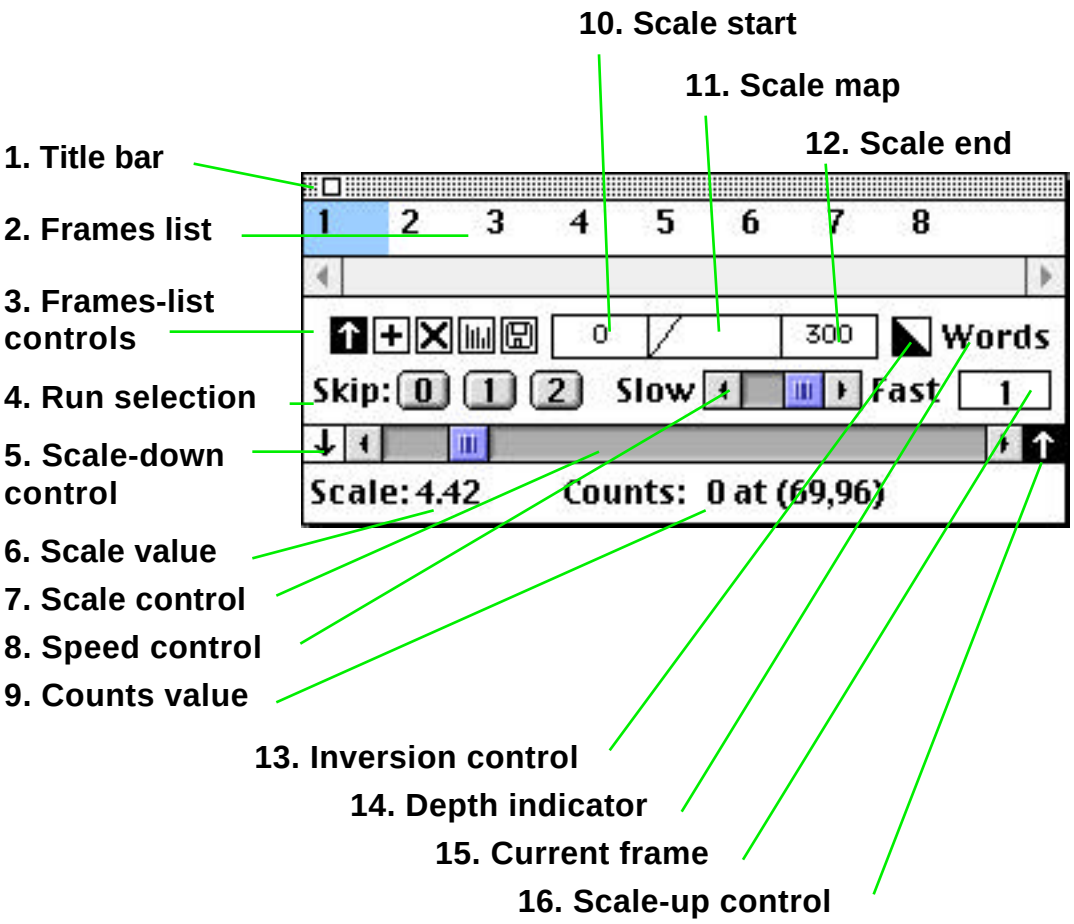
7. Lower saturation color. Click to determine whether the lower saturation color comes from the top (white) or bottom (black) of the color table.

8. Inversion control. Click to invert the color table. With a gray-scale color table, black becomes white, white becomes black, etc.

9. Saturation control. Click and drag down to bring the upper saturation color down into the color table. Click and drag up to bring the lower saturation color up into the color table.

3.2.2.2 Controls Window

The Controls window provides information about the active study window and lets you change view parameters of the same. The Controls window becomes inactive (grayed out) when no study window is active.



Controls window

1. Title bar. Click in and drag the title bar to reposition the Controls window. Click in the close box to hide the Controls window. Select **Controls** from the **Windows** menu to show the Controls window.

2. Frames list. This is a list of all frames in the active study window. Use the scroll bar and arrows to view the entire list. The last entry in the list tells you how many image frames are in the active study. Clicking in the frames list performs various actions depending on the current frames list control.

3. Frames-list controls. Only one of the five controls can be selected at one time. The selection determines what happens when you click in the frames list:

Current frame selection. This makes the frames list reflect the active image frame in a study window—the highlighted item is the number of the active frame. Click on a different frame number to activate that frame. In this mode only one frame at a time can be highlighted in the frames list.

New composite selection. The image frames selected in the frames list are summed when the **Analyze>Image Math>New Composite** command is applied to the active study window. Double-clicking on the selection icon selects all

frames in the list. Hold down the command (⌘) key while clicking in the frames list to add or remove individual frames from the current selection. Hold down the shift key while clicking in the frames list to extend a selection. The default is for all frames to be included in a summation of study frames.

 Display exclusion selection. The image frames selected in the frames list are not displayed when the active study window is in movie format. This is most useful for excluding frames from a cinegram. Hold down the command (⌘) key while clicking in the frames list to add or remove individual frames from the current selection. Hold down the shift key while clicking in the frames list to extend a selection. The default is for all frames to be displayed.

 Graph inclusion selection. The image frames selected in the frames list are included in time-activity curves created from the active study window. Double-clicking on the selection icon selects all frames in the list. Hold down the command (⌘) key while clicking in the frames list to add or remove individual frames from the current selection. Hold down the shift key while clicking in the frames list to extend a selection. The default is for all frames to be included in a time-activity curve.

 Frames subset selection. The image frames selected in the frames list are extracted when the **Display>Views>Frames Subset** is applied to the active study window. Double-clicking on the selection icon selects all frames in the list. Hold down the command (⌘) key while clicking in the frames list to add or remove individual frames from the current selection. Hold down the shift key while clicking in the frames list to extend a selection. Hold down the option key while clicking in the frames list to immediately create a subset study with the selected frames. The default is for all frames to be included in the frames subset.

4. Run selection. These buttons automatically control the frame display exclusion selection. Press “0” to include all study frames in a movie-format display. Press “1” to display every second frame. Press “2” to display only every third frame.

5 Scale-down control. Clicking the scale-down control sets the scale control range to zero to one. This range is used for decreasing the brightness of image frames in the current study.

6. Scale value. This number is the slope of the (linear) brightness mapping of the active study.

7. Scale control. Use this control to adjust the image brightness. This sets the slope of the brightness mapping in the current range (zero to one or one to 255).

8. Speed control. Use this control to adjust the speed of a cinegram of the active study. Hold down the option key when clicking in the speed control to set the cinegram speed of all open studies. The default speed for newly acquired studies is set using the **Edit>Preferences>Image Defaults** command.

9. Counts value. This displays the value and location of any image pixel over which you place the cursor. The value can range from zero to 255 for byte-depth studies, and from

zero to 65535 for word-depth studies. The location is in the form (x,y), where x ranges from zero for the first (left-most) column of pixels to N-1 for the Nth column of pixels, and y ranges from zero for the first (top-most) row of pixels to M-1 for the Mth row of pixels.

10. Scale start. The scale-start box displays and controls the image pixel value used as the start of the linear brightness mapping. Image pixel values less than or equal to this start value are mapped to the bottom color of the color table. Click in the box and drag to set the scale start. Dragging right or left slowly increases or decreases the value. Dragging up or down rapidly increases or decreases the value. The scale-end value is kept fixed, while the scale value adjusts to the new start point. This is displayed in the scale map.

11. Scale map. The scale map is a graphic representation of the current brightness mapping for the active study window. Click in the scale map box and drag to simultaneously move the scale start and end points while maintaining the same scale (slope). Dragging right and left slowly increases or decreases the end values. Dragging up and down rapidly increases or decreases the end values.

12. Scale end. The scale-end box displays and controls the image pixel value used as the end of the linear brightness mapping. Image pixel values greater than or equal to this end value are mapped to the top color of the color table. Click in the box and drag to set the scale end. Dragging right or left slowly increases or decreases the value. Dragging up or down rapidly increases or decreases the value. The scale-start value is kept fixed, while the scale value adjusts to the new end point. This is displayed in the scale map.

13. Inversion control. Click on the inversion control to invert the brightness mapping for the active study window. In general this exchanges dark and light values.

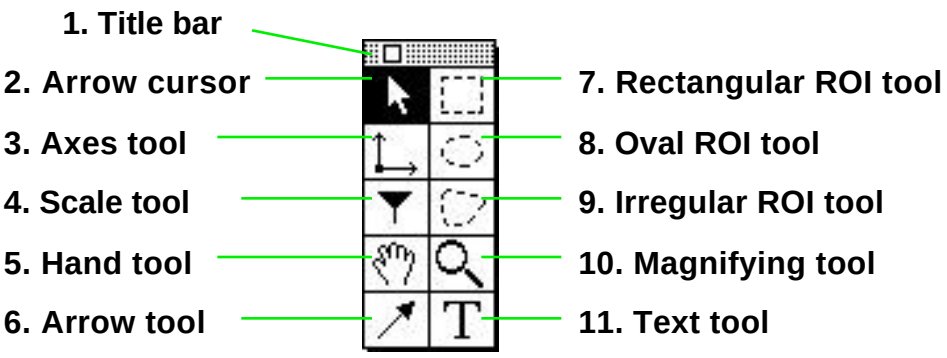
14. Depth indicator. This area displays "Byte" when the active study window contains images having one byte (eight bits) per image pixel. The area displays "Word" when the active study window contains images having two bytes (16 bits) per image pixel.

15. Current frame. This box displays the number of the active frame in the active study window. If a frames-list control other than "current frame selection" is enabled, then clicking in the current frame box selects the number of the current frame in the frames list.

16. Scale-up control. Clicking the scale-up control sets the scale control range to one to 255. This range is used for increasing the brightness of image frames in the current study.

3.2.2.3 Tools Window

The Tools window lets you select from different tools that you use to operate on the active study window. The cursor changes to reflect the current tool when the cursor is placed over an active study window.

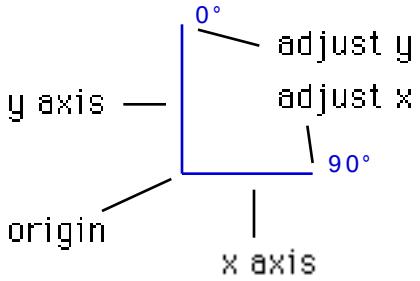


Tools Window

1. Title bar. Click in and drag the title bar to reposition the Tools window. Click in the close box to hide the Tools window. Select **Tools** from the **Windows** menu to show the Tools window.

2. Arrow cursor. Use the arrow cursor to manipulate objects in active study windows. This includes selecting, moving and resizing objects. You can analyze certain types of objects by double-clicking on them with the arrow cursor. If some other tool is active, you can temporarily switch to the arrow cursor by holding down the command (⌘) key. When resizing, if you hold the command (⌘) key down, then the resizing is based on the center of the selection. If you hold down the shift key, then the resizing is constrained to square proportions.

3. Axes tool. When the axes tool is selected, the cursor changes to when it is over an active study window. Click, drag and release to place the axes in a specific location. If the axes were not visible, they are made visible in all frames of the study window. To hide visible axes in the active study window, select **Hide Axes** from the **Edit** menu. To show invisible axes in the active study window, select **Show Axes** from the **Edit** menu.



The axes overlay

When the axes are visible in a study window, you can manipulate them with the arrow cursor. Place the arrow cursor over the origin and it changes to . Click, drag and release to move the axes. Place the arrow cursor over the end of the y-axis, near the zero degree point. The cursor changes to . Click, drag and release to change the length and angle of the y-axis. Place the arrow cursor over the end of the x-axis, near

the 90-degree point. The cursor changes to  . Click, drag and release to change the length and angle of the x-axis.

4. Scale tool. When the scale tool is selected, the cursor changes to  when it is over an active study window. Click, drag and release to change the data mapping for the study window such that value of the image pixel under the cursor is used as the saturation point (point of maximum brightness). This is similar to adjusting the scale end in the Controls window.

Use the scale tool to compensate for cases where one part of an image is so bright that the rest of the image is overly suppressed (For example, a bladder in a bone scan). To brighten the suppressed areas, click and drag in the brighter portions of these areas.

The scale tool also provides a shortcut to restore standard data mapping. Double-click in the scale tool icon to do the following to the active study window: 1) Restore the scale start-point to zero. 2) Restore the scale value to one. 3) Restore the scale end to the default value (255 for byte-depth studies, maximum pixel value for word-depth studies).

5. Hand tool. When the hand tool is selected, the cursor changes to  when it is over an active study window. If the image frames have been zoomed such that you cannot see the entire image, then when you click and drag with the hand tool you expose the invisible portions. Think of it like this: You are looking at a ten inch by ten inch picture through a two inch by two inch square hole cut in the center of an enormous piece of paper resting on the picture. The small viewing hole represents the visible image in the study window. You are allowed to slide the paper around on the picture to see other areas of the picture. This is equivalent to using the hand tool.

Occasionally you may encounter images too large to fit your monitor, even when there is a direct mapping between image pixels and screen pixels. Double-click on the hand tool icon to resize and clip the active study window to fit your monitor. Then click and drag in the study window with the hand tool to view the entire image area. To restore the full viewing area, click in the zoom box of the study window, then double-click on the magnifying tool icon in the Tools window.

6. Arrow tool. When the arrow tool is selected, the cursor changes to  when it is over an active study window. Click in the active image frame, then drag and release to place an arrow overlay. The start of the arrow is the point where you first click. The end of the arrow is at the mouse-release point.

7. Rectangular ROI tool. When the rectangular ROI tool is selected, the cursor changes to  when it is over an active study window. Click, drag and release to create a new, rectangular ROI. The dragging motion must be from any corner of the ROI to the diagonally opposite corner. Hold down the shift key while dragging to constrain the ROI to square proportions. Select **Undo Region** from the **Edit** menu to undo placement of a new region.

If there is a single selected ROI in an active image frame, then you can edit the ROI with the rectangular ROI tool as follows: 1) Union. Hold down the shift key while clicking. The cursor changes to  . The rectangular ROI you draw is added to the existing ROI, to make one larger ROI. 2) Difference. Hold down the shift and control keys while clicking. The cursor changes to  . The rectangular ROI you draw is subtracted from the existing ROI, to make one smaller ROI. Select **Undo Region Modify** from the **Edit** menu to undo editing of an existing region.

To place a rectangular ROI that encloses the entire active frame of the active study window, double-click in the rectangular ROI icon of the Tools window.

8. Oval ROI tool. When the oval ROI tool is selected, the cursor changes to  when it is over an active study window. Click, drag and release to create a new, oval ROI. The dragging motion must be from any corner of the ROI's bounding rectangle to the diagonally opposite corner. Hold down the shift key while dragging to constrain the ROI to circular proportions. Select **Undo Region** from the **Edit** menu to undo placement of a new region.

If there is a single selected ROI in an active image frame, then you can edit the ROI with the oval ROI tool as follows: 1) Union. Hold down the shift key while clicking. The cursor changes to  . The oval ROI you draw is added to the existing ROI, to make one larger ROI. 2) Difference. Hold down the shift and control keys while clicking. The cursor changes to  . The oval ROI you draw is subtracted from the existing ROI, to make one smaller ROI. Select **Undo Region Modify** from the **Edit** menu to undo editing of an existing region.

To place the largest possible oval ROI in the active frame of the active study window, double-click in the oval ROI icon of the Tools window.

9. Irregular ROI tool. When the irregular ROI tool is selected, the cursor changes to  when it is over an active study window. Click, drag and release to create a new, irregular ROI. Draw any shape you require. When you release the mouse, the ROI is completed with a straight line from the release location to the initial click location. Select **Undo Region** from the **Edit** menu to undo placement of a new region.

If there is a single selected ROI in an active image frame, then you can edit the ROI with the irregular ROI tool as follows: 1) Union. Hold down the shift key while clicking. The cursor changes to  . The irregular ROI you draw is added to the existing ROI, to make one larger ROI. 2) Difference. Hold down the shift and control keys while clicking. The cursor changes to  . The irregular ROI you draw is subtracted from the existing ROI, to make one smaller ROI. Select **Undo Region Modify** from the **Edit** menu to undo editing of an existing region.

10. Magnifying tool. When the magnifying tool is selected, the cursor changes to  when it is over an active study window. Click and release on an area of interest in the

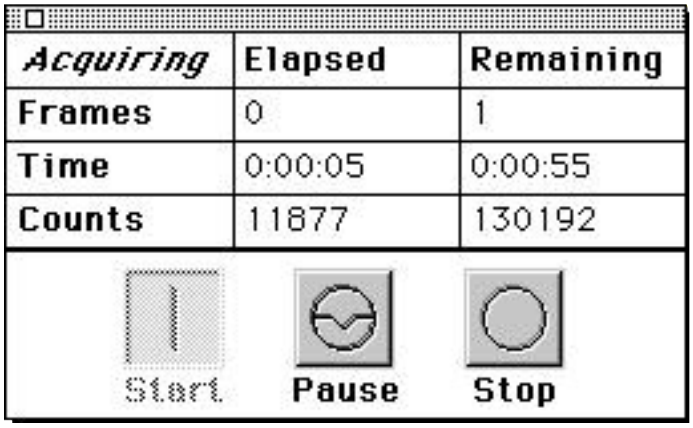
image. The area is doubled in size and centered in the image frame. This action affects the display of all frames in a study. If the computer beeps, then you have reached the limit of allowed zooming. Use the hand tool to view hidden parts of a zoomed image.

Hold down the option key to zoom back out. The cursor changes to . Click and release in the image to halve the size of the image pixels. As a shortcut to restoring the active study window to a non-zoomed state, double-click in the magnifying glass icon in the Tools window.

11. Text tool. When the text tool is selected, the cursor changes to when it is over an active study window. Click, release, type text, then click elsewhere to place a text overlay in an image frame. To change text characteristics for subsequent overlays, deselect all objects in the active study window, then make the selections you want from these submenus of the **Edit** menu: **Font**, **Size**, **Style**.

3.2.2.4 Progress Window

The Progress window displays information about and controls data acquisition. It is described further in the chapter on acquiring studies.



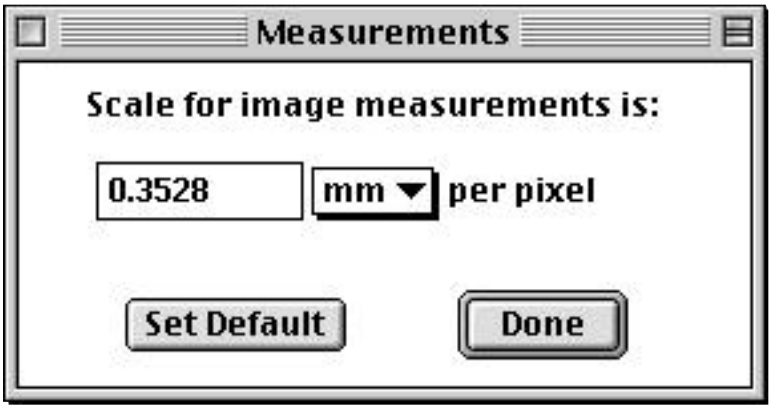
NucLear MAC Progress Window

3.2.3 Dialog Windows

Dialog windows are used for exchanging information with and controlling the NucLear MAC application. There are two types of dialogs: modal and mode-less. Modal dialogs require your attention before you can proceed with other actions in NucLear MAC. Modal dialogs lack close-boxes, and most menu items are disabled when a modal dialog is active. One common use for modal dialogs is to display error messages, since it is important for you to read the message before continuing.

A modal dialog displaying an out-of-memory error

Most NucLear MAC dialog windows are mode-less. You use these windows for things like entering parameters and initiating data acquisition. Mode-less dialogs have close-boxes, and most menu items remain enabled when these dialogs are active. Most information you enter in a mode-less dialog is used immediately. That is, if you are changing parameters that control NucLear MAC's operation, the changes take effect as soon as you type them—even before you close the dialog window. These changes remain in effect until you quit the NucLear MAC application. The next time you run NucLear MAC, all parameters are restored to their default values, which are saved in the "NucLear MAC Prefs" file.



A mode-less dialog used for parameter entry

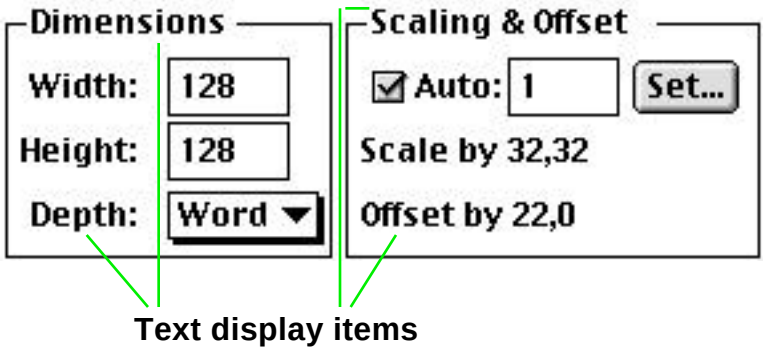
Dialog windows contain a number of standard items with which you should be familiar.

3.2.3.1 Dialog items

Internally, in any particular NucLear MAC dialog, the dialog items are arranged in a hierarchy. That is, each item is owned by another item, and each item can own other items. In general, you do not need to be aware of the hierarchy, but the behaviors of some dialog items are influenced by the items they own.

Text display items

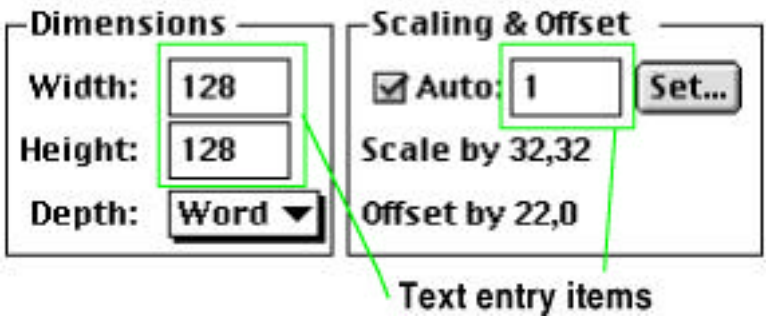
Text display items are most commonly used to label other items, although they are also used to provide information. A text display item is simply text drawn in the system font.



Text entry items

Text entry items accept typed information directly from the user. Edit text in these fields as you would any text under MacOS; click the mouse in the desired field and type. Use the standard **Edit** commands to operate on the current selection: **Cut**, **Copy**, **Paste**, **Clear**, **Undo** and **Select All**. Use the tab key to advance the selection to the next text entry item. Type shift-tab to move the selection to the previous text entry item.

There are two main types of text entry fields: numeric and alphanumeric. Both types of fields limit the number of characters you can enter. Numeric fields accept only numeric characters. For example, many fields accept only positive integers, meaning that only the digits '0' through '9' can be entered. Numeric fields can also have maximum and minimum values. Alphanumeric fields accept both alphabetic and numeric characters. These fields are used primarily for names and comments. Some alphanumeric fields may not accept either the colon character ':' or the return character. In cases where the return key normally performs an action (see *Default action* below), you need to hold down the option key while pressing the return key to insert a return character in a text field.



Buttons

Buttons are simple interface elements that respond to mouse clicking. Buttons are used for setting information or initiating actions. Buttons may be identified by either text or pictures.

Standard Buttons

Standard buttons always initiate an action in response to a mouse click. The following types of standard buttons are in several NucLear MAC dialog windows.

Cancel

The Cancel button appears only in modal dialogs. In general, choosing Cancel aborts any actions proposed by the dialog. The following are keyboard equivalents for clicking the Cancel button: 1) Press the esc key. 2) Press period '.' while holding down the command (⌘) key.

A cancel button

Default action

Most NucLear MAC dialog windows have a single button that responds to the return or enter keys on the keyboard, as well as mouse clicks. The extra bands drawn around their exteriors identify these default action buttons. There are a number of instances where the default action button dismisses the dialog after performing the specified action. To override this behavior, hold down the shift key while activating the button. The dialog window then remains open.

A default action button

Done

A Done button appears in mode-less dialogs that accept information but perform no other functions. The Done button is a default action button, so besides mouse clicks, it responds to the return and enter keys. The action of the Done button includes closing the dialog, as if you clicked in its close box or selected the **Close** command from the **File** menu. In addition, the Done buttons alerts you to any errors present in the dialog's fields.

A Done button

Set Default

A Set Default button is used to save permanently the values of fields of a dialog to the "NucLear MAC Prefs" file. The saved values are in effect whenever the NucLear MAC application is reactivated. You can override default values at any time simply by changing them, but when you quit and restart the program, the values revert to what they were when you last clicked on the Set Default button.

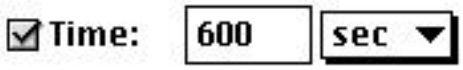
A Set Default button

Check boxes

Check boxes are a type of button with two states: on or off. Typically you use check boxes to enable or disable certain program features. You turn a check box on or off by clicking in it; the mouse click switches its state.

Check boxes, one turned off and one turned on

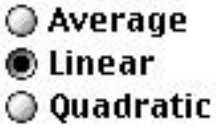
In certain situations, a check box can own other dialog items. This means that when the program option provided by this check box is available (the main option), then other sub-options also apply. When the main option is disabled, the sub-options are ignored. **Important note**: When you change a sub-option of a check box, the check box is automatically enabled.



A check box that owns other fields

Radio buttons

Radio buttons are similar to check boxes, except that they allow a selection from two or more states. Radio buttons appear in clusters of two or more; each radio button represents one option from a set of mutually exclusive states. Although any particular radio button can be on or off, in a cluster of radio buttons, only one radio button is on—the remaining buttons are off. When you click on a radio button that is off, it is turned on and all other radio buttons in the cluster are turned off.



A cluster of radio buttons

As with check boxes, radio buttons can own other dialog items. When you change any sub-fields of a radio button, the radio button is automatically enabled, and other radio buttons in the same cluster are disabled.

Popup menus

Popup menus are menus that appear when you click on items in a dialog window. They behave the same way standard menus do with regard to selecting individual items.

Fixed popup menus

A fixed popup menu is rather like a space-saving radio button cluster. It lets you select one item from a fixed list. That item is active until you select a different item. The active item is drawn in the popup menu box. In the “Depth” example below, you can choose between “Word” or “Byte” when you click in the popup menu. The “Word” item is selected.

A fixed popup menu

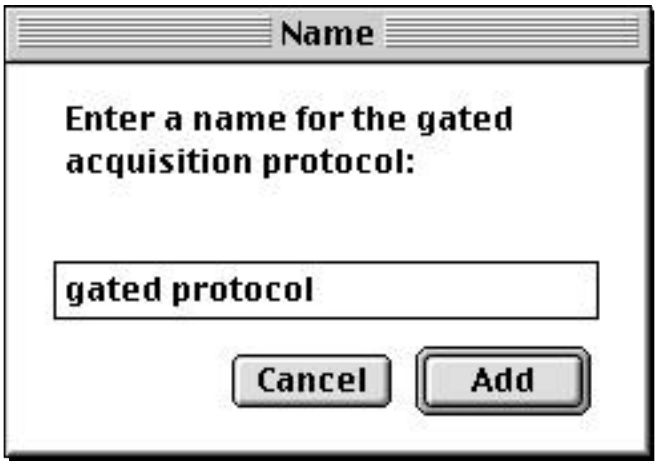
Protocol-defining popup menus

Popup menus are also used to define, activate and delete NucLear MAC protocols. A protocol is a generic name for a group of program parameters. Protocols are defined in cases where you may frequently need to change a group of settings. For example, for image acquisition, you set such fields as image dimensions, depth, stop conditions, etc. Rather than re-enter different values each time you perform a different type of acquisition, it is simpler to define a readily accessible protocol for each type of data acquisition.

A protocol-defining popup menu

Typically a protocol-defining popup menu is located near the top of a dialog window. A protocol-defining popup menu owns most or all of the fields in its dialog window. When any of its owned fields are modified, the popup menu display reverts to “Untitled.” This is because the parameter values no longer correspond to a previously defined protocol.

To define and add a new protocol, enter the desired parameters in the fields of the dialog window. The popup menu box should contain “Untitled”. Select the **Add...** item from the bottom of the popup menu. Enter an appropriate protocol name in the resulting modal dialog and click the Add button. For example:



Dialog window prompting for protocol name

The popup menu box changes to reflect the current protocol name, and the new protocol is added as an item in the popup menu.

Popup menu showing name of current protocol

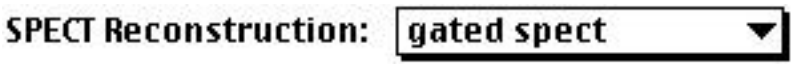
To activate an existing protocol, select the name of the protocol from the popup menu. The fields of the dialog change to reflect the values stored in the protocol. The popup menu displays the protocol name.

To delete an existing protocol, activate the protocol by selecting its name from the popup menu. Then select **Delete** from the bottom of the popup menu. The popup menu display changes to "Untitled".

To modify an existing protocol, activate the protocol by selecting its name from the popup menu. Make the desired changes to the dialog fields. The popup menu changes to "Untitled". Next hold down the option key while selecting the name of the protocol from the popup menu. This latter is a shortcut for selecting **Add...** from the bottom of the popup menu, and entering the original name of the protocol.

Protocol-selecting popup menus

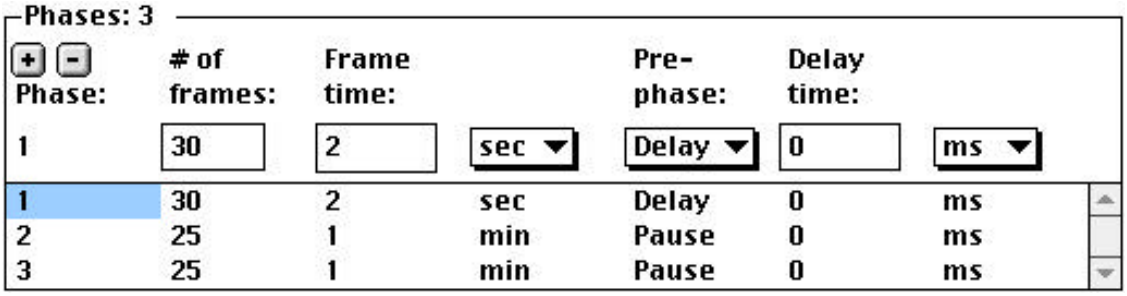
It is not unusual in NucLear MAC for protocols to be used in a number of places apart from where they are defined. For example, image filters are used in several different processing scripts. In these cases where you need only select a particular protocol, there are protocol-selecting popup menus. These popup menus are almost identical to protocol-defining popup menus except that instead of an **Add/Delete** item at the end of the menu, they have an item labeled **Use Current** or **None**. The popup menu box always displays the current selection. When present, the **Use Current** option means that the procedure uses the settings in the protocol-defining dialog. The **None** option is used to skip the particular procedure. For example, if **None** is used for filter selection, then no filtering is performed.



A protocol-selecting popup menu

Lists

List items are used to group a set of dialog fields and allow multiple sets of parameter values for these fields. For example, in data acquisition for a dynamic study, each list line represents one phase. Each phase has the same set of fields: number of frames, time per frame, etc. But each phase may have different values in these fields. Each list line occupies a separate line in the list box. Values of one or more owned fields are displayed in the line. A scroll bar in the right of the list box becomes active when there are too many list lines to see at one time.



A dialog list item with six owned fields

To add a line to a list, click on the "+" button. A new line is appended to the list. The new line is selected, so you can immediately change the owned fields for this list line. If a list line is selected prior to creating a new line, then the new line is a duplicate of the selected line until you modify it. If the "+" button is disabled, then the list already contains its maximum number of lines.

To select a list line, click anywhere in the line. If you click in a text field value, then the corresponding text field is also selected. A selected list line always has one of its value fields highlighted.

To edit values in a list line, select the line, then modify the owned dialog fields.

To delete a list line, select the line, then click on the "-" button. If the "-" button is disabled, then the list cannot have fewer lines.

Some dialog list items also have a button labeled "All". Clicking the "All" button sets values of all list lines to those of the selected list line, except that the first value of each line is unchanged.

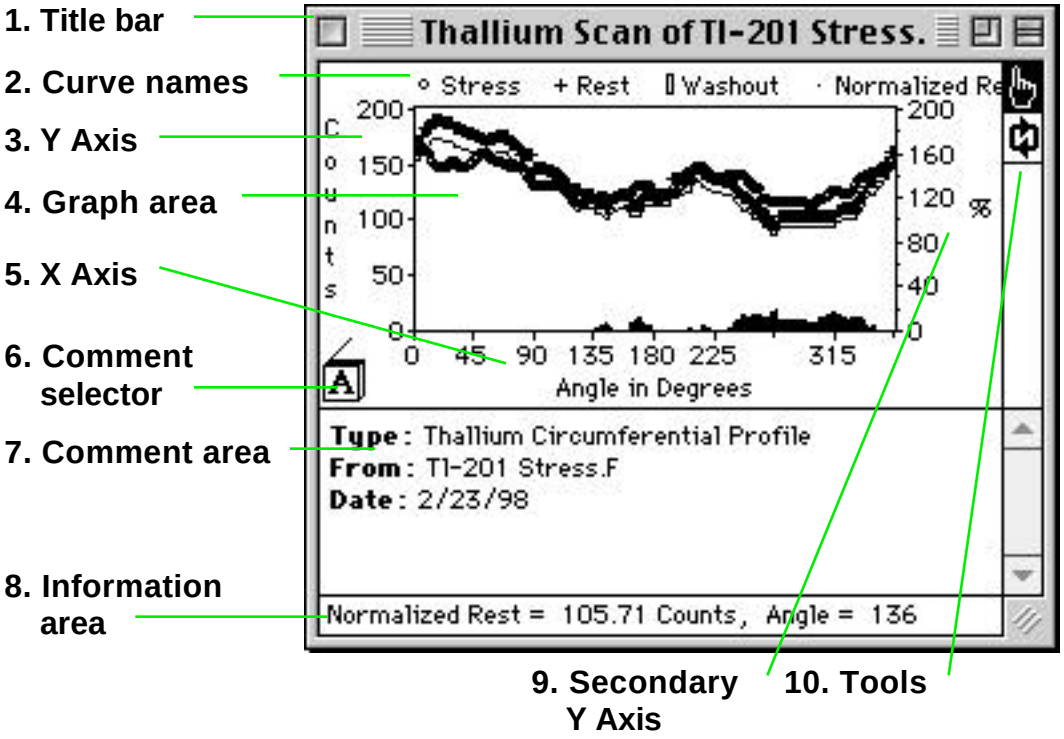
3.2.3.2 Subdialogs

Subdialogs are dialog windows owned by other NucLear MAC windows. Subdialogs behave the same as regular dialogs, except that when the owning window of a subdialog is closed, the subdialog is also closed. Subdialogs are used to 1) Provide

information about the owning window; 2) Modify the owning window's information; 3) Set parameters for and initiate an action using the owning window.

3.2.4 Graph Windows

Graph windows display curves that result from various NucLear MAC processing functions. These include time-activity curves, thallium profile plots, ROI count distributions, etc.

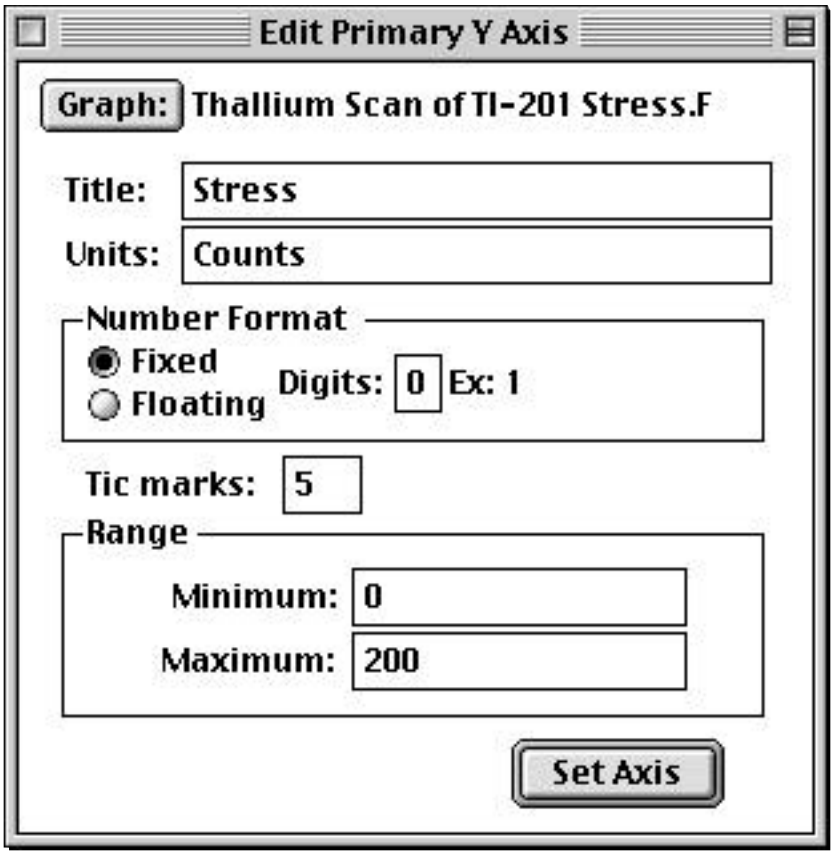


A graph window

1. Title bar. Use the title bar to move, close or zoom the graph window.

2. Curve names. This area of the graph window lists all curves present and serves as a legend of the symbols used to draw the curves.

3. Y Axis. The Y-Axis area displays the units used for the axis, along with the scale and numerical values. If there is enough room, the axis title is also displayed. Click in this area to edit the names and scale.



Subdialog to edit Y Axis of a graph

To shift the whole axis scale up or down, hold down the option key while clicking and dragging up or down in the Y-Axis area.

4. Graph area. The curves are drawn in this area. Click on specific curves or points to apply the active tool to these objects.

5. X Axis. The X-Axis area displays the title and units used for the axis, along with the scale and numerical values. Click in this area to edit the names and scale. The resulting subdialog is the same as for the Y Axis.

6. Comment selector. Click on the comment selector to show or hide the comment area of the graph.

7. Comment area. Use the comment area to enter text describing the graph. Some NucLear MAC processing functions insert comments in the graphs they create.

8. Information area. The information area displays directions for using the selected graph tool. Some graph tools display their results in this area. If the comments area is visible, then clicking in the information area copies the information to the comments area.

9. Secondary Y Axis. The secondary Y Axis provides a second scale for curves that may have different units than those in the primary Y Axis. Click in this area to edit the

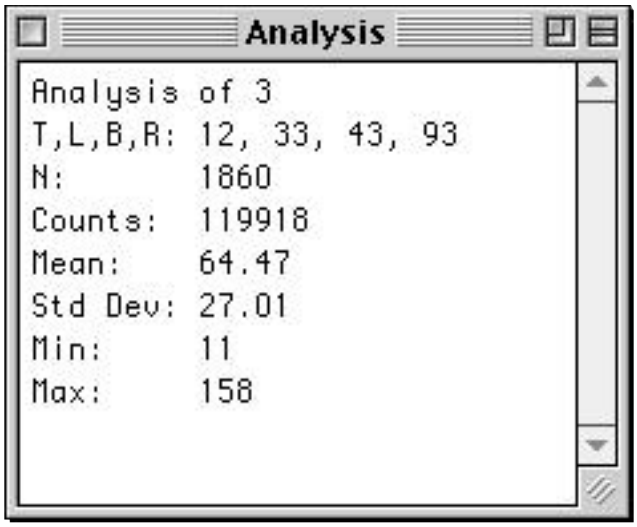
names and scale. To shift the whole axis scale up or down, hold down the option key while clicking and dragging up or down in the secondary Y-Axis area.

10. Tools. The number and type of graph tools varies depending on the type of graph. To use a tool, select it by clicking on its icon. When the cursor is over the graph area, it changes to match the selected tool. Click on the point or curve for the tool to use.

One tool common to all graphs is the hand tool , which serves as an identifier. Click on a curve point with the hand tool to display the point's X and Y values in the information area.

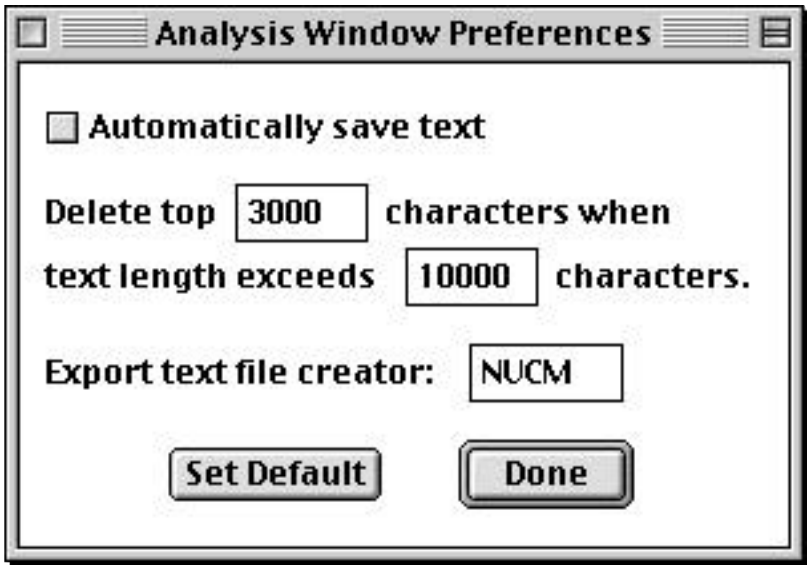
3.2.5 Analysis Window

The NucLear MAC Analysis window displays the results of various processing functions. You can also use the Analysis window as a scratch pad. To select the Analysis window, choose **Analysis** from the **Windows** menu.



NucLear MAC Analysis Window

The information in the Analysis window can be saved in the "NucLear MAC Prefs" file. Saved information is displayed even after quitting and restarting the NucLear MAC application. To manually save Analysis window information, select the Analysis window and choose **Save** from the **File** menu. Use the **Font**, **Size** and **Style** submenus of the **Edit** menu to change the appearance of text in the Analysis window. New results displayed by NucLear MAC are always inserted at the end of any existing text. The appearance of the new text matches the appearance of the last character of the existing text. Select **Analysis Window ...** from the **Edit>Preferences** submenu to edit preferences for the Analysis window.



Analysis Window Preferences Dialog

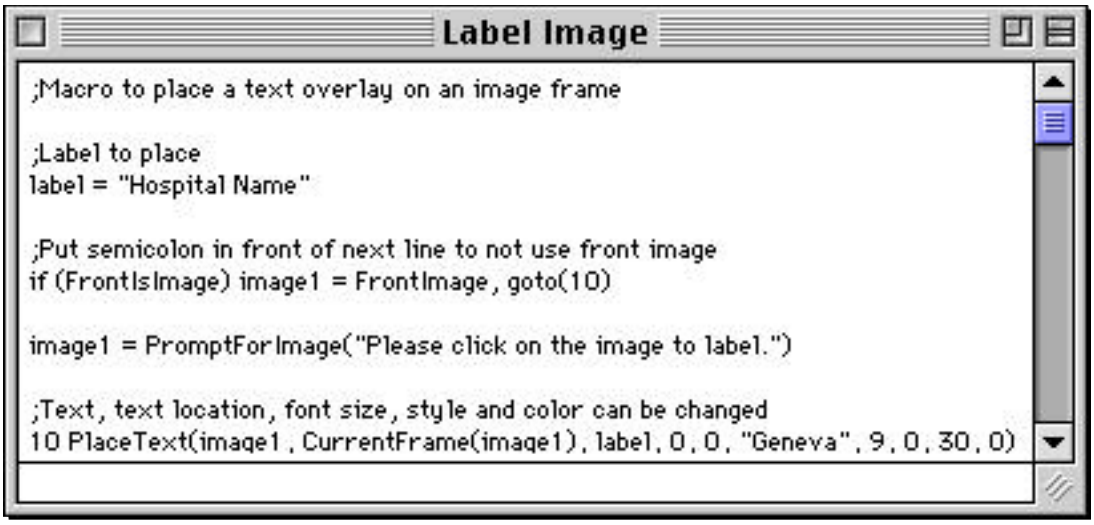
When 'Automatically save text' is enabled, the contents of the Analysis window are automatically saved when you quit the NucLear MAC application.

The Analysis window is limited to holding 32,000 characters, and may become sluggish even with a lot fewer characters. You can control the maximum number of characters in the Analysis window by having a number of characters deleted automatically when a specified threshold is reached.

You can export the contents of the Analysis window to a standard Macintosh text file. Use the **File>Export** menu item and choose 'Text" for the format. The 'Export text file creator' field of the preference dialog refers to the creator type given to the resulting text file. This determines which application opens the text file by default.

3.2.6 Macro Windows

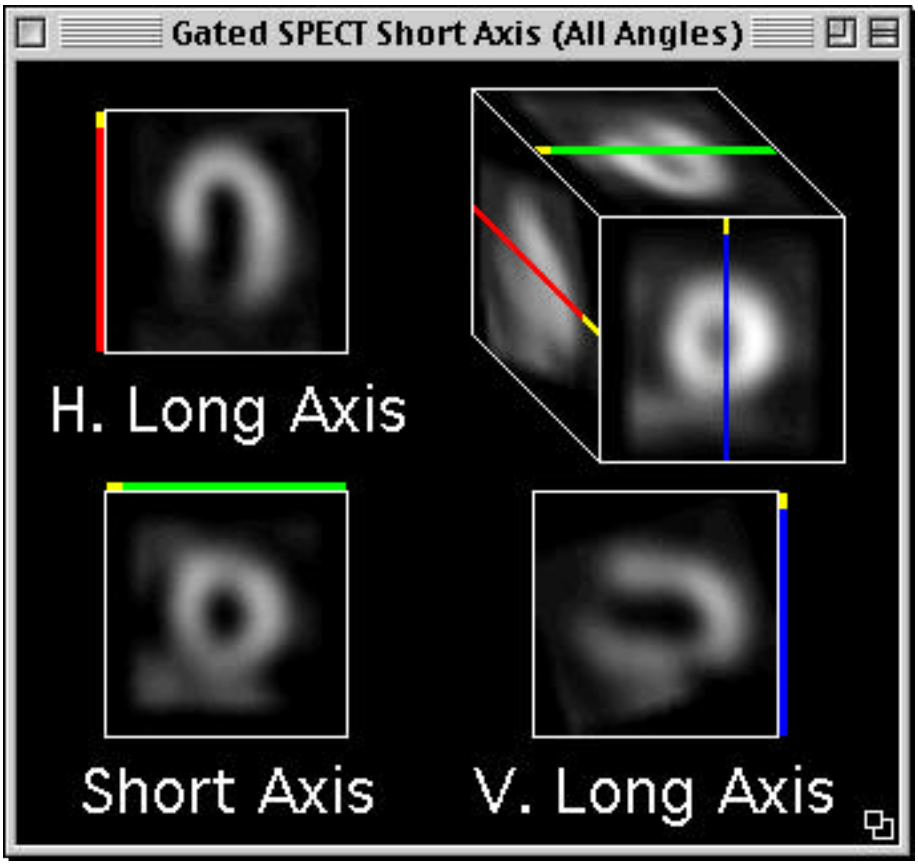
Macro windows are similar in appearance and behavior to the Analysis window. Macro windows are used to edit macros, which are text instructions for NucLear MAC.



A macro window

3.2.7 All-Angles Windows

All-angles windows provide an interactive display of reconstructed SPECT or gated-SPECT studies. To make a new all-angles window from a short axis or coronal SPECT reconstruction, select the study window, then choose **All Angles Window** from the **Display** menu. The terminology in the all-angles window depends on the source study. If the source study uses heart terminology, then the three views are referred to as Horizontal Long Axis, Short Axis and Vertical Long Axis. Otherwise the views are referred to as Transverse, Coronal and Sagittal.



An all-angles window

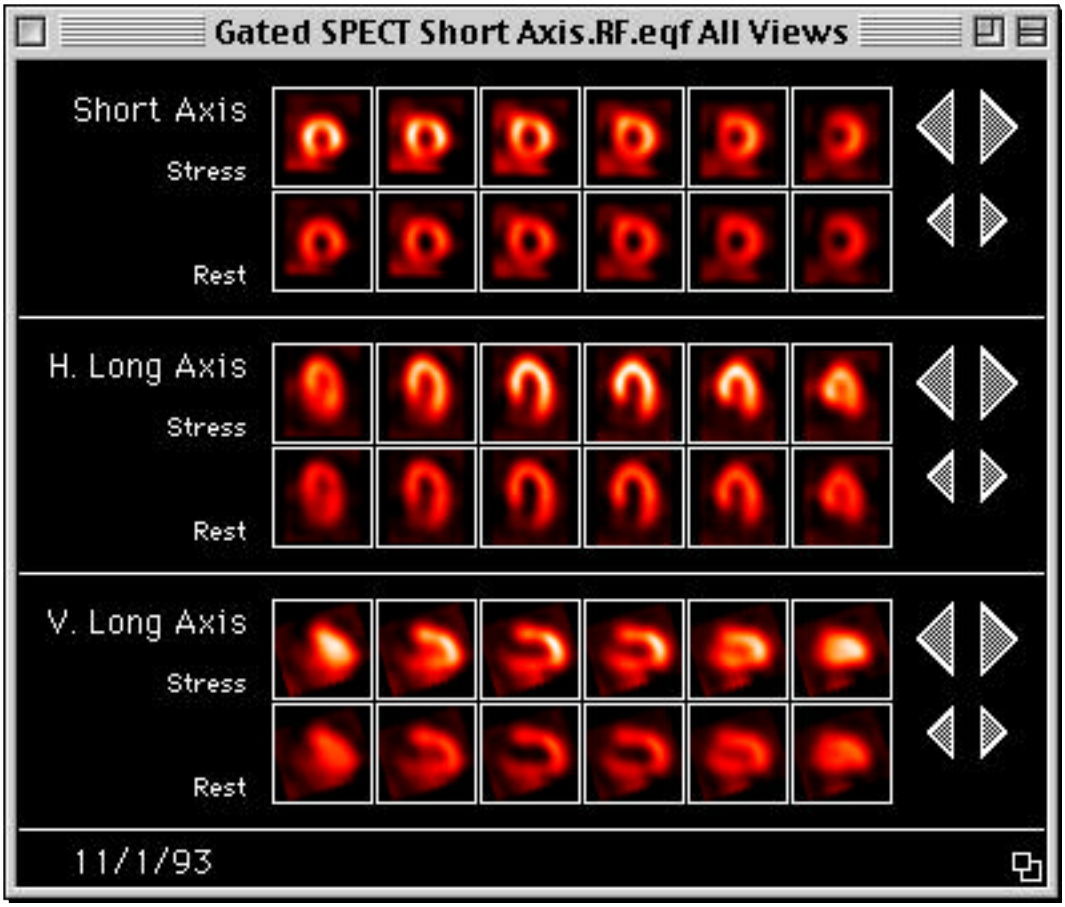
An all-angles window displays one slice from each of the three views, as well as a projection cube. The colored bars next to the slices correspond to the colored bars on the projection cube. Drag a bar on the projection cube to display any slice from that view. Click in any of the three views to select slices in the other two views that match the cursor location.

If the all-angles window is created from gated-SPECT data, then it responds to the standard cinegraphic menu commands: **Run**, **Halt**, **Step Forward**, **Step Backward**, etc. The speed of an all-angles window cinegram is taken from the source study when the all-angles window is created. To adjust the speed after the all-angles window is made, activate the all-angles window and use the speed control in the Controls window.

To adjust the size of an all-angles window, click and drag in the grow region in the bottom, right of the window. The modifier keys work the same as for study windows: Hold down the shift key to constrain the window size such that the slices are integer multiples of their actual sizes. Hold down the option key to remove the constraint on aspect ratio during resizing.

3.2.8 All-Views Windows

All-views windows are used to display reconstructed SPECT or gated-SPECT studies. You can compare up to four different studies in one all-views window. You can mix gated and non-gated studies in the same window. Typically you might choose to display stress and rest studies for the same patient. To create an all-views window, activate the first study window, then choose **All Views Window...** from the **Display** menu. You are prompted to click on additional studies, if any, for the all-views window. You can end the selection process by clicking anywhere besides a study window, or by clicking on a study window that has already been chosen.



All-views window

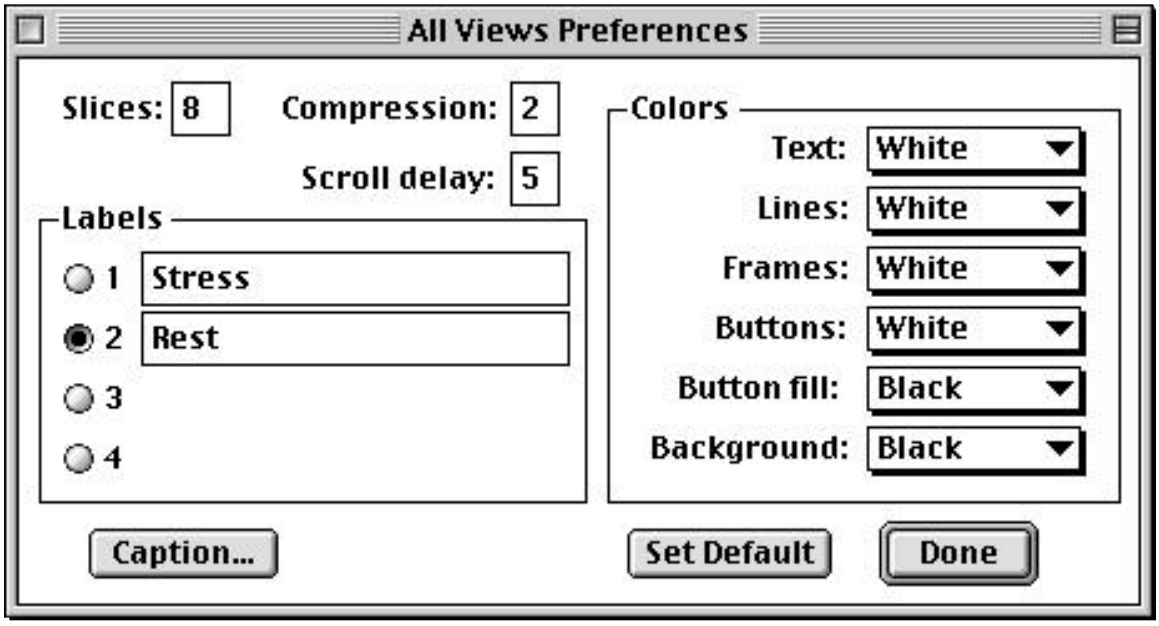
An all-views window consists of three main areas, one for each view. Within each area slices from different studies are displayed on separate rows. The first study in each view area has two large triangles to its right. Use these triangles to scroll all slices in the view area to the left or right, revealing slices not currently visible. If there are multiple studies in the all-views window, then the subsequent rows in each view area have small triangles to the right. These triangles are used for scrolling the particular row. With these triangles you can line up the slices of the different studies.

By default, each study beyond the first is normalized so that its highest-valued pixel has the same brightness as the highest-valued pixel of the first study. You can disable

normalization by clicking in any slice. This displays each study according to its relative pixel intensity. To re-enable normalization, click again in any slice.

If the all-views window is created from gated-SPECT data, then it responds to the standard cinegraphic menu commands: **Run**, **Halt**, **Step Forward**, **Step Backward**, etc. The speed of an all-views window cinegram is taken from the first source study when the all-views window is created. To adjust the speed after the all-views window is made, activate the all-views window and use the speed control in the Controls window.

To adjust the size of an all-views window, click and drag in the grow region in the bottom, right of the window. The modifier keys work the same as for study windows: Hold down the shift key to constrain the window size such that the slices are integer multiples of their actual sizes. Hold down the option key to remove the constraint on aspect ratio during resizing.



All-views window preferences

You can customize the appearance of any all-views windows you create. Choose **All Views...** from the **Edit>Preferences** submenu. This activates the preferences dialog. Changes made to these fields are applied to subsequent all-views windows you create; existing all-views windows are not affected.

The Slices field sets how many slices are drawn per row. The compression field sets the amount of reframing applied to the source studies; a value of one means the slices are displayed exactly as they appear in the source studies; a value of two means that every two slices are added together to be one slice in the all-views window. The Scroll delay field sets the number of tics (60ths of a second) to pause between scrolling rows in the all-views windows. The Labels fields let you set the study labels for the all-views windows. You can specify different labels for different numbers of studies. Use the radio buttons labeled 1, 2, 3, or 4. The Colors popup menus let you select the colors of

the various graphical elements in all-views windows. The Caption… button brings up another dialog in which you can define the text that is drawn along the bottom of the all-views window.

4 Acquiring Studies

4.1 Background

4.1.1 Data Acquisition

Gamma cameras generally share a common mode of analog data output: X, Y, and Z channels. The Z channel is used to indicate when the camera detects a gamma ray. Voltages on the X and Y channels reflect the position of the gamma ray interaction on the camera face. Typically gamma cameras use negative X and Y voltages to indicate the lower, left direction of the image and positive X and Y voltages to indicate the upper, right direction; zero voltages on X and Y represent the center.

For computer acquisition of the gamma camera data, therefore, hardware that can rapidly, simultaneously, and accurately digitize the X and Y voltages when triggered by the Z channel is needed.

To accomplish this task, NucLear MAC uses a nuBus or PCI bus interface board containing circuits to digitize gamma camera signals when triggered by the Z channel. Sample and hold amplifiers acquire the signals from the X and Y lines and a 12-bit analog-to-digital converter (ADC) operating at one or five megahertz then digitizes the sampled signals.

In the gamma camera interface the ADC is used to divide the range of the analog output of gamma cameras into discrete digital bins. The output voltages vary between cameras, but are usually between ±1 and ± 2 volts. In other words, the total voltage range generated by the camera runs from approximately two volts (±1 volt) to four volts (±2 volts).

Consider the example of ± 2 volts for the X and Y signals (a elapsed s or 4000 millivolt total range). NucLear MAC initially digitizes the signal in 12 bits or 4096 values. Therefore neighboring digital values are separated by approximately one millivolt. In other words, each digital bin is nominally one millivolt wide. The X and Y analog voltages are rounded into the nearest bin to yield digital values of the X (horizontal) and Y (vertical) position of an event. Therefore, it at first would appear that the accuracy of the position localization would be to within one millivolt.

However, because of the nature of ADCs, the widths of these bins differ slightly in size between one another. The parameter that describes this non-uniformity is differential non-linearity (DNL). If a small bin is next to a large bin, then voltages occurring between the mid-points of the bins round into the larger bin more often than into the smaller bin. For a uniform distribution of voltages, therefore, the large bin contains more counts than the small bin. Because the values represent columns or rows, DNL causes bright and dim horizontal or vertical stripes.

Randomization is used to reduce the effects of DNL. Small, randomly varying, voltages are added to the ADC input (the gamma camera output) voltage so that a given camera output voltage falls into different bins at different times. After digitization, the digital value of the added voltage is subtracted from the calculated output value.

Thus, a given voltage does not consistently fall into a large or small bin and it is more uniformly represented.

This approach trades off an improvement in uniformity for a small drop in precision. The drop in precision occurs because it is impossible to exactly correct for the randomly added voltage. Specifically, the final bin widths are increased from their nominal values of one millivolt to approximately three millivolts through the randomization process. In other words, the final number of discrete digital values is approximately 1300. The specification of maximum X and Y digitization of 1024 pixels used by NucLear MAC is derived from this value.

The degree of randomization is automatically customized for each acquisition board. Other randomization parameters can be adjusted in the Hardware Settings dialog obtained by choosing **Hardware...** from the **Acquire** menu. In general, there should be no need to change the values from their defaults.

4.1.2 Image Generation from Acquired Data

After the X and Y signals of the gamma cameras are digitized, their position values are used to generate the images of the patients. This process is performed as follows. The X and Y values are first stored in a buffer (512 counts) on the interface board. The X and Y values are then read from the interface board and placed in computer memory. After a certain number of these pairs of values are stored, they are read from memory and used to update the image. Each pair represents a location on the image, and when a particular pair is read, the brightness of the pixel at the corresponding position is increased by one. This process is performed until a desired number of counts or total acquisition time is reached.

Ideally, it would be desirable to display the forming image as it is generated point-by-point. This is impractical with modern operating systems, as direct access to the screen display buffers is discouraged, and point-by-point drawing is too slow. However, by storing 1000 to 10,000 points in memory between image updates and using efficient code for data processing, camera count rates greater than 100,000 cps can be captured with nearly 100% efficiency. Because this amount of data buffering represents only a second or two, the images appear to form in nearly real time.

4.1.3 System Performance

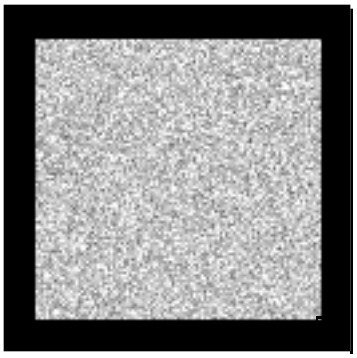
NucLear MAC can acquire data at high count rates. Actual maximum rates are determined by the type of Macintosh computer, whether a PCI or nuBus interface board is used, the type of acquisition, and the type of blanking of the gamma camera Z signal. Generally, when a Macintosh IIci or faster computer is used, NucLear MAC can digitize an event in less than 5 microseconds in frame-mode acquisition. Therefore a maximum count rate of 200,000 cps is supported for single frame acquisition (in non-simultaneous acquisition / processing mode). For very high count rates with cameras that have inter-Z pulse times of less than five microseconds, the data may not be captured with 100% efficiency; those events that follow closer than five microseconds to each previous event may be missed. However, this efficiency limitation does not affect most clinical situations because count rates of 125,000 cps can be achieved with approximately 95% efficiency.

For gated acquisitions in which real-time forward/backward framing of the data is performed, the maximum count rate that can be acquired is approximately 40,000 cps in non-simultaneous mode and greater than 100,000 cps in simultaneous mode.

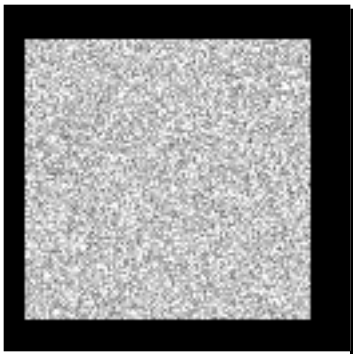
4.2 Center of Rotation

4.2.1 Basis

The gamma camera images are stored in the computer as matrices of pixels. Ideally, if the camera is functioning perfectly, then the signals that it sends to the computer will be perfectly centered; in other words, the center of the computer image is equal to the physical center of the gamma camera's head. Please see example of centered image below.



Unfortunately, small amounts of off-centering are common in most rotating gamma camera-type SPECT systems. The off-centering can be caused by mechanical misalignment of the gantry or electronic miscentering of the camera output signals. Please see example of off-centered image below.



Such off-centering, if not corrected, will cause artifacts in SPECT reconstructions. With most computer systems, correction is done by calculating the offset error, and then using this calculation during the SPECT reconstruction to modify the images. NucLear MAC can take this approach. Normally, the value in the "COR offset" part of the SPECT reconstruction dialog box defaults to the center of the image, but the user can enter a different value to include any offset error.

However, NucLear MAC can also adjust the center point of the original SPECT images so any camera errors are corrected during the acquisition. In other words, the

off-centered image can be centered by the computer as it is acquired. This process simplifies SPECT reconstruction since COR errors are eliminated in the original images and can be ignored during reconstruction.

To measure center of rotation error a point source is placed near the center of the camera field of view. Then images are acquired 180 degrees apart from one another, e.g. from the top and bottom. Consider a point source sitting on the SPECT table that is two pixels to the left of the center of the field of view of the camera when viewed from the top. If there is no COR error, then the source will appear to be two pixels to the right when viewed from the bottom. If we add the top and bottom views together, we will have two points in the image. If we determine the midpoint between the two, it will be in the exact center of the image. Now consider the situation where there is a one pixel rightward COR error and the point source is still sitting on the table two pixels from the true center. The top image will show a point located one pixel to the left of center and the bottom image will show a point located three pixels to the right. When the images are added together, the midpoint will be one pixel to the right of center, rather than at the exact center if there were no COR error. Thus we have determined that there is a one pixel rightward COR error. Because COR error can change somewhat as the camera moves around the patient, e.g. from slight bending of the camera support arms when the heavy camera head is to the side of the table, COR error is usually measured from multiple positions, e.g. top-bottom, left-right, etc. Then the average error is calculated from the different pairs of opposing images.

4.2.2 Processing

Acquire a calibration study, which can be either of the following: 1) A 360° SPECT study with an even number of views. 2) One or more pairs of images. Each pair is acquired such that the camera is rotated 180° between the two images. The pairs need not be acquired in order, so, for example, you could acquire an image sequence with the camera at these successive angles: 0°, 90°, 180°, 270°. A vertically oriented line source is recommended for either type of acquisition, but a point source near the center of the field of view is adequate.

Select **Auto-Scale Script...** from the **Analyze>COR** submenu and follow the instructions. This computes a COR value that is used for all acquisitions in which the "Auto" scaling check box is enabled. Be sure to click on the "Set Default" button if you want to permanently retain the new COR value.

The **Graph X & Y** selection from the **Analyze>COR** submenu can be used to check that the camera center of rotation has been properly set. Acquire a 360° SPECT series (30 to 64 frames - 20 to 60 seconds each) of a slightly off-centered point source (approximately 0.5 mCi) or line source (2 to 5 mCi). If a line source is used, it should be oriented along the axis of rotation. After the acquisition is complete, select the COR analysis and follow the script. The routine will generate plots of the Y axis center of rotation and X axis center of rotation. The X axis plot is used to verify proper camera centering. If the camera is perfectly centered, then the line or point values will have an offset of 0.0 pixels (the values are read from the plot). Absolute values for the point or the middle of the line less than 0.5 pixels indicate acceptable centering.

4.3 Quality Assurance

4.3.1 INL and DNL

4.3.1.1 Basis

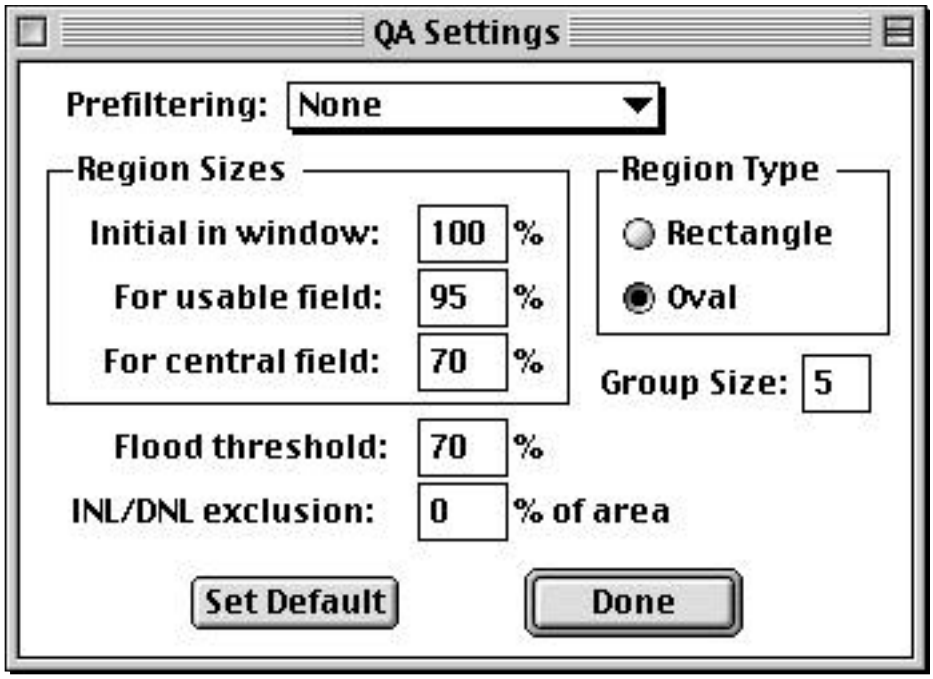
Some method of standard measurement of gamma camera imaging parameters is of obvious utility in quality assurance. NEMA standards for image uniformity have been developed for this purpose. The standards contain two parameters, differential (DNL) and integral (INL) non-uniformity, for the total field of view and useful field of view. Total field of view on a circular gamma camera is defined as follows. Radii are extended outward from the center of an image obtained at 64 x 64 pixels. The camera edge is defined as where the counts fall to half-maximum and full field of view is a radius of 95% of that of the edge. Useful field of view is a radius 70% of that of the edge. While the specifications are old and do not apply to rectangular cameras, our program allows this option. Therefore, our program is a modified NEMA approach.

An image (usually 64 x 64) is acquired with approximately 10,000 counts per pixel (e.g. 30,000,000 for circular field of view camera). It is smoothed with a 9 point filter (most physicists now recommend an unweighted filter to avoid artifacts in INL calculations. Small, overlapping boxes of 5 x 1 pixels are then mapped over the image in both the horizontal and vertical directions. The parameter, (Maximum - Minimum) / (Maximum + Minimum) x 100, is measured within each box and between all pixels. Differential non-uniformity is the maximum parameter value within the boxes and integral non-uniformity is the maximum value between all pixels.

NucLear MAC follows the same general approach as the NEMA measurements buts adds some flexibility. The user can set the image resolution, the user can set the radial fractions for the different fields of view, the system can be used with rectangular or circular cameras, and histograms of the parameter values are generated, as well as the actual DNL and INL measurements.

4.3.1.2 Processing

Use the **Settings...** selection in the **Analyze>Quality Assurance** submenu to set the processing parameters.



The Region Sizes are used to define the sizes of the regions that the computer analyzes in the study window. The "Initial in window" value is used by the computer to define how much of the user-defined region to analyze; if the region is drawn exactly around the flood image (but avoiding the smoothed edges by 2 pixels), then a value of 100% should be used. The "For usable field" defines the fraction of the total region that should be considered to be the usable field of view, e.g. 95%. The "For central field" defines the fraction of the total image that should be considered to be the central field of view, e.g. 70%.

The "Group Size" is the length of the small, one-pixel wide or one-pixel high rectangle that is moved and analyzed in the usable and central fields of view. The NEMA standard calls for this value to be five.

Flood threshold limits the magnitude of the pixel value increase when the Flood Correct function is used. For example, 70% is a common choice and means that the maximum correction that can be applied would correct a pixel whose sensitivity is 70% that of a mean flood value. This feature reduces edge amplification artifacts.

Region Type allows a choice between circular and rectangular analysis regions.

After setting the parameters, acquire a flood image of desired resolution. Choose the **Run Script...** selection of the **Analyze>Quality Assurance** submenu to process the image.

4.3.2 Uniformity Correction

4.3.2.1 Basis

Even with energy, linearity, and uniformity corrections, gamma camera images are not perfectly uniform. If a flood image is acquired of a uniform source there may be areas

that are too bright (excess counts) or too dim (reduced counts). While small non-uniformities are not visible on normal planar images because of statistical noise in the image, they can cause SPECT artifacts. Therefore it is important to correct for these errors before reconstructing SPECT data. NucLear MAC uses the inverse-multiplication method instead of older, less accurate addition methods. The average value of all of the pixels in the flood (A) is first computed. Then each pixel in the patient image (P) is corrected by multiplying it by the ratio of A divided by the flood pixel value at the same location (F). Pnew = P * A / F. Thus areas in the uncorrected image that are too bright are dimmed and areas that are too dim are brightened, yielding a corrected image that is uniform.

4.3.2.2 Processing

Acquire a word mode flood field image of a sheet source (with the collimator on) containing a large number, e.g. 30 million, counts. The matrix size of the image must match the matrix size of any SPECT images that will be corrected. Smooth the flood with the unweighted smooth kernel and save it for later use. Bring up the smoothed flood image. Activate the SPECT study that needs to be corrected. Select **Flood Correct** from the **Analyze** menu and follow the directions to click on the flood window. A new window appears with the flood-corrected SPECT study. Note: the maximum level of correction that can be applied is set with the “Flood Threshold” value in the **Quality Assurance** settings of the **Analyze** menu.

4.4 Other Corrections

4.4.1 Motion Correction

While camera manufacturers have gone to great lengths to stabilize the gantries of SPECT cameras, patient motion remains a major problem. NucLear MAC allows the user to move images within a SPECT acquisition series to correct for motion. Bring the SPECT series window to the front then select **Mouse Track** from the **Display** menu. The arrow keys on the keyboard then shift whichever image is in the front of the window. By moving through all of the images, they can be aligned properly. This is most useful in thallium studies when the patients slide up as they move their arms out of the way of the camera as it passes their left side. Note: the “Axis” tool can be used to provide a horizontal reference line for setting the vertical position of the images.

4.4.2 Attenuation Correction

4.4.2.1 Basis

As they travel along the path from their origin to the surface of the patient, some gamma rays are absorbed or scattered so that the original emission level is attenuated. If not accounted for, this phenomenon causes the centers of reconstructed images to be dimmer than the peripheries. This dimness causes qualitative unpleasantness in the reconstructed images and quantitative inaccuracies in pixel activity values. While it is presently impractical to correct the attenuation to the point of accurately restoring pixel count values, approximation methods can be used to at least restore the image qualitatively. One approach, which is used by NucLear MAC, is illustrated as follows.

Consider that a tomographic body section can be represented by an ellipse. "x,y" is a point within the ellipse and "A(x,y)" is the activity at x,y. The gamma camera views the point from an angle "ø". The amount of activity that is detected by the gamma camera is a function of A(x,y)exp(-µd(ø)), where "µ" is the linear tissue attenuation coefficient and "d(ø)" is the distance from x,y to the surface of the ellipse along a line perpendicular to the camera. Thus, the attenuation will cause the activity, A, to be underestimated by exp(-µd(ø)). The total underestimation that will occur over a 360 degree acquisition will be the summation of the underestimations for each angle.

As a first approximation, the true activity, A, at x,y can be estimated by dividing the originally calculated value by the normalized summation of the attenuation values for each acquisition angle. This approach is referred to as the Chang method after the author of an article in which it was first described. While non-linearities in the filtered backprojection model make it imperfect, it is nevertheless useful, particularly for qualitative improvement of reconstructed images from tissues with fairly homogeneous attenuation profiles such as the abdomen or head. It is not useful in situations in which attenuation varies greatly from point to point as in the thorax.

One problem with the method is that the effective attenuation coefficient is influenced by scatter, so that effective µ may not be exactly equal to the theoretical µ. This effect can be observed when applying attenuation correction to data from uniform cylinders of activity. Ideally, the Chang method would cause the centers of the sections to be exactly brightened to the level of the peripheries when using a value of µ that is accurate for the particular tissue. However, the degree of brightening will tend to be too great at the center because of the unaccounted for presence of scattered radiation.

While methods have been described that attempt to subtract the scattered radiation, they are somewhat tedious and require the acquisition of a "scatter" image from an energy window below the photopeak in addition to the acquisition of the primary photopeak image. However, if the main desire is to qualitatively improve the image rather than to attempt to quantify pixel data, a simpler method can be employed; µ can be varied until a line-profile from a cylindrical phantom's reconstructed image is flattened.

Of course, phantom data may not apply exactly to patient data, and the scatter component can vary as patient size changes. Therefore, NucLear MAC offers an interactive approach to attenuation compensation. The user can vary µ and observe the effects on the image data through displayed horizontal and vertical line profiles. In other words, the user can choose between the use of some theoretical µ value and arbitrary values that best flatten the profiles. The process, while not intending to provide quantitative attenuation correction, offers the potential advantage of an increased level of user-interaction and control compared to some other methods.

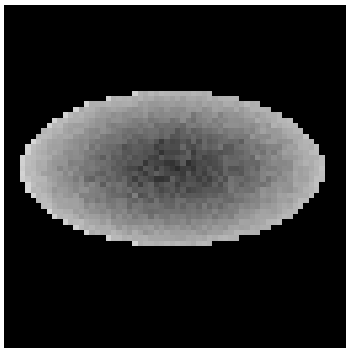
Another important consideration is how best to represent the border of the tissue section. As mentioned earlier, a convenient approximation is the use of an elliptical shape. Of course, an ellipse is not a perfect definition of the border, but it is a close representation for many situations. However, even if an ellipse is accepted as representing the shape of tissue sections, a problem arises if the dimensions of the sections change from slice to slice. For example, the dimensions of transverse brain

SPECT slices increase when moving downward from the vertex. Also, the relative lengths of the axes of the ellipses can vary.

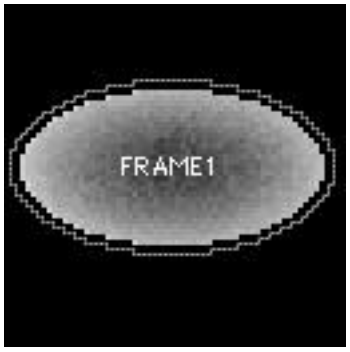
NucLear MAC therefore provides for variations in the size, position, and relative axes-lengths of ellipses for different sections. The basic program works through the implementation of user-defined ellipses. If a user draws the ellipse on one section only, that same ellipse will be used for all sections. However, the user can draw ellipses on more than one section, and the NucLear MAC will create interpolated ellipses for sections between those that are defined by the user. Thus, depending on the relative shapes and positions of the different sections and the desired precision, the user can draw as many ellipses as desired with the computer interpolating the others. For example, if the sections from a brain SPECT gradually increase then decrease in diameter from the apex to the base, then the user may wish to define three ellipses, at the apex, the middle, and the base. The important concept is that the decision is always up to the user for any particular patient, and that it is not difficult to implement. This helps avoid unappreciated artifacts, while increasing processing time only a little as compared to fully automatic methods.

4.4.2.2 Processing

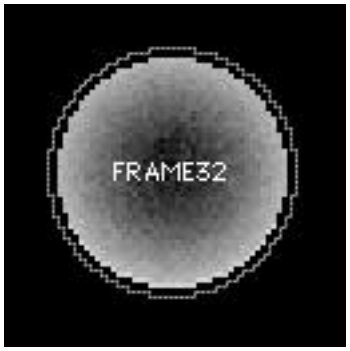
Bring up the window with the reconstructed transverse SPECT sections. If desired, you can draw elliptical regions on selected slices at this time or you can draw them from within the **Attenuation Correct...** selection of the **Analyze** menu. The difference is that if you draw the ellipses before starting the script, you can move them using the “Arrow” tool, but if you draw them from within the script, you can only redraw improperly centered ellipses, not move them. The ellipses should define the approximate boundary of the body section. You can draw an ellipse on one slice or on as many slices as you wish. If you draw an ellipse on one slice, the same boundary is used for all slices. If you draw an ellipse on two or more slices, NucLear MAC interpolates boundaries for in-between slices (slices beyond the drawn-on slices use the endpoints).



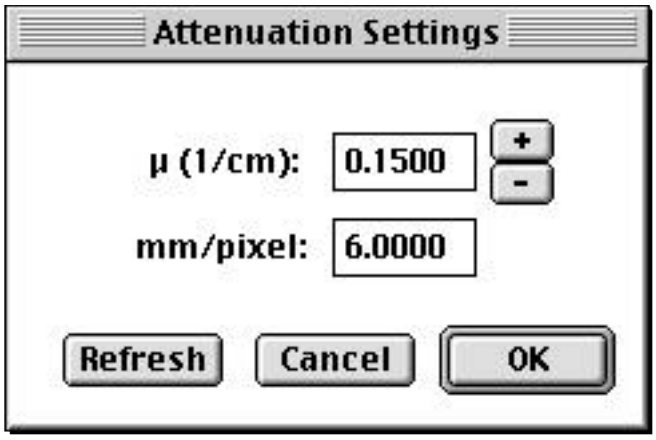
Select **Attenuation Correct...** from the **Analyze** menu. Click on the window to be corrected. If you have already drawn the desired ellipses, hit “Continue”. Otherwise, click on the “Frames” button and move to the frame that you wish to draw your first ellipse on by sliding the mouse horizontally. Click the mouse when you reach the desired frame. Draw the ellipse by holding down the Mouse button. If the ellipse is not to your liking, just redraw it; the old ellipse is replaced.



Hit "Frames" again if you wish to draw ellipses on other slices.



When you are done placing the ellipses, hit "Continue". The program creates four windows; a window showing the attenuation correction applied to a slice, a window that allows the user interaction, a horizontal profile window, and a vertical profile window. The profile windows show the original data's line profiles and the line profiles from the attenuation-corrected image. Type in the value for the mm / pixel. This value depends on specific settings for a given SPECT acquisition. If desired, type in a specific value for the attenuation coefficient (μ).



Click on "Refresh". The attenuation corrected image and line profiles are updated. You can then click on the "+" or "-" buttons and the μ value is increased or decreased in 5% increments and the windows are updated. Proceed until the line profiles are as desired, then click on "OK". The attenuation correction is applied to all slices. This process is very fast if only one ellipse was used, but takes some time if the program must

generate interpolated ellipses and calculate their attenuation parameters. The user-interaction window shows a countdown if interpolated ellipses are generated.

5 Study Types and Information

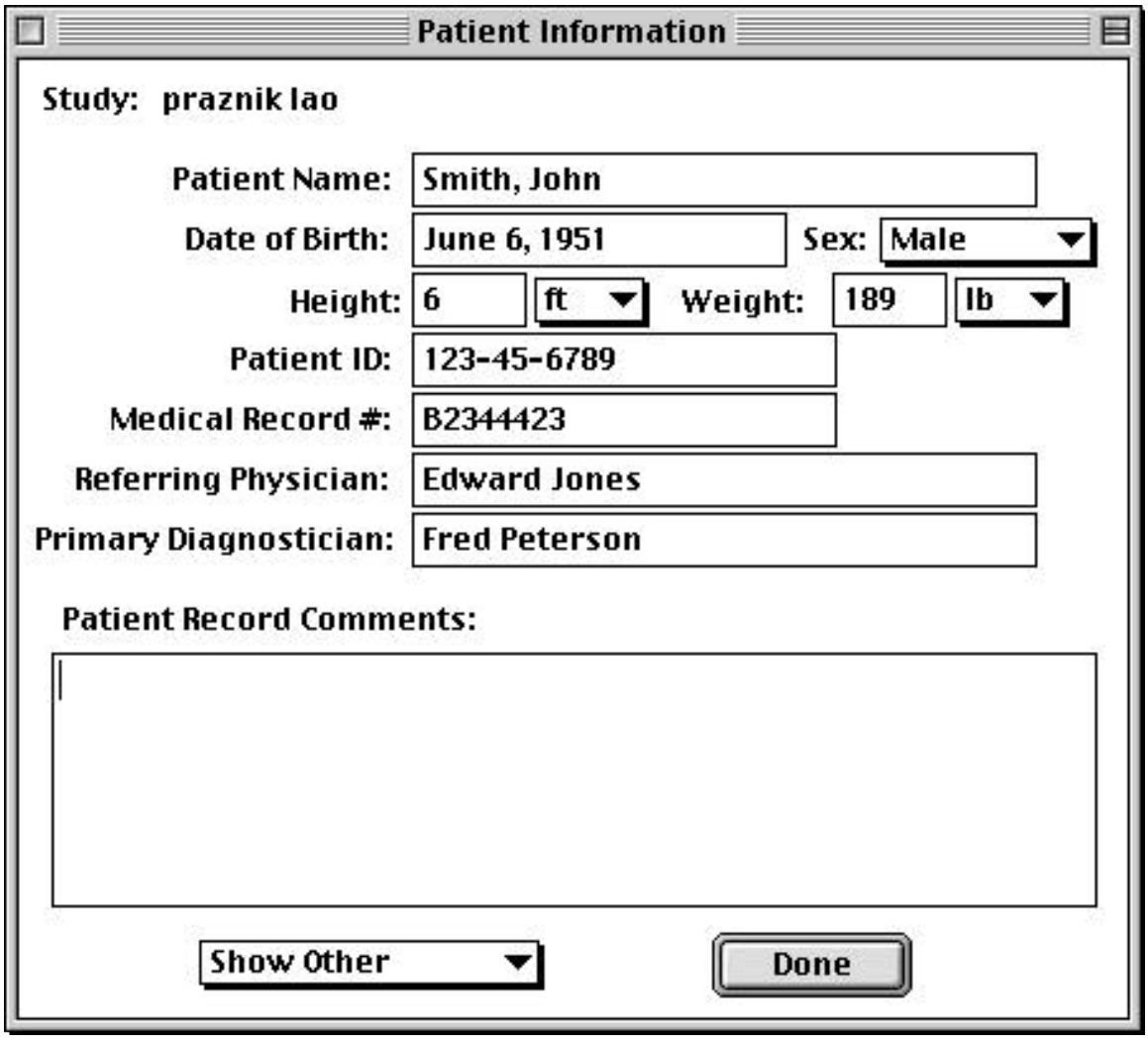
NucLear MAC studies retain substantial information describing the study contents. You can view and edit much of this information with various dialog boxes. To view or edit study information, select the study window and do one of the following:

• Choose the desired type of information from the **Edit>Information** submenu.

• Hold down the control key, click in the content area of the study window, and then select the desired type of information from the resulting popup menu.

5.1 Patient Information

When you choose to edit or view patient information, you are presented with the following dialog:



Patient Information Dialog

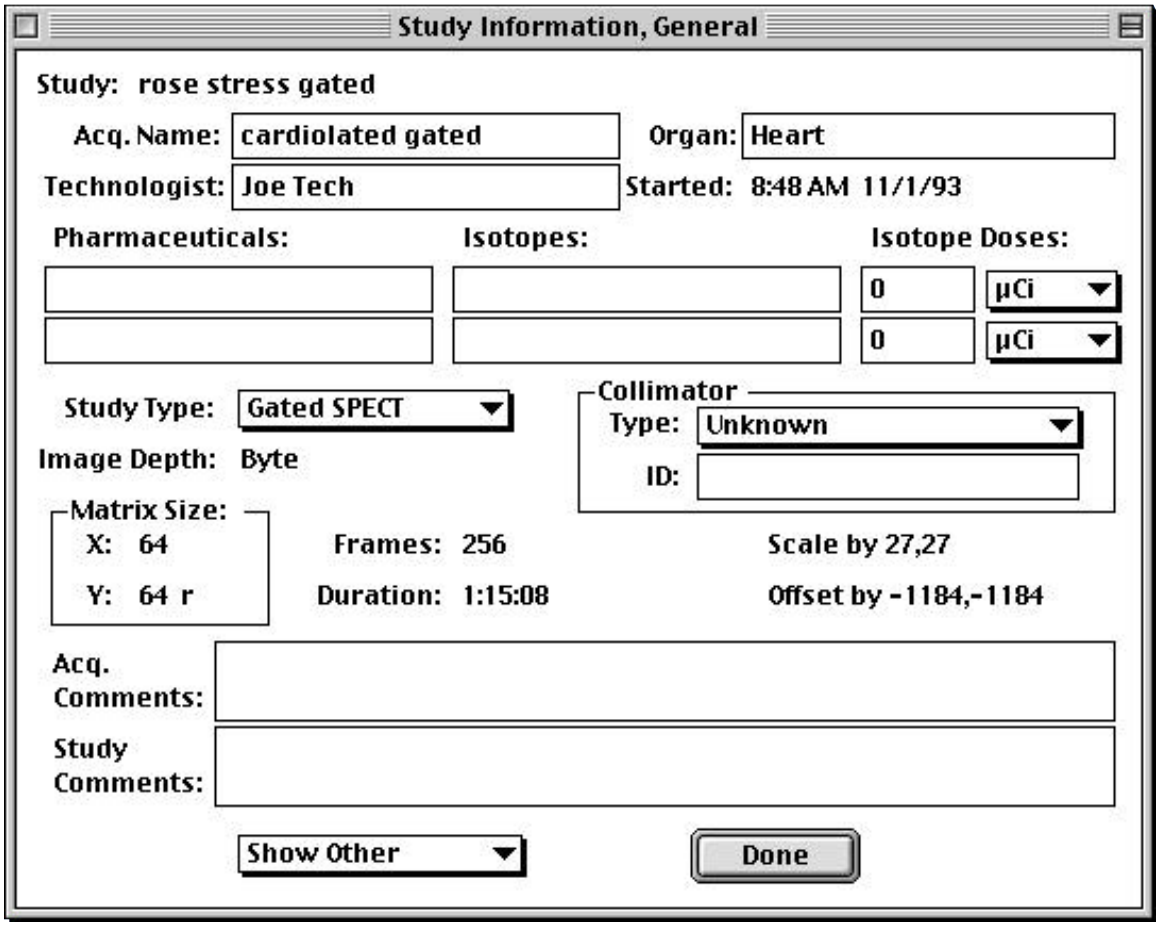
The first line of the Patient Information dialog displays a study name. The patient information applies to that study, and this Patient Information dialog is owned by the study's window. If a study window is closed while its Patient Information dialog is open, then the dialog is closed automatically.

The fields of the Patient Information dialog are mostly self-explanatory. Be sure to select the correct units for the height and weight fields. Use the popup menus located to the right of the fields. For height, choose meters (m), centimeters (cm), feet (ft) or inches (in). For weight choose kilograms (kg) or pounds (lb).

The Show Other popup menu allows you to activate the other informational dialogs or the study that owns this dialog.

5.2 General Study Information

When you choose to edit or view general study information, you are presented with the following dialog:



General Study Information Dialog

The first line of the General Study Information dialog displays a study name. The study information applies to that study, and this General Study Information dialog is owned by

the study's window. If a study window is closed while its General Study Information dialog is open, then the dialog is closed automatically.

For the following field descriptions, the term "original study" refers to the study containing the raw (unprocessed) data acquired from the gamma camera. In many cases, information from an original study is copied directly to studies resulting from processing of the original.

For original studies acquired on a NucLear MAC acquisition system, the Acquisition Name field is the name of the protocol that was used to acquire the original study. This field may have somewhat different meanings for studies imported from other systems.

The Started field contains the date and time when the acquisition of the original study was initiated.

The Study Type popup menu displays the category to which the study belongs. This field is important because it can limit the types of processing allowed on the study. Additionally, this field controls the content of the Study, Specific informational dialog. The possible study types are covered in the next section. Typically you will only need to change a study's type when it has been imported from another system.

The Collimator fields provide information about the collimator used to acquire the original study. The Type field is the category to which the collimator belongs, and the ID field is the identifying label for the particular collimator. For original studies acquired on NucLear MAC systems, these fields will contain useful information only when the acquisition was conducted with the Acquire Any Study dialog, which allows the user to define collimator records.

The Image Depth field displays whether the image pixels consist of bytes (eight bits, 256 values) or words (16 bits, 65536 values).

The Matrix Size fields display the dimensions of the study's images in image pixels. X is the horizontal dimension and Y is the vertical dimension. An "r" next to either dimension indicates that that dimension was acquired in reverse polarity from the default (positive voltages for the right and upward directions).

The Duration field shows how long it took to acquire the study.

The "Scale by" field, when present, displays the size of the original study's image pixels in analog-to-digital converter (ADC) units. In most NucLear MAC acquisition systems, there are 4096 ADC units per five volts of input from the camera in the X or Y dimension. The "Scale by" field contains the X and Y sizes separated by a comma.

The "Offset by" field, when present, displays the horizontal and vertical offsets of the original study's image pixels in the field of the analog-to-digital converter. More specifically, the offsets are the horizontal and vertical differences between the top, left corner of the ADC field and the top, left corner of the image pixel matrix. The ADC field consists of 4096 rows of 4096 elements.

6 Analysis

6.1 Image Processing

6.1.1 View Changes

6.1.1.1 Orientation of SPECT Tomographic Slices

There is much confusion and debate as to what should be the standard orientation and ordering of SPECT data slices. Traditionally, radiographic orientation of computed tomographic (CT) slices is such that the cross-sectional anatomy is displayed as if a supine patient is viewed from the feet, but the order of the cross-sectional slices in a movie or all frames window is usually reversed, running from head to feet. Therefore, the data is not consistent with respect to cubic geometry. This problem in terminology is made worse when non-cross-sectional orientations are considered. For example, anatomic sagittal position is based on viewing the patient from the left side, while the latest recommendation regarding cardiac terminology requires that the heart be viewed from the right side.

In an attempt to reach a consistent compromise, the NucLear MAC uses cubically consistent modified cardiac terminology as a default. In other words, slices are oriented and ordered so that the concept of slicing a cube of data is supported. However, the user has the option to change the order of cross-sectional slices to be consistent with the standard CT approach.

Unless otherwise indicated by the user, SPECT cross-sectional (“transverse”) slices are displayed as if a supine patient is viewed from the feet; the patient’s anterior surface is at the top of the study window and the order of the images within the movie or all frames window is from the bottom to top (reversed from standard CT presentation). “Sagittal” slices are displayed as if the patient is being viewed from the right with the anterior aspect of the patient to the right side of the window and the order of the slices being right side to left side (this differs from anatomic standards in which the patient is viewed from the left side). The order of coronal slices is from front-to-back with the patient’s head at the top of the window. In this manner, the various projections can be derived from the concept of revolving a cube of data and slicing it from front to back.

The order of the initially reconstructed cross-sectional slices can be changed by checking the “Top to bottom frame order” box in the SPECT reconstruction settings window if the user prefers the traditional CT approach.

When reconstructing cardiac images, the data can be flagged for later automatic use of recently published guidelines for standard oblique display terminology. This is done by checking the “Use heart terminology” box in the SPECT reconstruction settings box. Cross sectional slices are then consistent with horizontal long axis nomenclature. Menus and windows are labeled as “H. long axis” and displayed with the apex at the top of the window and with the order of frames being inferior wall to anterior wall.

Short axis slices are labeled as "short axis" and displayed in apex-to-base order with the anterior wall at the top of the window. Vertical long axis slices will be labeled as "V. long axis" and displayed with the apex to the right side of the window and in right-to-left (septum to LV lateral wall) order.

Thus, the cubic geometry and viewing perspective is preserved between cardiac and non-cardiac images in the default mode (with the "Top to bottom frame order" box not checked).

6.1.1.2 Sagittal or Coronal SPECT Images

Reconstructed SPECT images initially appear as transverse sections. At times, you may wish to view them at some different angle. To create orthogonal slices, use the commands **Transverse to Coronal** or **Transverse to Sagittal** in the **Display>Views** submenu. If the study employs heart terminology, the menu commands are **H. Long to Short Axis** and **H. Long to V. Long Axis** respectively.

Sagittal slices are displayed as if the patient is viewed from the right with the anterior aspect of the patient to the right side of the window. The order of the slices is right side to left side (this differs from anatomic standards in which the patient is viewed from the left side). The order of coronal slices is from front to back with the patient's head towards the top of the window.

Note: In order to conserve memory, you should use the **New Window** selection from the **Regions>Utilities** submenu to eliminate unwanted background from the transverse sections before creating the other views.

6.1.1.3 Oblique Axis SPECT Reconstructions

Oblique angle reconstructions are also possible, e.g. along the long axis of the heart in a thallium SPECT study. NucLear MAC allows selection of any oblique angle using a rapid and easy-to-perform method.

Bring the study window containing the reconstructed transverse images to the front. Choose **Set View Angle** from the **Display>Views** submenu. A new window appears. The window is a volume mapping of the SPECT data. (The image in the window is created by adding up the pixels in the SPECT data that are perpendicular to the viewer, in other words, a pseudo-planar image is created.) The user can change the viewing angle with the mouse. Moving the mouse to the left or right rolls the data horizontally and moving the mouse up and down rolls the data vertically. The two motions can be performed at the same time, but it is usually easier to limit the motion. Press the horizontal arrow on the keyboard to limit the motion to horizontal only. Press the vertical arrow on the keyboard to limit the motion to vertical only. Press the option key to spin the data in the viewing plane.

If you want to change the size or contrast of the rotation window, hold down the shift key and the rotate function will be temporarily interrupted. Then enlarge the window by

dragging on the size change box. Release the shift key and the rotation function resumes.

Move the data set until you are viewing the data from the front of the desired orientation. For example, to create short axis images, roll the data cube until you are viewing the heart from its apex with the right ventricle on the left side of the image (down the barrel of the ventricle like a 45 degree LAO view). Then click on the mouse to exit this function.

Now choose **Reslice from view** from the **Display>Views** submenu. A new window appears that contains the data reconstructed along the selected angle (front to back).

6.1.2 Filtering

6.1.2.1 Filtering Images with Kernels

Basis

Images are filtered to extract or enhance information. For example, a noisy image may be filtered with a smoothing filter to suppress noise and better demonstrate objects of interest. Filters can be one-, two-, or three-dimensional. For the purposes of this explanation, we will use one-dimensional filters as examples; the principles can then be extrapolated to two- or three-dimensional filters.

Consider one row of a 32 x 32 image of background activity. Statistical noise causes some random variation in the data. Note: the "noise" in the example below is not exactly random but serves the purpose of illustrating the effects of filtering.

```
1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3
```

The average value is 2 counts per pixel but the actual values are all 1 or 3. We can smooth the data to remove the noise with the proper kernel; in this instance a 1 x 3 dimensional kernel with values of 1 2 1. Note: kernels always have odd numbers of elements, e.g. 3, 5, 7, etc. The kernel is applied as follows: for any pixel value, p_i, the new value after application of the kernel with values k1 k2 k3 is $(k1* p_{i-1} + k2* p_i + k3* p_{i+1}) / (k1 + k2 + k3)$. If we choose any pixel in the data and apply the kernel we will get either (1 + 6 + 1) / (1 + 2 + 1) or (3 + 2 + 3) / (1 + 2 + 1), both of which equal 2. Thus, by applying the kernel to every pixel we transform the data into:

```
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
```

Now imagine that we have a one-pixel wide bright point near the center of the image having 2 counts per pixel above background.

```
1 3 1 3 1 3 1 3 1 3 1 3 1 3 3 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3
```

It is not so easy to see the point in the presence of the noisy background. However, if we apply our smoothing kernel, the point stands out better.

```
2 2 2 2 2 2 2 2 2 2 2 2 2 2.5 3 2.5 2 2 2 2 2 2 2 2 2 2 2 2 2 2
```

Note that we have sacrificed some resolution; the data from the bright point is spread over 3 pixels.

With respect to smoothing kernels, kernels with larger sizes or with outer values more similar to the center value increase the smoothing.

Examples: 2 2 2 or 1 1 2 1 1 are smoother than 1 2 1.

Some kernels contain negative values; such kernels can be used to enhance contrast. For example, consider the kernel -1 1 2 1 -1. Applying this kernel to our image with the bright center point yields:

```
1 1 1 1 1 1 1 1 1 1 1 2 2 5 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

Now the center point stands out even better because, in addition to brightening the center, the kernel has suppressed the background from an average value of 2 to an average value of 1.

In general, a user will wish to filter an entire two-dimensional image, rather than just one line. For this process, two-dimensional kernels as opposed to one dimensional kernels are used. For example, the two dimensional equivalent to our one dimensional smoothing kernel is:

```
1 2 1
2 4 2
1 2 1
```

The two dimensional kernel is applied to all of the pixels around the center pixel that is being filtered.

Specific cases where one-dimensional filters are used are temporal filtering and SPECT filtering.

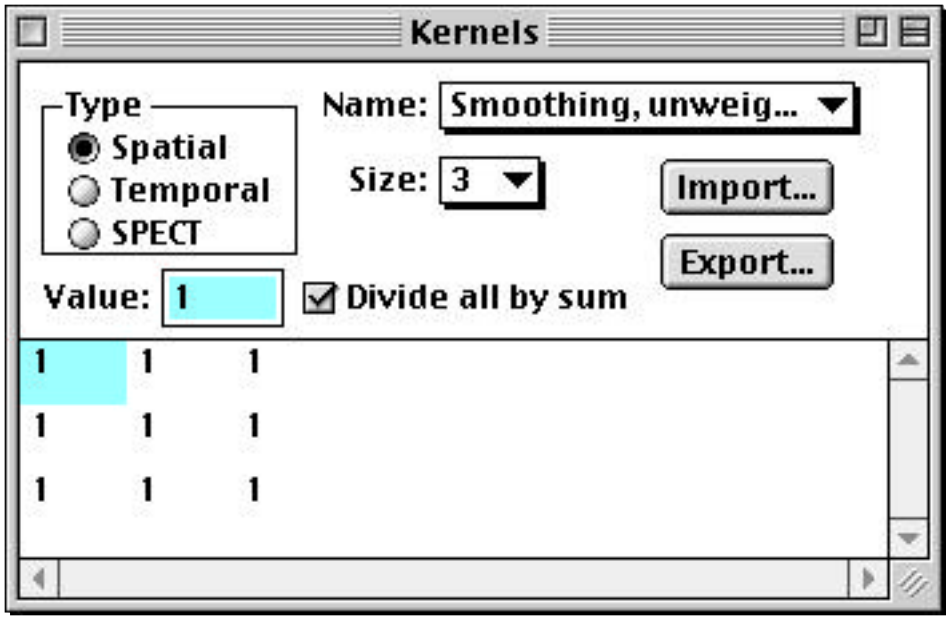
With temporal filtering, a one-dimensional smoothing filter is applied to images in a cinegram. The filter is applied to the same pixel for three or more adjacent frames in the cinegram rather than to pixels within the same frame. This reduces noise from frame to frame as the cinegram is played.

In SPECT filtering, a complicated one-dimensional filter is applied to each row of data before it is backprojected. This process recovers the image data in the transverse reconstructed slices. It will be described in more detail in the section **SPECT Reconstruction**.

Processing

Images can be processed with user-defined convolution kernels. The method of creating such kernels will be demonstrated at the time of installation. Basically, the **Kernels** selection from the **Windows** menu is activated and the kernel is then typed into the dialog box that appears. These kernels can be any size, e.g. 3 x 3, 5 x 5, etc., up to 21 x 21. Larger matrices better approximate Fourier filters, but require more

processing time (values of 11 to 15 can approximate even complicated Fourier filters). Use size popup menu to set the dimension of the kernel. The new kernel is then saved by adding it to the **Type** popup menu. Also, some example filters are provided with the software in a folder labeled “Filters”. To add them, double-click on the file named “NucLear MAC filters”. For standard kernels in which the sum of the products of the kernel and pixel values are divided by the sum of the kernel elements, make sure that the "Divide by all sum" box is checked. If the sum of the kernel values is zero, the program will substitute one for the divisor to avoid a divide-by-zero error.



Some useful kernels:

SPATIAL

Unweighted smooth

```
1 1 1
1 1 1
1 1 1
```

Weighted smooth, i.e. NEMA-type

```
1 2 1
2 4 2
1 2 1
```

SPECT resolution enhancement prefilter - enter as a spatial filter (similar to Metz)

```
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1  0  1  1  2  1  1  0 -1 -1
-1 -1  1  2  3  3  3  2  1 -1 -1
-1 -1  1  3  5  6  5  3  1 -1 -1
-1 -1  2  3  6  8  6  3  2 -1 -1
-1 -1  1  3  5  6  5  3  1 -1 -1
-1 -1  1  2  3  3  3  2  1 -1 -1
-1 -1  0  1  1  2  1  1  0 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
```

TEMPORAL

Unweighted smooth

```
1 1 1
```

Weighted smooth

```
1 2 1
```

SPECT

Ramp Equation (center pixel number = 0): i = pixel number, p = term in kernel, p = π/4 when i = 0, p = 0 when i is otherwise even , p = -1 / (π i2) when i is odd. Scale values to 2400 for convenience.

Ramp (27 element)

-6 0 -8 0 -12 0 -20 0 -39 0 -108 0 -972 2400 -972 0 -108 0 ... 0 -6

Shepp-Logan Equation (center pixel number = 1): p1 = -1 / [(2π a) (i2 - 0.25)], p = R p1(i) + 0.5 RR p1(i-1) + 0.5 RR p1(i+1), where R = 1, 0.9, 0.8, 0.6, 0.4 and RR = 1 - R. Vary R to change sharpness with lower values being softer. Scale terms for convenience.

Shepp-Logan 1.0 (27 element)

-4 -4 -5 -6 -7 -9 -12 -17 -24 -38 -69 -160 -800 2400 (mirror about 2400 as above)

Shepp-Logan 0.9 (27 element)

-4 -4 -5 -6 -7 -9 -12 -17 -25 -39 -72 -187 -608 2080 (mirror about 2080)

Shepp-Logan 0.8 (27 element)

-4 -4 -5 -6 -7 -10 -12 -17 -25 -40 -75 -215 -416 1760 (mirror about 1760)

Shepp-Logan 0.6 (27 element)

-4 -4 -5 -6 -8 -10 -13 -17 -26 -41 -81 -270 -32 1120 (mirror about 1120)

Shepp-Logan 0.4 (27 element)

-4 -4 -5 -6 -8 -10 -13 -18 -26 -43 -87 -325 352 480 (mirror about 480)

Once the kernels are defined they are accessed as in the following example. Activate the study window containing the desired image or images to be filtered. Choose a type of filter from the **Analyze>Spatial Convolution** submenu. The process can take several seconds, during which time the cursor becomes an hourglass. The new, processed image then appears in a new window. Note: the original image remains in its window. Follow the same process for other types of filtering.

Time smoothing is performed using the **Temporal Convolution** submenu.

Filtering for backprojection is applied with the **Analyze>SPECT...** menu selection.

6.1.2.2 Filtering Images with Fourier Filters

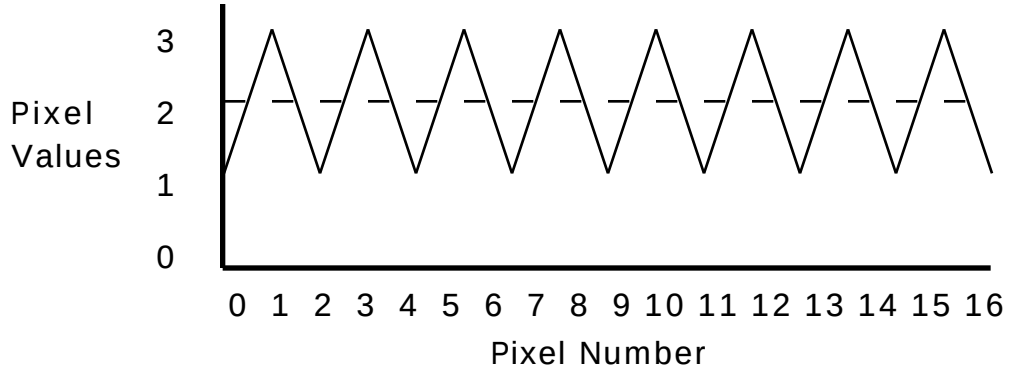
In addition to filtering images by convolution kernels as described above, NucLear MAC can filter images by Fourier filtering.

Basis

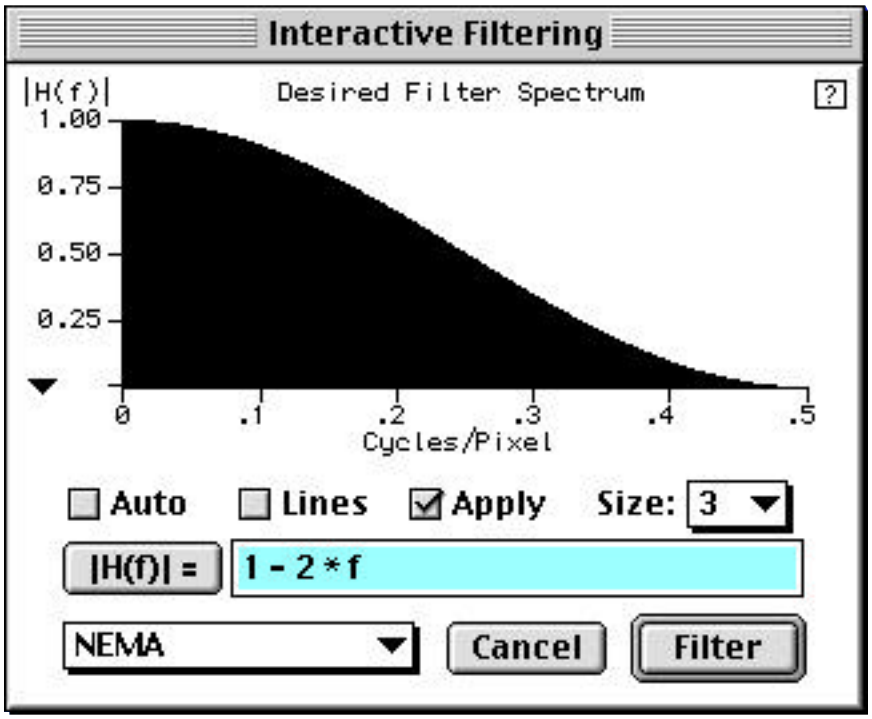
While Fourier filters operate in the frequency domain and convolution kernels operate in the spatial domain, the two types of filters are related to one another. If the size of the convolution kernel is great enough, then any Fourier filter can be approximated by a kernel. Similarly, any kernel which is symmetric about its center pixel can be approximated by a Fourier filter. Returning to the example used in the section describing kernels,

```
1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3
```

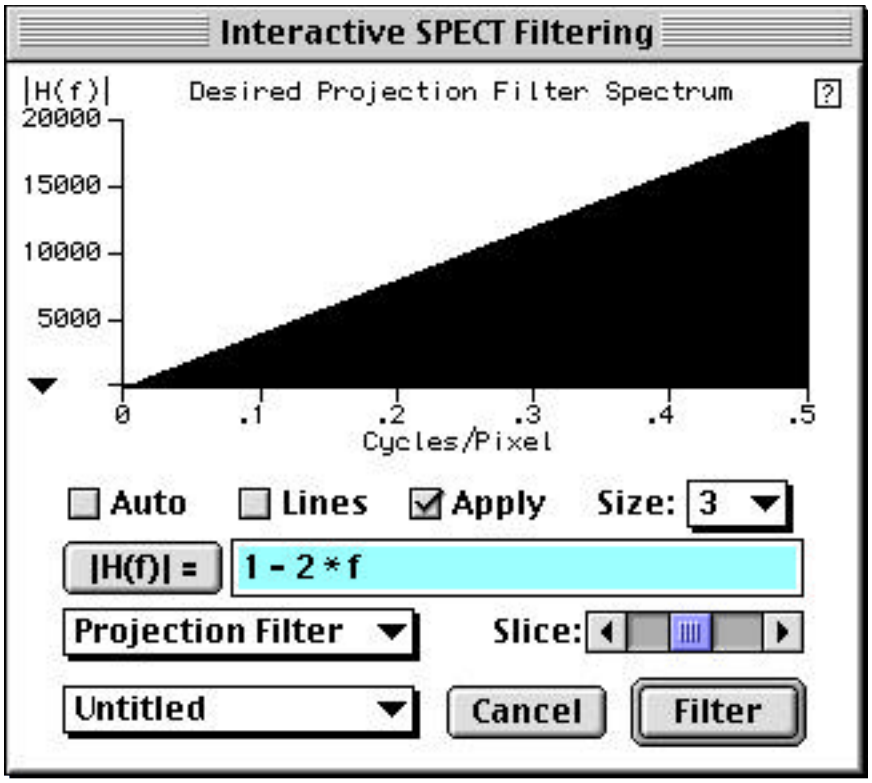
we could make a plot of activity versus pixel number. The first 16 pixels would appear as:



If we imagine that a sine wave can represent the plot, then we would have a sine wave with a frequency of one cycle per 2 pixels, or more conventionally, 0.5 cycles per pixel. We desire to smooth out the data. We know that we can smooth the data with our 1 2 1 kernel in the spatial domain. The question is, how can we smooth the data in the frequency domain? The answer is that we can perform a Fourier transform of our 1 2 1 smoothing filter and apply it in the frequency domain. The Fourier transform is generated by a standard method known as Fast Fourier Transform, or FFT. The resulting transform appears as illustrated in the graph below.



The graph tells us that we wish to preserve data with frequencies below 0.5 cycles per pixel (y axis value or multiplying factor near 1) but that we wish to suppress data with frequencies near 0.5 cycles per pixel (y axis value or multiplying factor near zero). We can make Fourier filters that suppress differences between neighboring pixels as illustrated above or that amplify differences between neighboring pixels as illustrated below.

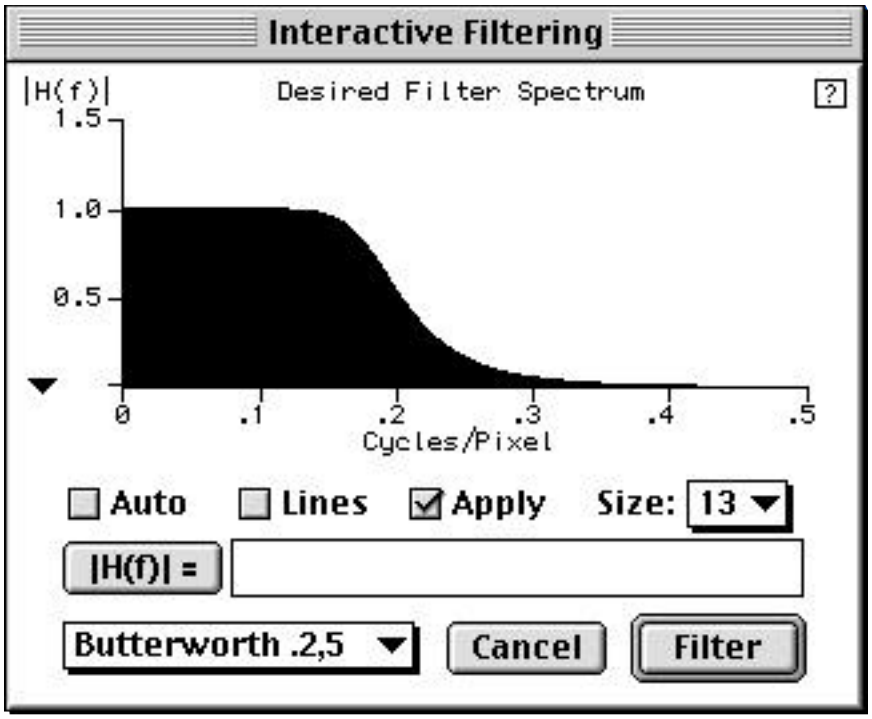


This difference-amplifying type of filter, which is known as a ramp filter because of its shape, is very important in SPECT because it is used to filter data before backprojection so that proper two-dimensional transverse sections can be generated from the projection images.

We can also make filters that suppress high differences between pixels (reduce noise) and suppress low differences (reduce background) while leaving mid-range differences alone. These filters are known as resolution-recovery filters because they can sharpen the normal blur in camera images caused by imperfect collimators and electronics without amplifying noise. Common types are known as Wiener and Metz filters after the persons who first described them. Such filters are commonly used as two-dimensional prefilters to improve the quality of SPECT images.

Processing

Activate the window containing the image or images to be filtered. Select **Interactive...** from the **Analyze>Other Filtering** submenu. The following window appears.



If you have already made some filters, then select a filter from the popup menu in the lower left, as illustrated by Butterworth .2,5 in the example above. The graph displays the shape of the desired filter spectrum. The actual filter spectrum used is an approximation of the desired filter spectrum. To view the shape of the actual filter spectrum, click on the small "?" box in the upper right. To improve the approximation of the actual filter to the desired filter, increase the value in the "Size" popup menu in the middle right (but processing time will increase). Values from 11 to 15 can adequately approximate most desired filter spectrums.

Filters can be made by one of several methods. Activate a study window containing an image or images to be filtered. The "Apply" box should be checked to see the effects of the filter on the original image.

1) The popup menu shows any Spatial convolution kernels that have been created and saved using the **Kernels** selection of the **Windows** menu (see the section **Filtering Images with Kernels**). To change a kernel into a Fourier filter, first select it from the list, then change the "Size" parameter to the desired value. The filter popup menu now reads "Untitled". Add the new name of the filter to the filter popup menu and it is stored as a Fourier filter.

2) Draw the desired filter by holding down the mouse button and drawing in the graph window. (To preserve scaling, make sure that "1" is selected from the popup menu indicated by the small triangle beneath the y axis.) Again select the "Size" value. To check to see how accurately the filter has been approximated, click on the "?" box.

3) Type the filter equation in the equation area (just right of the "|H(f)|" box). Then click on the "|H(f)|" box to activate the equation.

If the images to be filtered are in a movie window, the processes above will only show the effects of the filtering on the front image. To filter all images once a desired filter has been chosen, click on the “Filter” button.

6.1.2.3 Interactive Fourier Filtering

Interactive Fourier filtering is a very powerful tool in which the effects of changing equation parameters on a filtered image can be observed in real time.

MAKING THE FILTERS

First the interactive equation must be created. Select **Equation Plotting…** from the **Regions>Manual Plot** submenu. We will illustrate the process by making an interactive Butterworth filter. The Butterworth equation is:

1/sqrt(1+x/cutoff frequency)^order times 2)

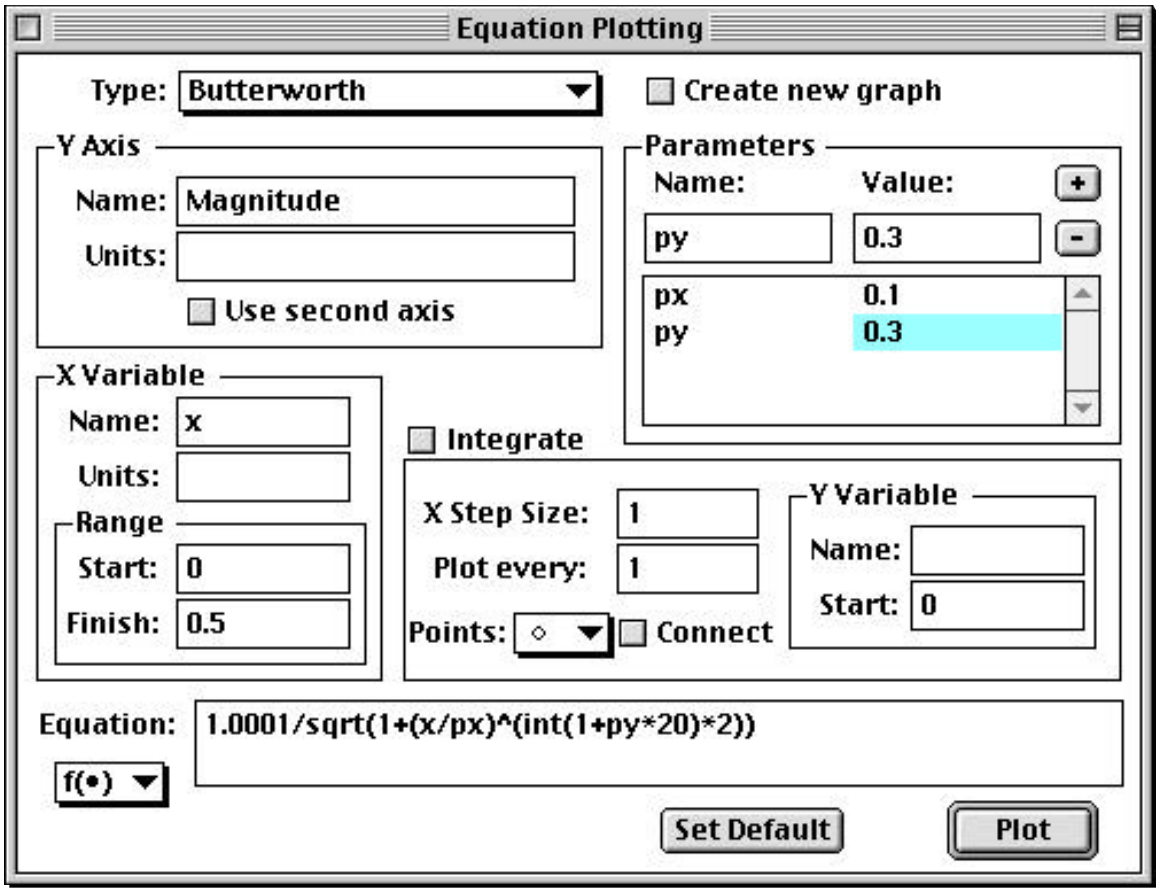
We want to be able to adjust the “cutoff frequency” and “order”. Call the “cutoff frequency”, “px”, and the “order”, (1+py*20). The reason for the complicated description of “order” is that the parameter “py” can only be varied between 0 and 0.5. The expression (1+py*20) thus allows a range of 1 to 10 for the “order” which is a useful range. For the numerator, 1, we substitute 1.000001; this results in better scaling of the graph.

Type the equation below into the equation area.

```
1.000001/sqrt(1+(x/px)^(int(1+py*20)*2))
```

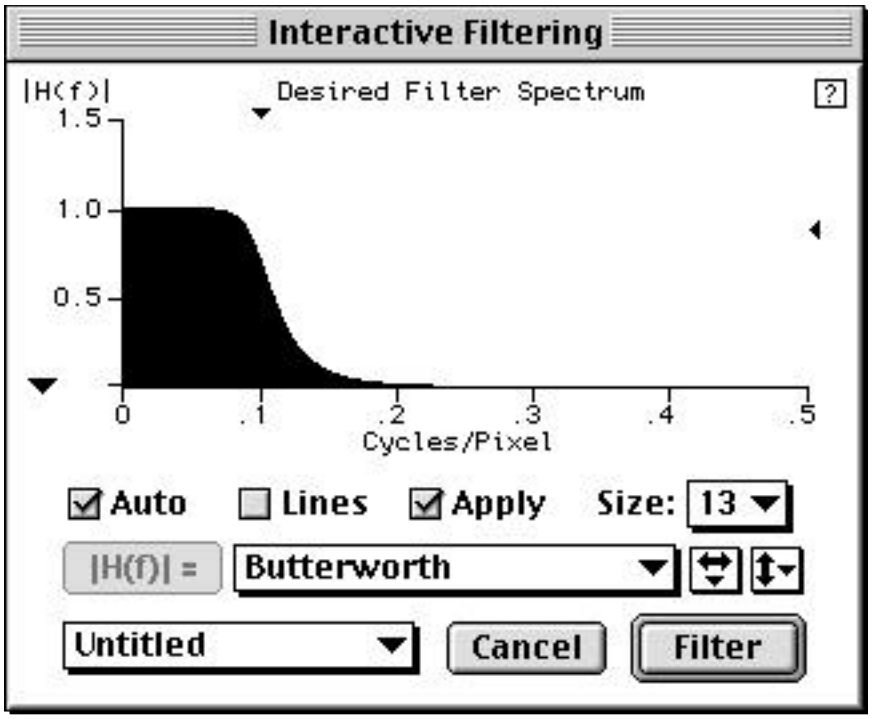
Type px into the parameter box and set a default value of 0.1. Click on the + sign to add the parameter to the list. Click on the + sign again to add a second parameter. Type py as the second parameter and set a default value of 0.3.

To view the filter, click on “Plot”. The filter graph is displayed. Note that in cases where the graph goes negative, the negative values are set to zero when the graph is used as a Fourier filter. Reactivate the “Equation Plotting” window. Add a name to the “Type” popup menu for the newly created filter. Leave the other settings in the window alone. The window should now appear as:



USING THE FILTERS

Activate a study window window. Select **Interactive...** from the **Analyze>Other Filtering** submenu. When the "Interactive" window appears, click on the "Auto" button. A popup menu of available interactive Fourier filters appears next to the "|H(f)|" button. Select the desired filter. Set the "Size" value, e.g. 13. Click on the horizontal arrow (just to the right of the filter list) and select the parameter to be controlled by horizontal movement of the mouse (px in our Butterworth example). Click on the vertical arrow and select the parameter to be controlled by vertical mouse movement (py in our Butterworth example).



To change the “cutoff frequency” of the filter, drag the small triangle at the top of the graph with the mouse. To change the “order”, drag the small triangle at the right side of the graph

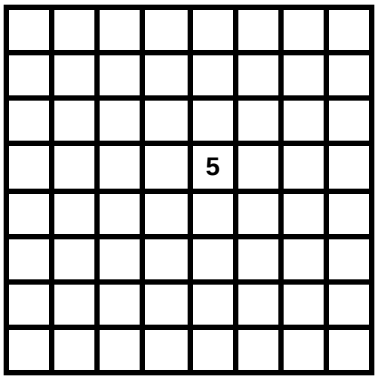
The new study window shows the effects of the changes. If the original window was a movie window, click on the “Filter” button to filter all images after you have selected the desired filter parameters.

6.1.3 SPECT Reconstruction

6.1.3.1 Basis

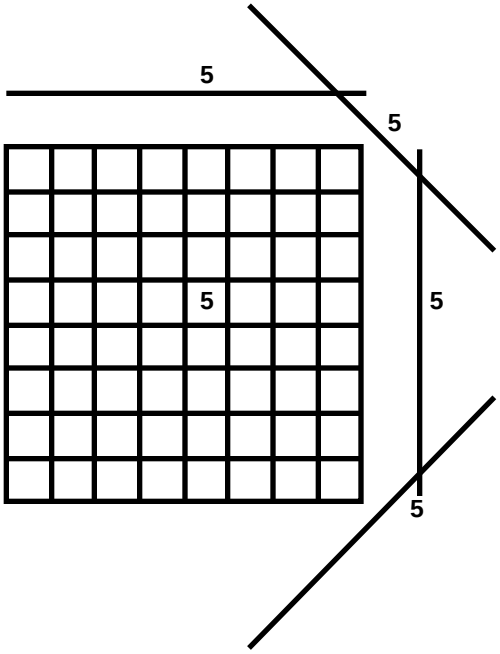
Single photon emission computed tomography (SPECT), put succinctly, is the process whereby projection image data obtained from a gamma camera is filtered and backprojected to generate a backprojection image that reflects the activity distribution in a cross-sectional image plane of the object of interest. Confusing? Not if considered step-by-step.

Consider an object containing activity as in the example below. A one-pixel thick slice of the object is illustrated. For the purposes of this example, we will assume that there is a point source containing 5 units of activity located near the center of the object and that there is no activity in the remainder of the object.



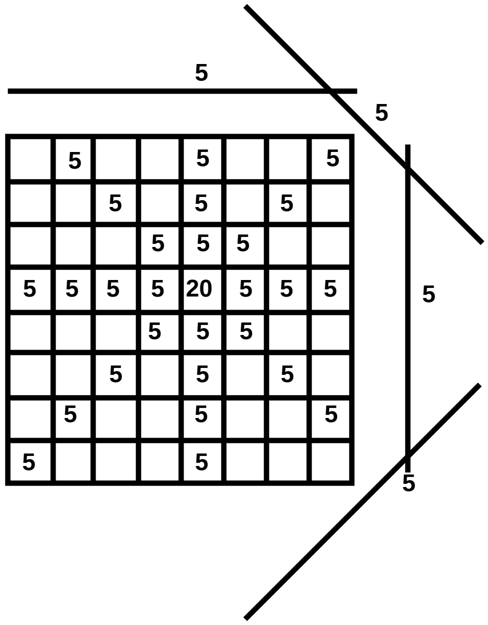
Original Image Plane: A point source containing 5 units of activity is located near the center.

For simplicity, we will also assume that there is no attenuation of activity as the emitted photons travel outward from the point source. Therefore, if we rotate a one-pixel-axial-length gamma camera about an axis perpendicular to the slice, we can obtain images that each show the location of the source along one dimension of the slice. (Of course, gamma cameras are actually many pixels in axial length so that data for many slices can be obtained at each view; the following discussion applies to each slice.)



Projection Images: Images are obtained at equally spaced angles from around the image.

Next, we generate a matrix in the computer memory with which we will attempt to create an image (the backprojection image) that reflects the two-dimensional activity distribution in the original object. As a first approximation, we can simply project the data from each gamma camera image perpendicularly to the angle that it was obtained relative to the backprojection image. This process, known as simple backprojection, will generate a matrix in which the pixel associated with the point source will have a greater value than the other pixels. However, other pixels along lines from the source to the gamma camera images will also have non-zero values. This phenomenon will cause the backprojection image to have a "star" artifact as compared to the true activity distribution in the object slice.



Simple Backprojection: The projection image data is backprojected to generate the backprojection image. The back-projection image approximates the original image, but a "star" artifact is present.

The “star” artifact can be reduced by filtering the data before backprojecting it. The standard type of filter that applies to this geometry is known as the “ramp” filter.

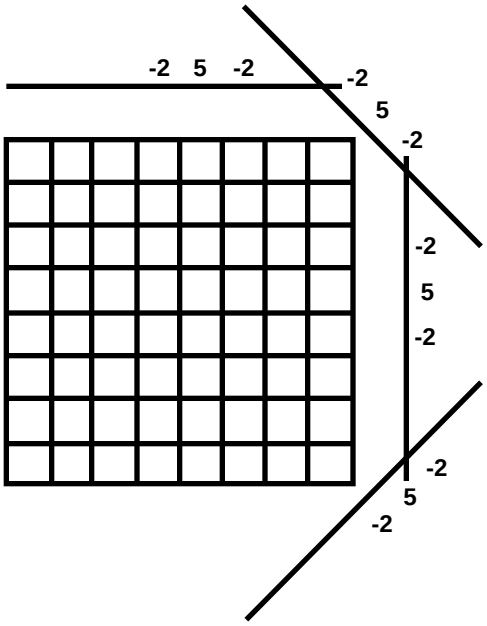
The ramp filter can be expressed in frequency (Fourier) or spatial (convolution) domain. The two methods are equivalent, and the choice of whether to process SPECT data using Fourier or convolution analysis largely depends upon which method is optimal for a given computer. Fourier processing requires a large amount of floating-point arithmetic and can be slow if the computer does not have a floating-point or array processor. On the other hand, convolution processing requires a large amount of memory. Because the Macintosh supports large memory sizes, and because the cost of memory is inexpensive compared to the cost of dedicated array processors, the NucLear MAC performs SPECT reconstruction using convolutions in the spatial domain.

The ramp filter (in the spatial domain) causes oscillating negative values to be placed to each side of the original positive gamma camera data values. The negative values are separated by zero terms and the amplitude of the negative values diminishes with distance from the center. For example, the ramp filter can be expressed as a series of “n” terms, where “n” is the number of pixels in the gamma camera data.

... -0.008 0 -0.016 0 -0.045 0 -0.405 1.000 -0.405 0 -0.045 0 -0.016 0 -0.008 ...

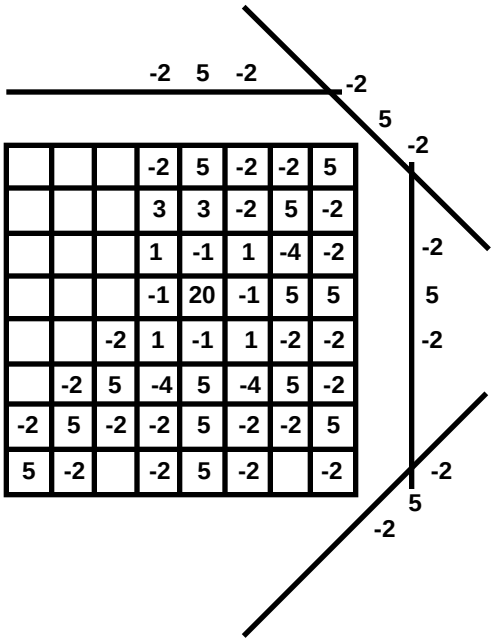
Because the amplitude of the negative terms decreases rapidly with distance away from the center, the number of terms that are actually used in the filtering is generally limited to a value where the outermost term is less than some fraction of the center term, e.g. 0.5%. Terms beyond such low thresholds increase computing times without significantly improving the images.

For illustration purposes consider a simple three-term filter: -0.4 1.0 -0.4. If the filter is applied to the example projection data, the following values will result.

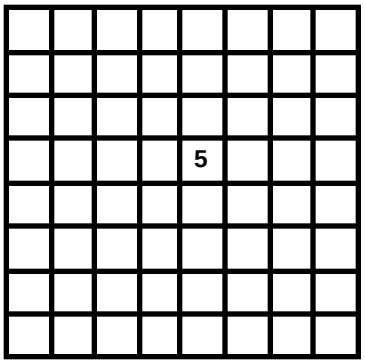


Filtering: The projection image data is filtered using a ramp filter. The ramp filter causes negative data values to occur to each side of positive values.

Then, if the filtered data is backprojected, the backprojection image will more closely reflect the "true" activity distribution in the object slice.



Filtered Backprojection: The negative values offset the positive values in the "star" artifact, but do not affect the region where the activity was located.



Filtered Backprojection: If the filter is chosen properly, the negative terms will nearly cancel the positive values in the "star" artifact so that the backprojection image accurately reflects the original image plane (after scaling).

Because of the finite number of images (e.g. 32 or 64) obtained during SPECT acquisitions, filtered backprojection is an approximate technique; it cannot perfectly generate the original activity distribution in the object, even if the camera data were "noise free".

Additionally, both statistical and camera-generated noise is present in the gamma camera images. This noise is amplified by the ramp filter and the resulting image may be unacceptable for viewing and interpretation. Therefore, some type of smoothing is usually applied to the data before backprojection.

The smoothing can be applied by modifying the backprojection filter and various approaches (e.g. Butterworth-type filtering) and degrees of filtering can be used depending on the characteristics of the gamma camera data.

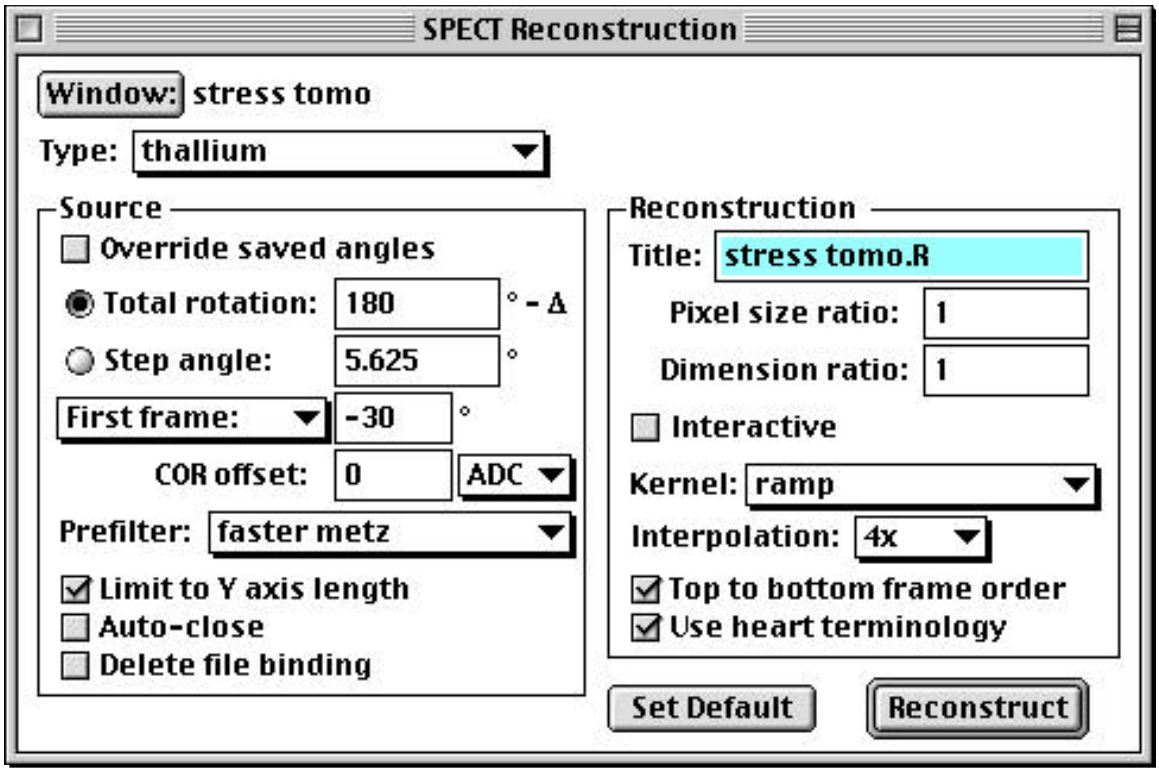
Alternatively, smoothing can be applied to the two-dimensional gamma camera images using a pre-filter before the back-projection step. The SPECT images can then be generated using the ramp backprojection filter. While this procedure requires more processing time than the use of modified backprojection filters, it can better reduce noise because the smoothing is derived from two-dimensional as opposed to one-dimensional effects.

6.1.3.2 Processing

Note: While the processes of Fourier and convolution analysis are equivalent, the relationships between the various rolled-off ramp filters in the frequency domain and the modified damped sinusoids in the spatial domain are not intuitive. Examples of convolution filters with various degrees of smoothing are therefore provided as a base from which the user can construct the desired final filters. However, it must be emphasized that the theory behind the creation of filters is beyond the scope of this manual.

Backprojection, filtering, and other SPECT processing steps should only be performed by persons trained and experienced in their use.

To reconstruct a SPECT series, choose **SPECT...** from the **Analyze** menu.



For most applications you should leave the "Override saved angles" checkbox unchecked. The next three lines are then ignored. In those rare cases where the correct angle information is not stored with the study, then you can set the specific angles. Enter the "Total rotation" value in number of degrees. This number can be negative. The degree increment is calculated automatically. (Note that the total rotation is normally 180 or 360.) Select whether the starting angle is at the first frame of the study or the current active frame. Enter the starting angle in degrees.

NOTE: NucLear MAC assumes normal tomographic patient positioning, i.e. patient supine, head towards gantry, patient anterior equals 0 degrees, leftward (clockwise camera movement) to the patient is positive. For counterclockwise motion, or feet-in position, use negative degree increments. When overriding saved angles, be very careful with these orientations. Errors can cause images to be reconstructed in a mirrored fashion, or upside-down.

Enter the Center of Rotation (COR) offset, if any. A value of zero means that the center of the image is assumed to be the center of rotation. When properly calibrated, NucLear MAC acquires all studies such that the COR offset can be set to zero.

If you wish to reconstruct only a selected group of slices rather than all of the slices (to save time and memory), use the "Axis" tool. Only those slices contained within the vertical length of the axis will be reconstructed if the "Limit to Y axis length" box is checked.

Choose a prefilter if desired (metz, for example) and then a backprojection filter from the Kernel popup menu (this should be a ramp). Check the "Top to bottom frame order" if you wish the SPECT slices to be ordered top to bottom; otherwise they will be

bottom to top. Check the “Use heart terminology” box if your wish standard cardiac orientation and nomenclature (short axis, vertical long axis, horizontal long axis) for oblique axis reconstructions; otherwise standard SPECT orientation and nomenclature are used (transverse, sagittal, coronal). Select an “Interpolation” value; greater values give more accurate results but slow processing slightly. “4X” is a common value. Then click on “Reconstruct”. The images are filtered, interpolated, and reconstructed and then appear in a new window as a set of transverse images. Normally, the original raw acquisition data window will remain open, but it will be closed automatically to save memory (after the filtering) if the “Auto close” box is checked.

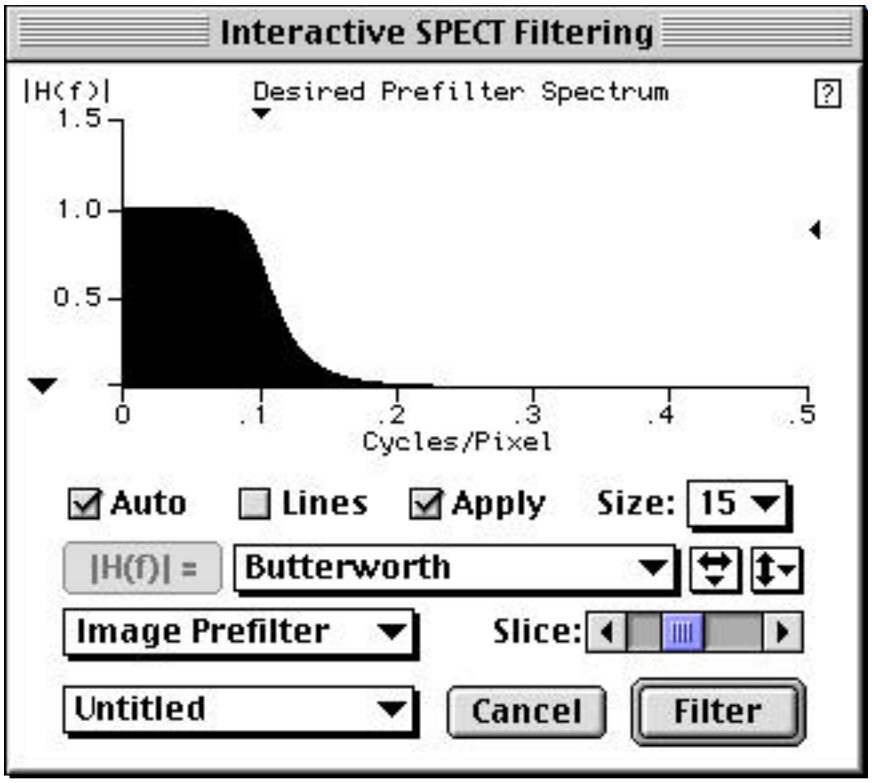
Note that you can save the settings for later use by naming them in the “Type” popup menu. See also the section on Bull’s-eye plotting regarding naming the SPECT reconstruction settings.

If you wish to interactively generate two-dimensional prefilters or backprojection filters, check the “Interactive” box. For more details on this important feature, please see the **Interactive SPECT Reconstruction** section below.

6.1.3.3 Interactive SPECT Reconstruction

NucLear MAC offers the ability to generate two-dimensional prefilters and backprojection filters and to observe their effects on SPECT reconstructions in real. To use this feature, activate a window containing raw SPECT data, place the axis to define the slices to be reconstructed, select **SPECT…** from the **Analyze** menu, enable the “Interactive” checkbox in the SPECT Reconstruction dialog box, and click on “Reconstruct”. The interactive Fourier filtering dialog box appears. Select “Image Prefilter” from the middle popup menu. Click on “Auto” to activate the interactive Fourier filters that you have made (see “Interactive Fourier Filtering”). Select one from the list and set the “Size” value, e.g. 13. Then adjust the parameters as you would with interactive Fourier filtering. The difference will be that the program now prefilters the SPECT data using the Fourier filter and backprojects the center slice. To change slices, move the “Slice” slider control. After selecting the desired filter, click on the “Filter” button to reconstruct all slices.

Once you have made a filter that you wish to use on other studies, add its name to the pop filter menu in the lower left. This name can now be selected from the “Prefilter” list in the standard SPECT reconstruction dialog box.



You can also perform SPECT filtering without using the interactive equation-filtering mode. Disable the "Auto" checkbox. Choose "Image Prefilter" or "Projection Filter" from the upper popup menu. As with Fourier filtering, filters are chosen by one of three methods: 1) select it from the list in the lower popup menu, 2) draw it on the graph, or 3) type in the equation and click on the "|H(f)|" button. The desired filter appears in the graph. To view other slices, adjust the "Slice" control. To reconstruct all slices, click on "Filter".

After making a new filter, you can save it by adding it to the filter popup menu in the lower left.

Note: Butterworth filtering is a common type of filtering. The Butterworth equation (illustrated in the above window) is:

1/sqrt(1+x/cutoff frequency)^order times 2).

In the above example the cutoff frequency is 0.1 cycles per pixel and the order is 7.

7 Clinical Processing

This chapter describes the procedures for using NucLear MAC's clinical software for nuclear medicine image processing.

Most processing can be performed either step-by-step with user control of analysis at each step or by using sequences that are predefined. These two options allow you to program your own method of processing when desired or to use rapidly executed macros ("scripts"). Both approaches are explained. Because the rapidly executed scripts will probably be used most often, they are included in this chapter, while examples of step-by-step methods are presented in "Manual Processing Examples."

While they will not be employed often by experienced users, the step-by-step examples are particularly helpful in gaining familiarity with the various tools and controls.

In certain sections, subsections entitled "Basis" are included. These provide a brief overview of clinical reasoning related to the analysis. These clinical overviews represent the professional opinions of Dr. James Lear based on his review of the literature as well as his personal experience. They are for illustrative purposes only and are not meant to be comprehensive. Scientific Imaging does not warrant their clinical accuracy. Any questions should be directed to Dr. Lear.

7.1 Renal Function by Plasma Samples

7.1.1 Basis

After intravenous administration, hippuran first distributes within the blood and then distributes within the extravascular fluid space. As these processes are occurring, the kidneys are also excreting hippuran. Experiments have shown that the time course of hippuran activity is consistent with its moving back and forth between the two distribution spaces within the body by simple diffusion while it is being removed from the vascular space by active transport in the renal tubules. Thus, the time course of plasma activity follows an approximate double-exponential decline. The overall activity declines more rapidly in patients with good renal function than in patients with poor renal function. By obtaining many plasma samples over 60 minutes, one can fit the data to a two-compartment model and calculate the rate of hippuran renal clearance, which has been termed effective renal plasma flow (ERPF). The "effective" indicates that hippuran is not completely cleared from the plasma as it passes through the kidney.

It was empirically observed that the plasma concentration of hippuran at a single time point (44 minutes) as a fraction of injected dose correlates with the calculated ERPF based on multiple samples (the better the renal function, the lower the plasma concentration). This observation is the basis for a method that generates ERPF values based on the 44-minute plasma hippuran concentration (Tauxe et al., 1982). The technique uses a quadratic equation to compute ERPF from the measured plasma hippuran concentration obtained at 44 minutes.

$ERPF = a + bx + cx^2$

x = I / c, I = injected dose (cps), c = plasma concentration at 44 minutes (cps per liter)

Empirically, the best values of a, b, and c for 44 minutes were found to be:

a = -54

b = 8.47

c = -0.02

While ERPF does strongly affect the plasma concentration of hippuran at 44 minutes, the size of the distribution space within the patient also affects the concentration. As a result, the single sample method has been shown to introduce errors in ERPF values in patients whose distribution spaces differ from those of the patients used to derive the empirical relationship. A method which uses two plasma samples has been developed to reduce these errors (Lear et al., 1989). The method uses the standard two compartment model, but makes certain assumptions about the relative sizes of the distribution spaces to reduce the number of required plasma samples to two.

Standard Model

$dC1 / dt = - [ERPF \times (C1 / V1)] - [k / V1 \times (C1 - C2)]$

$dC2 / dt = k / V2 \times (C1 - C2)$

$Vd = V1 + V2$

C1 = hippuran concentration in distribution space 1

C2 = hippuran concentration in distribution space 2

V1 = volume of distribution space 1

V2 = volume of distribution space 2

k = diffusion constant

Simplifying Assumptions

$V1 / Vd \approx 0.6$ (in normal patients)

$V1 / Vd \approx 0.5$ (in patients with severe renal disease)

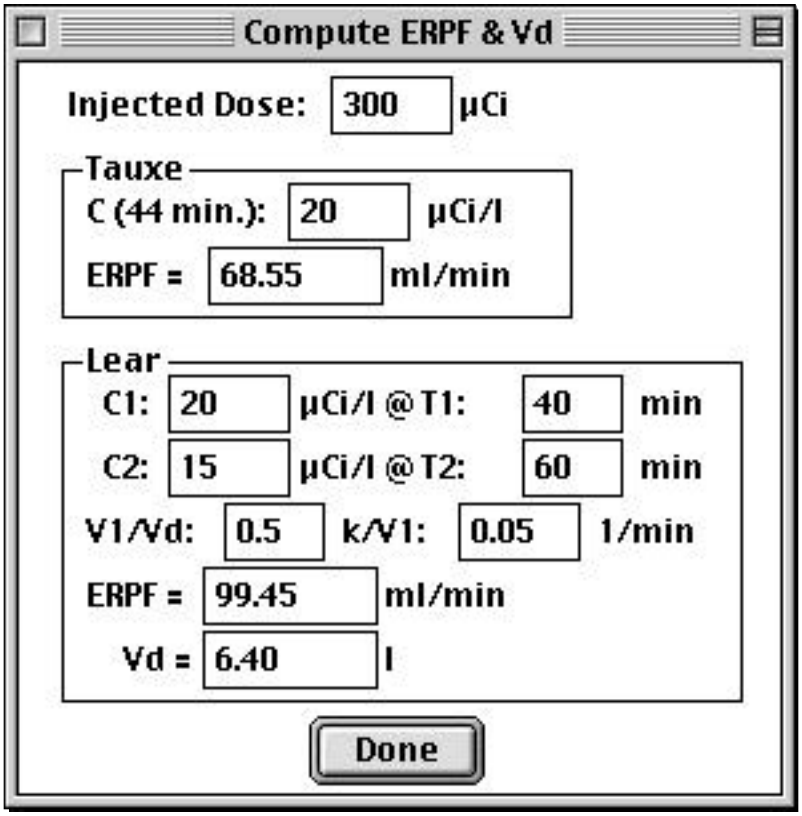
$k / V1 \approx 0.05$

Tauxe W, Dubovsky E, Kidd T, Diaz F, Smith L. New formulas for the calculation of effective renal plasma flow. *Eur J Nucl Med* 1982; 7: 51-54.

Lear J, Feyerabend A, Gregory C. A two compartment, two plasma sample technique for accurate estimation of effective renal plasma flow: theoretical development and comparison with other methods. *Radiology* 1989; 72: 431-436.

7.1.2 Processing

NucLear MAC can compute values based on either the single- or dual-plasma sample techniques. Select **ERPF (Samples)...**, from the **Analyze** menu. A dialog box appears with Tauxe (single sample) and Lear (dual-sample) analysis programs. Type the plasma sample values in the program of your choice. The ERPF values are computed nearly instantly. (Note: Scientific Imaging has implemented these techniques as suggested in the referenced articles but does not warrant their clinical validity.)



Dialog to computer ERPF

7.2 Renal Function by Regions of Interest

7.2.1 Basis

As described under the *Renal Function by Plasma Samples* section, hippuran is cleared from the blood through the kidneys via tubular excretion in proportion to the effective renal plasma flow. Similarly, MAG3 is cleared through the kidneys by excretion, but at approximately 60% the rate of hippuran. DTPA is also cleared through the kidneys, but by glomerular filtration.

In the first few minutes post intravenous administration, all the agents above enter the renal parenchyma but will not have had a chance to exit through the collecting system into the bladder. Thus, the greater the renal function, the greater the amount of activity in the kidneys during this period. Empirical relationships have been developed to estimate ERPF and GFR using the renal activities of hippuran (ERPF) or DTPA (GFR).

Renal activity from one minute acquisitions is compared to the injected activity. The injected activity is obtained from a one minute image of the full syringe held 30 cm from the collimator. For hippuran, the renal activity is measured between one and two minutes, in other words from a 60 second image beginning 60 seconds after administration (Schlegel et al., 1979). For DTPA, the renal activity is measured between two and three minutes, in other words from a 60 second image beginning 120 seconds after administration (Gates, 1982). Quadratic equations are then used to estimate the renal function parameters.

Hippuran:

U = 1-2 minute kidney count (background corrected) / $I * Y^2 * 100$

$ERPF = 1.86\ U - 0.00116\ U^2$

I = syringe counts corrected for residual (1 minute)

Y = kidney depth in centimeters = 13.3x + 0.7 (right kidney) = 13.2 x + 0.7 (left kidney)

x = weight in kg. / height in cm.

DTPA:

$GFR = (\text{right counts} / e^{-0.153\ (13.3\ x\ +0.7)} + \text{left counts} / e^{-0.153\ (13.2\ x\ +0.7)}) / I * 100 * 9.76 - 6.2$

I = syringe counts corrected for residual (1 minute)

x = weight in kg. / height in cm.

Note: Problems occur with these techniques, however, in that it is difficult to exactly correct for tissue attenuation and that distribution space-size also affects the renal activity. Scientific Imaging has implemented them as suggested in the referenced articles, but does not warrant their clinical accuracy.

Schlegel J, Halikiopoulos H, Prima R. Determination of filtration fraction using the gamma scintillation camera. *J Urol* 1979; 116: 447-450.

Gates G. Glomerular filtration rate: estimation from fractional renal accumulation of 99mTc-DTPA (stannous). *AJR* 1982; 138: 565-570.

7.2.2 Processing

Bring up the studies containing the one minute-duration syringe activity and the one minute-duration renal activity.

Notes: If you acquired the syringe image along with the kidney images in the same series, then display the study in all-frames format. Be sure to use the correct image frame, i.e. the second one (60 to 120 seconds) for ERPF and the third one (120 to 180 seconds) for GFR.

Select **Renal Functions...** from the **Analyze** menu. Follow the directions for identifying image frames and drawing ROIs. A dialog box appears allowing you to select the agent, e.g. DTPA. Enter the height and weight of the patient, then click the Compute button. The results are displayed in the Analysis window. These values are based on the above reports from the literature. (Note: Scientific Imaging has implemented these techniques as suggested in the articles but does not warrant their clinical validity.)

7.3 Split Function

Split function refers to the estimation of relative function between the left and right sides of a dual-organ system (e.g. lungs). For the following, use the **Lung** or **Kidney** submenu of the **Regions** menu. This works even if you are analyzing some other organ system. Select the study with the organ system. Draw an ROI around the left organ, then choose **Left** from the submenu. Draw an ROI around the right organ, then choose **Right** from the submenu. The Analysis window displays an activity ratio that is not corrected background activity. If background subtraction is desired, then draw a background region and select **Background** from the submenu. The Analysis window displays the fractional counts corrected for background.

7.4 Parathyroid Imaging

7.4.1 Basis

Parathyroid adenomas are usually imaged by performing thallium (or Cardiolite) and pertechnetate images over the neck with the patient in exactly the same projection for each image. The thallium (or Cardiolite) accumulates in both the thyroid and parathyroid tissue while the pertechnetate accumulates only in the thyroid tissue. Thus,

parathyroid tissue can be imaged by performing a weighted subtraction of the pertechnetate image from the thallium (or Cardiolite) image.

7.4.2 Processing

This processing can be performed very rapidly and intuitively. First, if the images have been obtained as a series, motion correct them and then make composite thallium (or Cardiolite) and pertechnetate images. Activate the composite thallium (or Cardiolite) window. Smooth the image if desired, then select the **Set First Term** selection from the **Analyze>Image Math** submenu. Then activate the composite pertechnetate image, smooth it if desired, and select the **Subtract Weighted** item from the **Analyze>Image Math** submenu. A new window appears containing a difference image. The mouse is then used to control the fraction of subtraction and the image brightness scale. Vertical movement changes the fraction (X) and horizontal movement changes the brightness scale.

New Image = Thallium Image - X * Pertechnetate Image

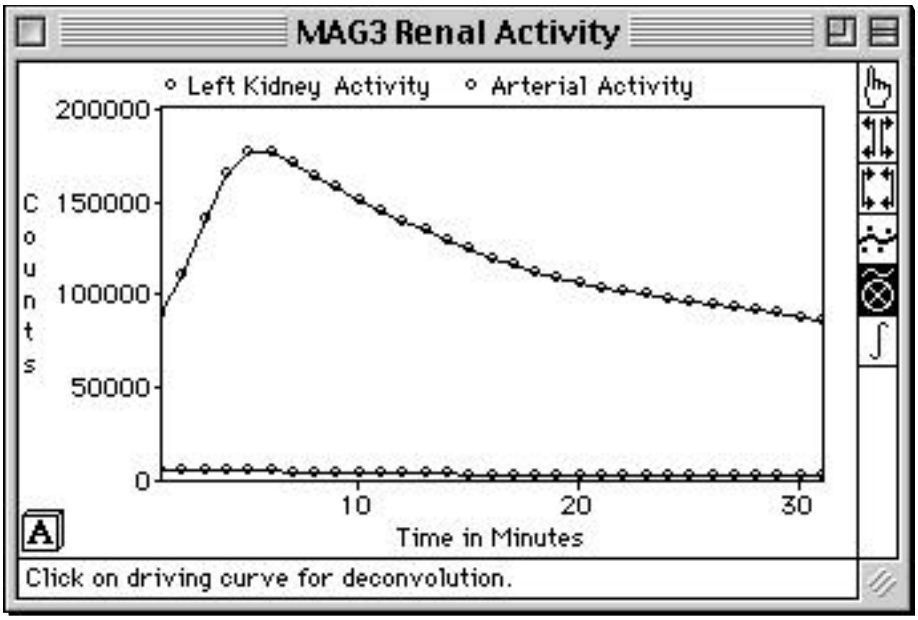
Note: if the cursor gets too close to the edge of the monitor, it can be moved back to the center without affecting the images by holding down the shift key during the movement; release the shift key to return to subtraction mode.

At times, the patient may have moved slightly between the two images. This error can be corrected by using the arrow keys on the keyboard while in the subtraction mode. Pushing an arrow key will move the pertechnetate image by one pixel in the direction of the arrow and will automatically change the difference image to reflect the movement. By observing the edges caused by misalignment on the difference image, you can align the images properly.

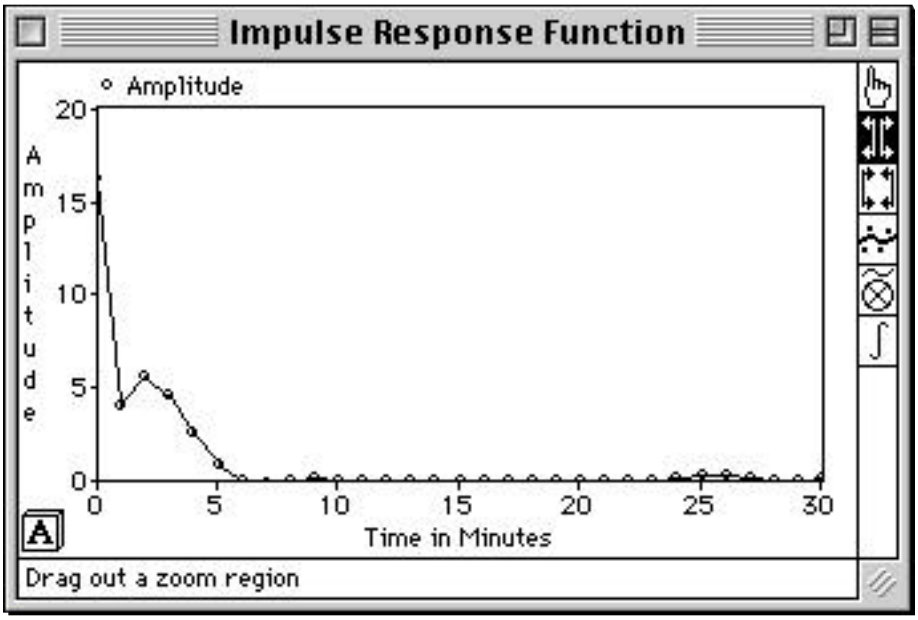
7.5 Transit Times Computed by Curve Deconvolution

Curves can be deconvolved to generate input-response functions. Typically an arterial input drives activity in an organ. Time-activity curves from the organ and artery are deconvolved to produce the response function of the system, which in turn leads to the transit time of the system.

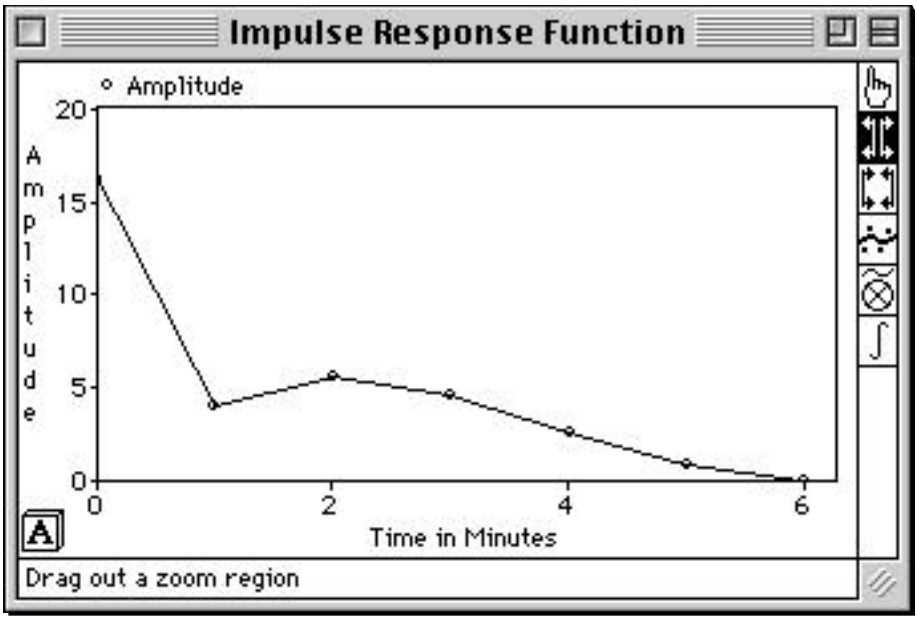
If necessary, reframe the target study so that every frame has the same duration (acquisition time). Substantial temporal smoothing of the study may be necessary to achieve good results. One source recommends three applications of a (1 2 1) smooth. This is equivalent to one application of a (1 6 15 20 15 6 1) smooth. Generate the arterial and organ time-activity curves by drawing the two ROIs, selecting them both, and using the **Regions>Frames Plot** menu command. Click on the tool icon that looks like a circle with an X.



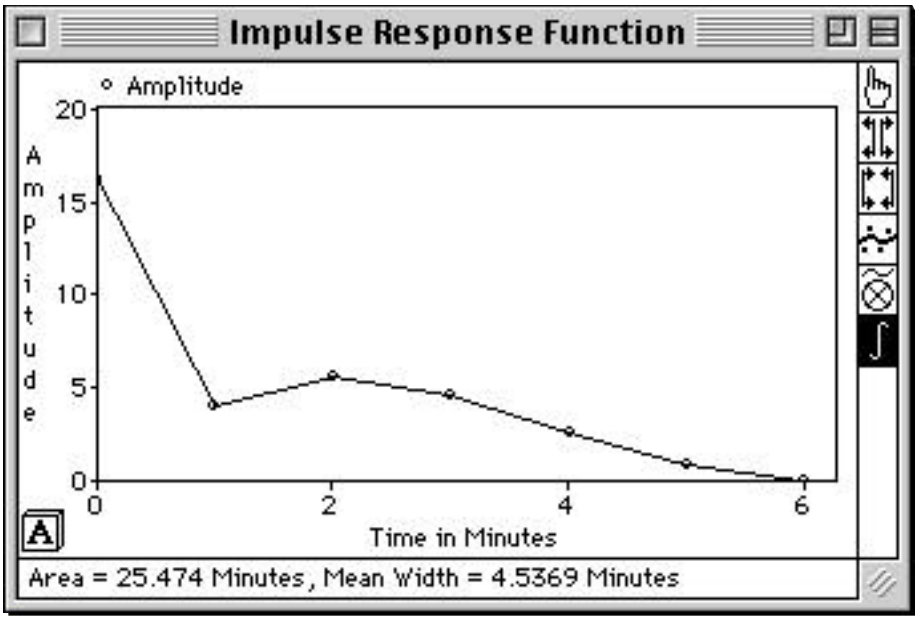
Follow the instructions that appear at the bottom of the graph window. Note: the driving curve is the vascular curve and the response curve is the tissue curve. This process creates a new graph window containing the response function.



If you wish to compute the tissue transit time, then drag out the portion of interest with the expand tool. The portion of interest will usually be from the origin to where the curve drops to near zero (from time zero through six minutes in the above example).



Next, select the "integrate" tool. Then click on the point where the drop-off begins (at two minutes in the example above) and the mean width and integral are displayed.



7.6 First Pass Examinations of the Right Ventricle

7.6.1 Basis

A first pass examination of the right ventricle (RV) consists of a series of images obtained in the right anterior oblique projection as the bolus of activity passes through the RV. These images are obtained rapidly so that each cardiac cycle is spread over several frames. For example, a common acquisition consists of 400 frames obtained at 20 frames per second (50 milliseconds each). Thus, each cycle will have approximately 10 to 20 frames.

As the bolus passes through the right ventricle, the resulting overall time-activity curve rises and then falls. Superimposed upon this profile are the ups and downs of diastole and systole. The principle behind first pass ejection-fraction calculation is that an estimate can be gained by computing the average fraction ejected from each of several beats (two to six) as the bolus passes through. In other words, using a composite image, the user draws a region around the right ventricle from which the computer generates a time-activity curve. The user picks the diastolic and systolic points for each beat from the curve using the “Large circle tool” and “Small circle tool” respectively, the computer calculates the ejection fraction for each beat, and then the computer calculates the average. This method is often referred to as the fixed-region method.

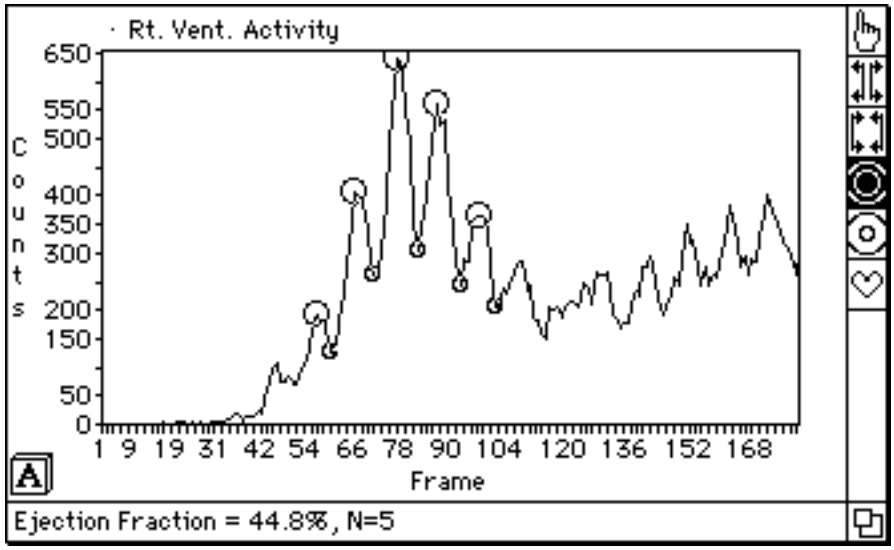
A problem with this approach can occur when the tricuspid valve plane moves significantly during systole. Then, the region for the RV based on the composite image may not be correct for the systolic frames, as activity from the right atrium can be included. This phenomenon causes slight elevation of systolic counts and slight underestimation of ejection fraction (diastole - systole / diastole). This problem can be avoided by using different regions for the RV in diastole and systole. (Note that NucLear MAC supports both methods so that the user can choose.) In typical patients, the EF by variable region analysis will be approximately 5% greater (in absolute terms) than that obtained using fixed region analysis (e.g. 45% compared to 40%).

7.6.2 Processing

Bring up the study window containing the first pass images. Select **First Pass…** from the **Analyze** menu. Follow the directions on the screen, which will lead you through the processing steps.

During the processing, the activity vs. time curve is generated and you are asked to select diastolic and systolic point-pairs on the curve. Use the large circle tool and small circle tool as in the example below.

Note: if you misplace a circle, then just click on it again to remove it.



After the user places the pairs of circles for diastole and systole, the ejection fraction by fixed region of interest appears at the bottom of the graph window. To compute the EF by variable region of interest analysis, click on the heart-shaped tool on the right side of graph window. The program sums the systolic and diastolic frames and tells the user to draw the RV in the summed images. The EF is then be displayed in the Analysis window.

7.7 Cardiac Shunt Calculations

7.7.1 Basis

If a bolus of activity is injected intravenously, it passes through the right atrium, the right ventricle, the lungs, the left atrium, the left ventricle, and finally the systemic circulation. Thus, a time-activity curve obtained from the lungs rises to a peak and then falls as the bolus passes through.

If there is a left-to-right shunt, then a second peak occurs in the lung activity curve about four to seven seconds after the first, because the shunted blood enters the right ventricle and then passes through the lungs a second time. The amount of the shunt as fraction of the total ventricular output is proportional to the amount of activity in the second peak compared to that in the first. The problem in performing this calculation is that the first bolus curve does not reach zero by the time the shunt activity is seen. Thus, if the areas under the two peaks or the heights of the two peaks from the total curve are measured directly and used to compute the shunt ratio, then the calculated shunt ratio is in error.

This problem is reduced by using gamma variant analysis. Gamma variants are types of equations that generate curves that rise and then fall like bolus time-activity curves. A gamma variant equation is fit to the original activity curve from the beginning of the rise to just beyond the first peak. A curve is then generated which simulates the normal fall in activity beyond the peak. This simulated curve is subtracted from the overall time - activity curve. The remaining curve is assumed to represent the shunt curve. Another gamma variant is then fit from the beginning of its rise to just beyond its peak. The ratio of the heights of the two gamma variant curves or the areas under them are then used to compute the shunt ratio.

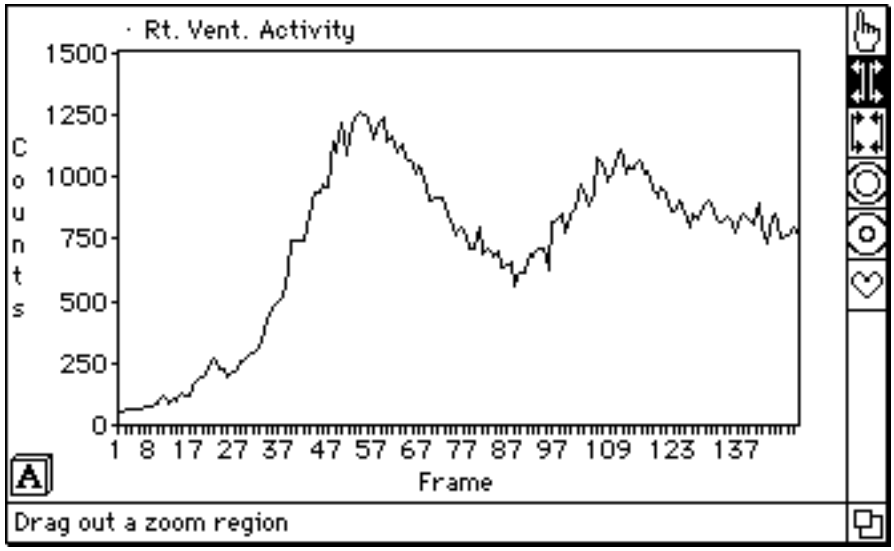
Note: First pass shunt study data is acquired slightly differently from first pass data for right ventricular performance evaluation. Usually shunt study data is acquired in the anterior projection without zoom on a LFOV camera. Framing is usually about five per second (200 milliseconds) for a total of 30 seconds (150 frames).

7.7.2 Processing

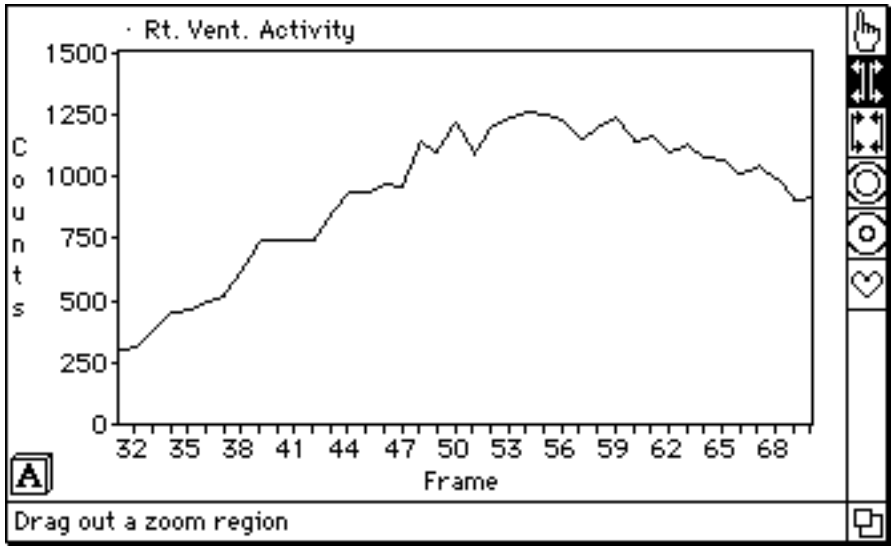
Bring up the window containing the frames from the shunt study. Select the **Shunt...** item from the **Analyze** menu and follow the directions. The program calculates the shunt ratio based upon both the height and area methods.

The program initially shows a composite sum of about half of the frames. The user can then change the number of summed frames with a scroll bar to get a good representation of the lung activity. The user draws a region within the lungs (usually the

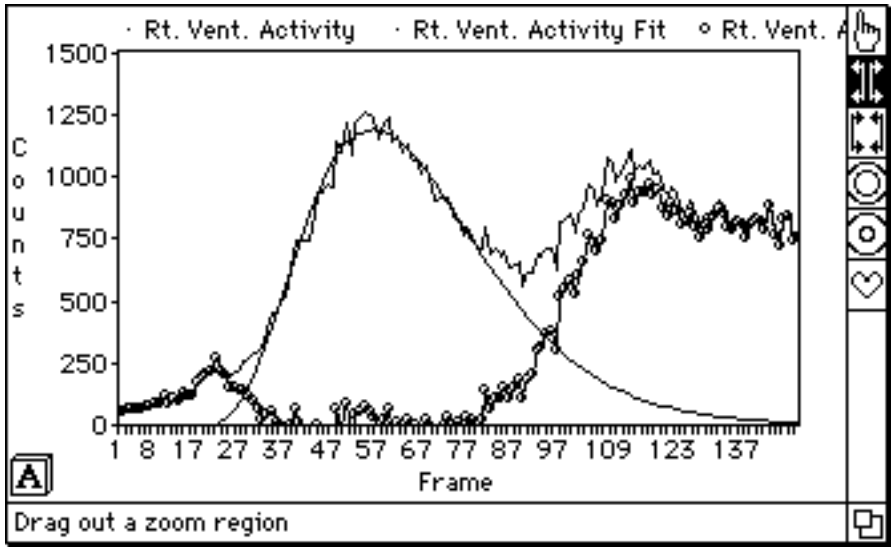
right) with care to avoid vascular (heart and great vessels) activity. If desired, a background region can be drawn, but this is usually not needed. The computer responds with an activity vs. time curve.



The curve expand tool is highlighted and the user is directed to drag out the first peak. Start where the curve begins to rise rapidly (around frame 32 in the above example) and end just before the rapid decline rolls off (around frame 70 in the above example). The computer shows the highlighted region of the curve, and the user can adjust it again if necessary to make it smaller by dragging out the desired portion.

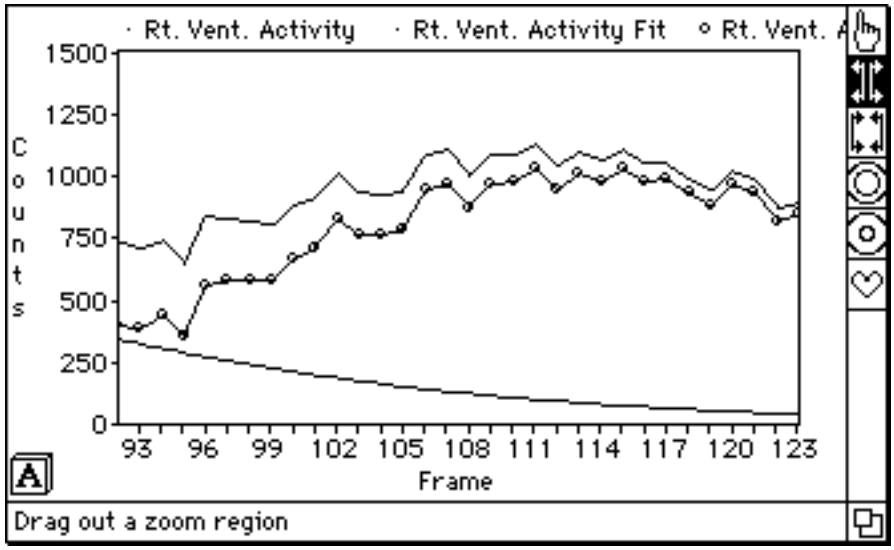


The computer then performs a gamma variant fit and subtracts it from the total activity curve.

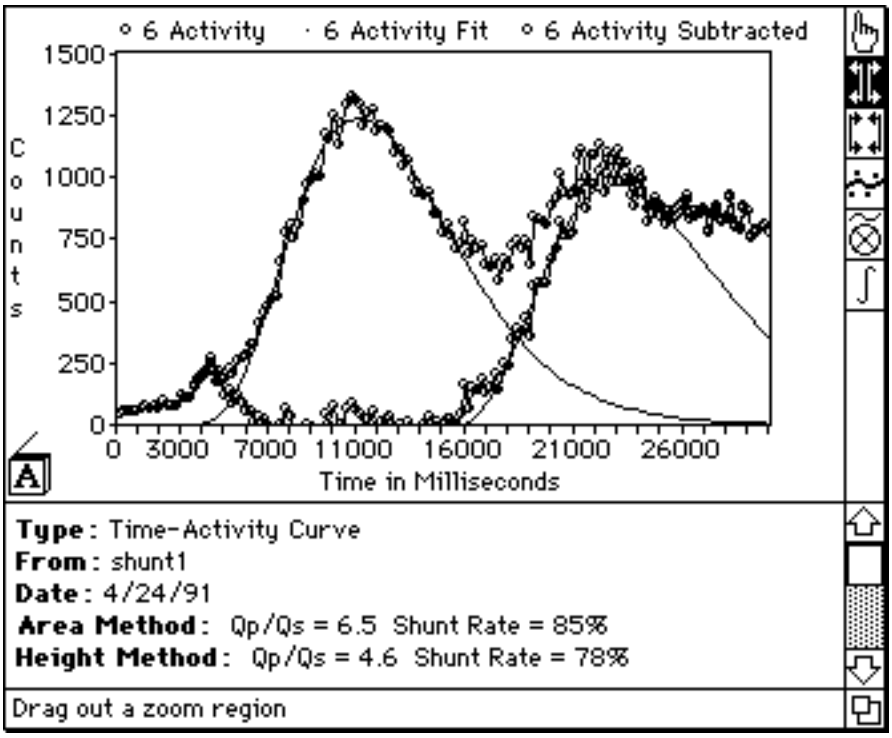


The computer directs the user to drag out the second peak based on the curve derived from the subtraction (the one with the circular points). Again start at the rise (frame 90 in the example above) and end before the decline rolls off (frame 122 in the example).

The computer shows the selected portion of the curve, which can be adjusted if needed as with the first curve.



The computer then performs the second gamma variant fit and computes the Qp / Qs ratios based upon the relative heights and areas of the gamma variants.



7.8 Gated Examination of the Left Ventricle

7.8.1 Basis

Gated studies of the heart consist of a set of frames representing the heart at various times during the contraction cycle, i.e. between R waves. The frames are composite images based on data acquired from many cardiac cycles.

Data from gated cardiac blood pool examinations can be processed using frame-mode or list-mode techniques to generate cinegrams reflecting cardiac wall motion. In the more commonly used frame-mode processing, the median cardiac cycle length (L) of each patient is first determined and divided into some number (N) of discreet time-frames, usually 12 through 32. Then, beginning with the next R wave, a series of N images (pieces) is obtained, each having a duration of L / N. These sequential pieces are stored into N respective frames in the computer's random access memory. Each successive R wave recycles the acquisition process and data from each successive cardiac cycle's pieces are added to their corresponding frames. In this manner, a series of frames is generated over time which represents the sum of R-wave synchronized pieces from many cardiac cycles.

Frame mode acquisition is convenient because the aggregate image frames are available for viewing and analysis immediately upon completion of data acquisition. However, problems often arise because of deviations in cardiac-cycle lengths away from the median value. For example, if a particular cardiac cycle is shorter than the median value (but not so short as to be rejected), then a new cycle of data acquisition begins before the old cycle is finished. The frames near the end of the cardiac cycle therefore tend to have fewer counts than the frames near the beginning since pieces from some cycles are missing. This phenomenon causes the cinegrams to have "dim"

frames near the end and can cause artifacts in quantitative analyses based on time-activity curves.

List-mode processing was proposed to avoid such frame-depletion problems. Rather than being framed directly into one of the N images in computer memory, position data from each detected gamma ray is written to the computer's disk drive and stored in a long list. Inserted into the list of event positions are timing marks, usually every 1/100 of a second, and R wave indicators. The list of individual points is processed after the acquisition is over to generate the composite image frames.

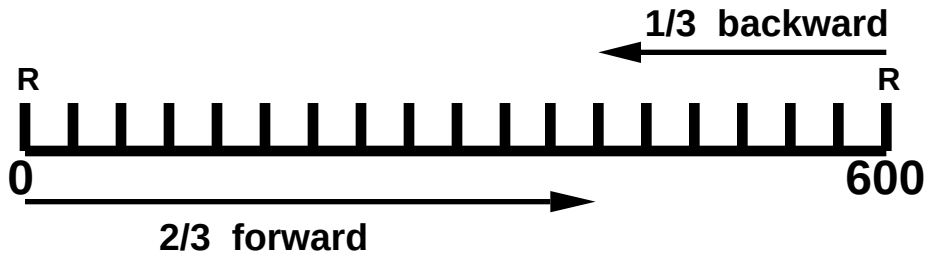
Various methods for organizing the event list into image frames have been suggested and each method includes two specifications: 1) whether the duration of each piece is fixed or variable and 2) whether the framing of the pieces is forward, backward, or forward-backward with respect to the R wave markers. For example, fixed interval, forward framing simulates standard frame-mode acquisition. Data from each cardiac cycle is organized into pieces having a fixed duration equal to L / N and the pieces are added to the composite frames in a forward sequence from the R wave that starts the cycle. Fixed interval, backward framing indicates that fixed duration pieces from each cardiac cycle are added to the composite frames in a backward sequence beginning with the R wave that ends the cycle.

The variable-interval, forward-framing type of list-mode processing was first used to eliminate the dim end frames. Data from each cardiac cycle is framed into N pieces, each having a duration equal to the individual cardiac cycle's length (rather than the median length of all cycles) divided by N. Thus, each cycle's image data is uniformly distributed into the N pieces because the duration of each piece varies with the cycle's length. While this method does generate cinegrams with uniform frames, it is physiologically inaccurate because cardiac cycles are not uniformly stretched or shortened in proportion to changes in their duration. Rather, the time from the beginning to the end of systole tends to remain fixed while the time between end-systole and end-diastole varies.

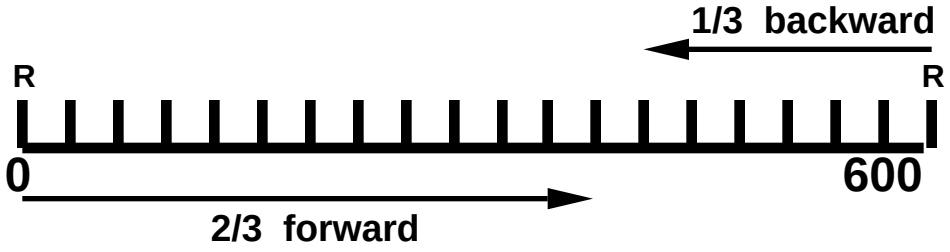
Fixed-interval, 2/3 forward - 1/3 backward processing has been shown to be the most physiologically accurate method of framing list-mode data uniformly (Bacharach et al., 1979).

Data from each cycle is organized into pieces each having a duration equal to L / N. However, the individual cycles' pieces are added to the composite frame data using the R waves at both the beginning and end of the cycle for synchronization. N x 2/3 pieces are framed forward from the R wave which begins the cycle and N x 1/3 pieces are framed backwards from the R wave that ends the cycle. Thus, if a particular cycle has a length equal to the median value, all of its data will be exactly sorted into the N pieces. If a cycle is longer than the median value, counts occurring between the N x 2/3 forward portion and the N x 1/3 backward portion are ignored. A short beat causes data points near the junction of the forward and backward components to be used twice, once in each component. Because the physiologically important portions of the cardiac cycle (end diastole, peak ventricular emptying time, end systole, peak ventricular filling time) all occur within the first 2/3 of the cardiac cycle, no significant errors occur because of the data over- or undersampling at the junction point. No dim frames occur

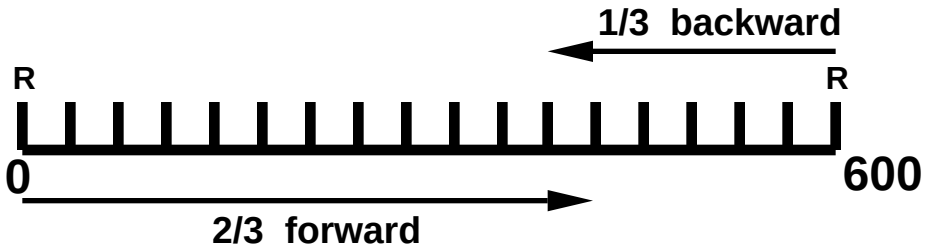
in the cinegram because the end of the cycle always is always contained within the final 1/3 of the pieces.



Median Length Beat

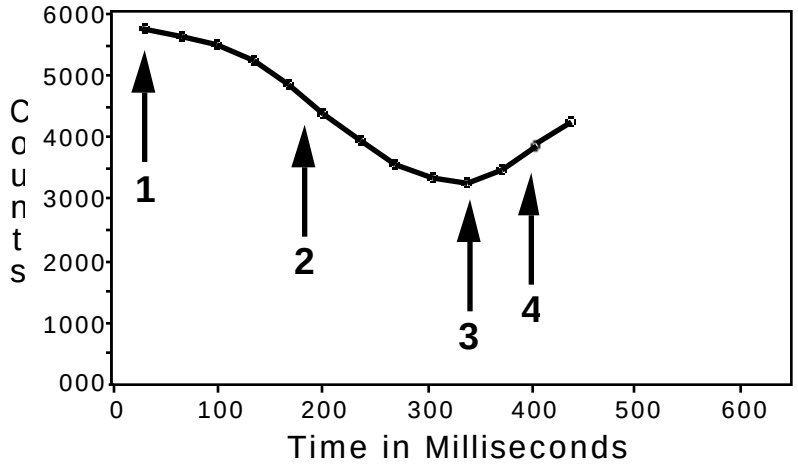


Long Beat

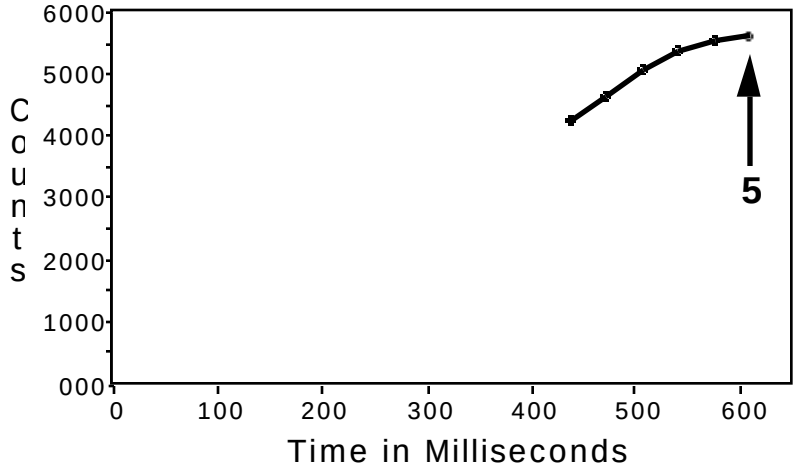


Short Beat

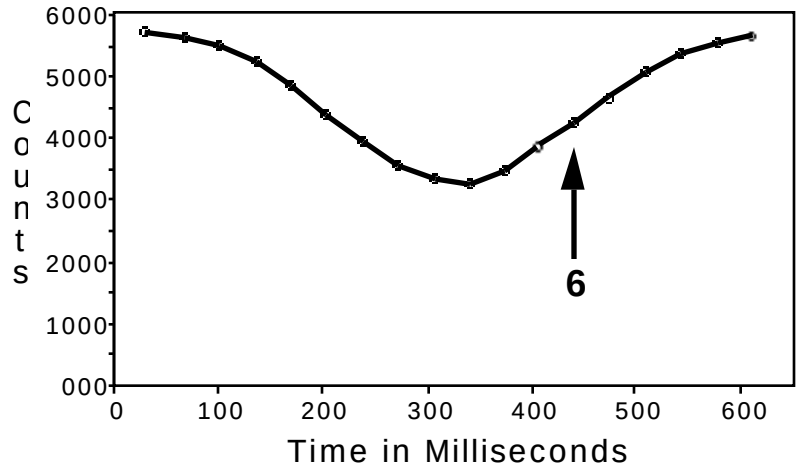
Figure above: Illustration of principle of data framing used in forward-backward list mode processing. Data is framed forward from an R wave for a duration equal to 2/3 of the median cardiac cycle length. Data is framed backwards from the next R wave for a duration equal to 1/3 of the median cardiac cycle length. The two data segments are then combined. Beats which are slightly shorter than or longer than the median cycle length have slight over- or undersampling at the junction point, but these errors are not clinically significant as shown below.



2/3 forward



1/3 backward



Composite

Illustration of generation of composite activity vs. time curve from the forward-framed and backward-framed data

The physiologically important points in the cardiac cycle (1 = end-diastole, 2 = peak emptying time, 3 = end-systole, 4 = peak filling time) all occur within the initial 2/3 of the cardiac cycle, even in patients with prolonged systole, as in this example. The last frame of the cinegram (5 = last frame in cinegram) always occurs in the final 1/3. Heart rate variations can cause small artifacts at the junction point (6 = junction point), but these are not clinically significant. Unfortunately, computational problems have impaired the widespread use of fixed-interval, 2/3 forward - 1/3 reverse processing, despite its physiological advantages. (In fact, some recent reviews have even recommended a return to the use of simple frame-mode processing because of its much greater ease-of-use compared to list-mode processing.)

NucLear MAC solves the above problems by performing real-time list-mode processing during acquisition. In other words, the benefits of list-mode processing are gained without the need for tedious post-processing. The system operates by processing the data from a given beat at the same time data is being acquired for the subsequent beat (Lear and Pratt, 1992). This approach allows a user to choose the method of beat framing without losing the convenience of standard frame-mode acquisition.

The type of framing is controlled in the Acquire Gated Images dialog box. The following settings (in the Beat Range part of the dialog) represent commonly used approaches.

Standard frame mode processing: Fixed bins, 99% at start

Variable interval processing: Variable bins, % at start does not matter

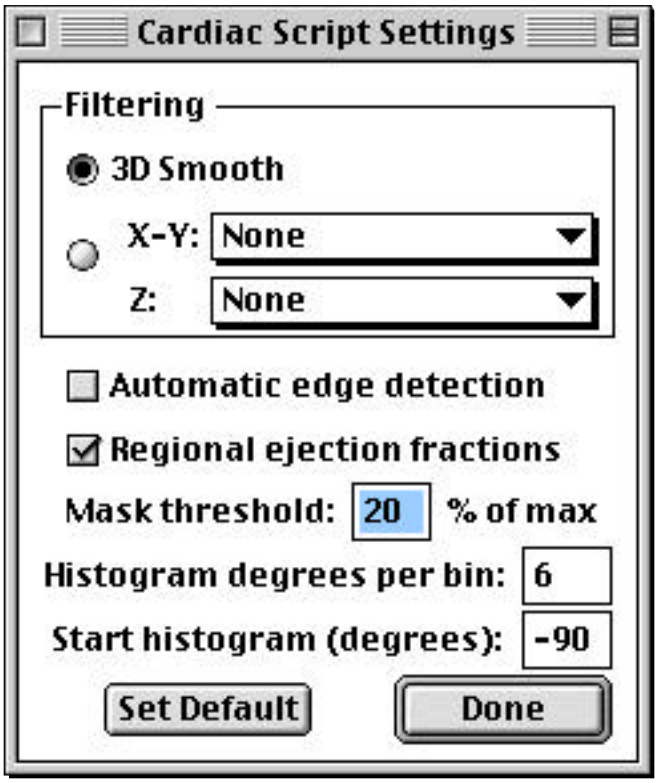
Forward - backward framing (NIH method): Fixed bins, 67% at start

Bacharach S, Green M, Borer J. Instrumentation and data processing in cardiovascular nuclear medicine: evaluation of ventricular function. *Semin Nucl Med* 1979; 9: 257-274

Lear J, Pratt J. Real-time list-mode processing of gated cardiac examinations with forward-backward framing. *Eur J Nucl Med* 1992; 19: 177-180, 1992

7.8.2 Processing

Select **Settings...** from the **Analyze>Gated Cardiac** submenu to set the parameters for processing the gated data.



Choose the type of filtering for the raw cinegram data. 3D smooth applies a temporal and spatial unweighted smooth automatically or temporal (Z) and spatial (X-Y) smoothing can be chosen independently. Activate the "Automatic edge detection" if desired. Activate the "Regional ejection fractions" if desired. The "Mask threshold" value is used in the phase analysis and functional images to zero out background below a fraction of the maximum value. "Degrees per bin" and "Start histogram" values are used in the phase histogram image. Click the "Set Default" button to save the choices permanently.

Bring up the study window containing the gated images. Select the **Run Script...** item from the **Analyze>Gated Cardiac** submenu and follow the directions. The program computes the ejection fraction and displays it in the Analysis window. The program generates a graph window with the background-subtracted, LV activity versus time data (using the diastolic region of interest in the non-edge detect mode and the individual regions of interest in the edge detect mode). From this data, the program computes and displays in the Analysis window the peak filling and peak emptying rates and the time to peak filling. The program also generates a study window containing 9 frames: smoothed end diastole, smoothed end systole, amplitude, phase, phase histogram, peak filling rate, peak emptying rate, time to peak filling, and regional ejection fractions.

7.9 Planar Thallium Scan Quantitative Analysis

7.9.1 Basis

Thallium ion is a potassium ion analog. It therefore accumulates rapidly in muscle (and other) cells and normal cells maintain a high intracellular / extracellular ratio.

Following intravenous administration, about 10% of the thallium accumulates in the lungs and the remainder enters the systemic circulation.

Approximately 5% of the cardiac output passes through the coronary arteries and most (>80%) of the thallium in the arterial blood is extracted by the myocardium as the initial bolus passes through. Thus the initial concentration of thallium in normal myocardium represents the fraction of the cardiac output that passes through the region. In other words, the initial thallium concentration (Ci) is proportional to perfusion.

If the thallium is injected during exercise, and if the patient has coronary artery disease, then regions of the myocardium supplied by the stenotic vessels may have lower perfusion than regions supplied by normal vessels. The stenotic vessel-supplied regions therefore have lower initial concentrations of thallium.

Regions of infarctions have smaller cell populations and therefore also have lower initial thallium concentrations.

Over the next four hours, the blood concentration of thallium decreases by about 50% compared to levels at 10 minutes as the thallium slowly enters various other distribution spaces in the body. The myocardial thallium concentration tracks the blood concentration downward and the myocardium and at four hours blood are practically in chemical equilibrium. In other words, the concentration of thallium remaining in the myocardium at the delayed time (Cd) is proportional to the myocardial mass.

Thallium concentration in areas of myocardium which were normally perfused during exercise in patients who achieved high levels of exercise reach a level at four hours of about half the initial levels. In other words, they lose about half of their initial thallium. Thallium concentration in areas of myocardium which were under-perfused during exercise because of being supplied by stenotic vessels drop to approximately the same level as that of the normally perfused myocardium. However, since they had lower concentrations to start with, they lose a smaller fraction of their original activity. Regions with infarction lose activity in approximate proportion to the normal areas.

Qualitatively, areas of ischemia therefore appear relatively more abnormal (low) on the stress images than on the delayed images. This appearance is known as redistribution. On the other hand, areas of infarction are abnormal on both stress and delayed images.

These qualitative patterns can be used to differentiate normal versus ischemic versus infarcted tissue in most patients by visual inspection. Also, various image manipulations can be performed to help compare the stress and delayed image patterns.

Relative quantification: A common method is to normalize the overall brightness of the delayed images to match that of the stress images. Ischemic areas thus appear brighter

on the delayed images than on the stress images. By aligning the stress and normalized delayed images, percent difference images can be generated (normalized delayed minus stress divided by normalized delayed). Ischemic areas are therefore brighter than normal areas on the percent difference images. This approach is known as relative quantification because the percent difference images are generated from the relative differences between stress and delayed images. Various groups have proposed various thresholds to differentiate ischemic tissue from normal tissue using percent difference images. For example, areas whose percent difference exceeds 20% are often considered ischemic.

Absolute quantification: However, some researchers have pointed out that if a patient has balanced triple vessel disease, then the entire heart may be underperfused evenly during exercise so visual inspection or percent difference images may not reveal ischemia. They have proposed using the basic thallium concentration relationships to aid in diagnosis, i.e.:

Normally perfused tissue: Ci, normal; Cd = normal

Hypoperfused (ischemic) tissue: Ci = low; Cd = normal

Infarcted tissue: Ci = low; Cd = low

They use a parameter known as "washout" for this purpose where washout is defined as stress activity minus delayed activity divided by stress activity, i.e. (Ci - Cd) / Ci. Normally perfused tissue in well-exercised patients has approximately 50% washout while ischemic tissue has lower washout. This approach is known as absolute quantification because quantitative washout images are generated from the absolute differences between stress and delayed images. Absolute thresholds are often used to diagnose ischemia, e.g. washout less than 25% represents ischemia.

While this quantitative washout analysis is commonly used, problems occur. For example, if the patient does not exercise well, then myocardial blood flow during exercise may not increase much above resting levels. Overall initial myocardial thallium concentrations are therefore lower than those of well-exercised patients. Since delayed concentrations are not affected by the exercise level, overall washout is lower in the patients with suboptimal exercise. Washout-based programs may then erroneously indicate that certain regions or even the entire heart are ischemic.

Background subtraction: Both relative and absolute quantification approaches share some initial processing requirements. First, background must be subtracted from the images. The original method developed for this correction involved drawing a region adjacent to the left ventricle on the computer images and calculating the average background per pixel. This value was then subtracted from all of the pixels in the image. However, it was pointed out that background may be greater than average in some regions of the image, e.g. near the liver. Interpolative or linear subtraction was developed to reduce this problem. A box or circle was drawn in the background around the left ventricle and the background was measured at each point in the region's perimeter. The background over each pixel in the left ventricle was determined by linearly interpolating the background measurements from orthogonal perimeter points. Quadratic, as opposed to linear, interpolation has also been used.

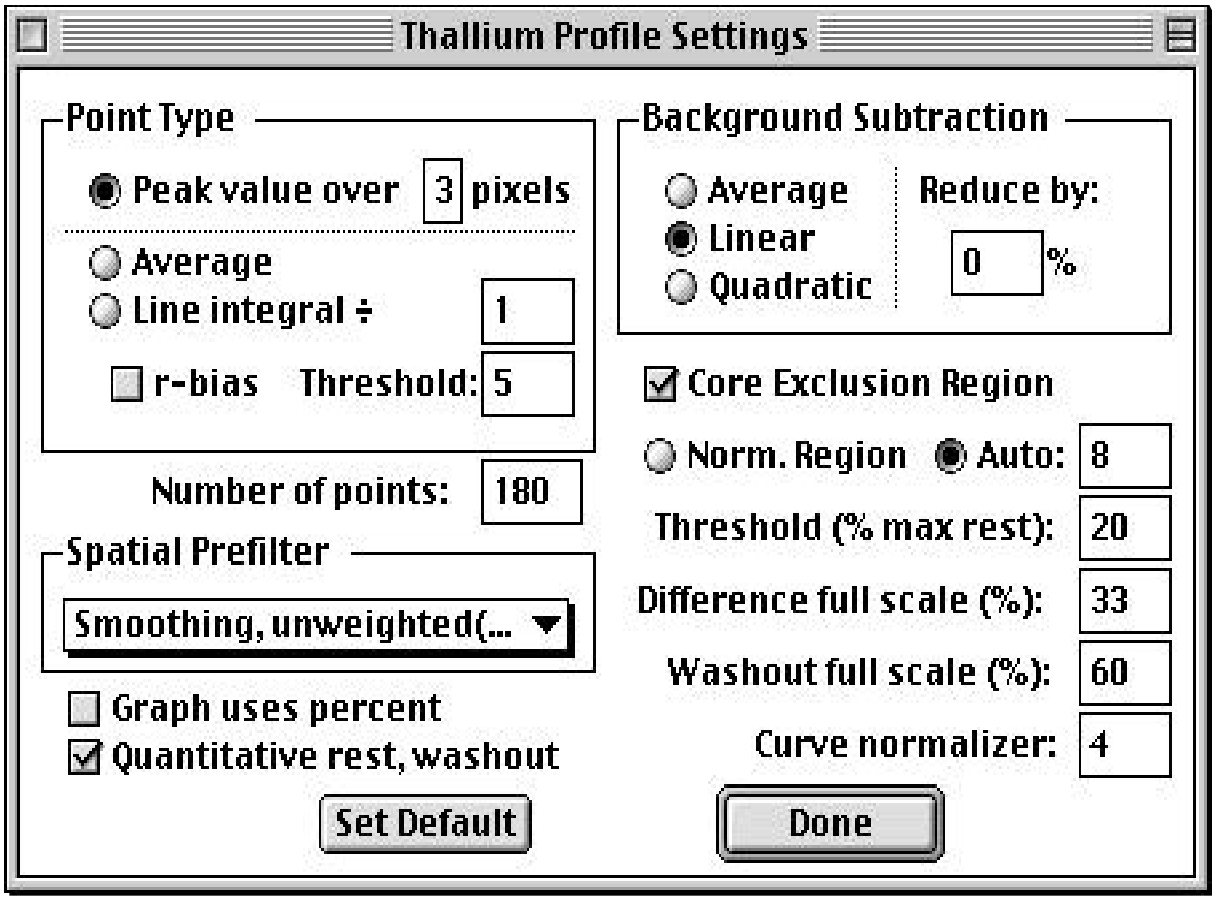
A problem with this background subtraction is that background is overestimated. Most background derives from the lungs and the lungs are thinner in the heart-region of the thorax. Thus background overlying the heart is lower than in the lung fields where the background regions are drawn. Some researchers have therefore proposed that the background that is subtracted from the heart be reduced by 25% to 50% from values calculated using interpolation. While this method probably yields more accurate washout measurements, it has not gained widespread use.

Circumferential profiles: After background subtraction is completed, some method is needed to compare the thallium activities in the stress and delayed (or normalized delayed) images. Both relative and absolute quantification approaches usually use circumferential profiles. Radii are drawn from the center of the LV out to the edge of the LV every few degrees and the amount of thallium along the radii is plotted as a function of the angle. The “amount” of thallium may be the greatest pixel value along each radian, the average pixel value along each radian, or the sum of the pixel values along each radian. Each approach has advantages and disadvantages which are argued vigorously between various researchers.

NucLear MAC allows the user to choose the type of analysis with which he is most comfortable.

7.9.2 Processing

First, define the type of analysis you prefer using the **Settings...** item of the **Analyze>Planar Thallium** submenu. Use the Set Default button to save the choices.



Choices include:

"Point type": Choose between peak value, average value, or line integral for the method of circumferential plotting. Peak value is the least subject to background artifacts but does not evaluate the entire width of the heart wall. The number of contiguous pixels used to determine the peak value is set in the box between "over" and "pixels". The program finds the maximum sum of that number of contiguous pixels along each radian. Therefore, the number of pixels is greater than one, the peak value reflects activity in user-determined widths of myocardium. Average or line integral methods better evaluate the entire heart wall, but can be affected by residual background. For example, if background is included in the average calculation, then the value is lowered artifactually. If different amounts of background are included in different parts of the profile, then the shape of the profile is affected. The "Threshold" value is used to help reduce background effects. Pixels with values below the threshold are not used in computing the average or line integral along the radii. "r-bias" weights the average or line integrals towards the periphery of the heart so that the myocardial wall as opposed to the central chamber is emphasized.

"Number of points" defines the number of radii over the entire 360 degrees.

"Spatial prefilter" is used to choose the type if filtering to be applied to the original images.

The units of the activity in the circumferential profile is counts unless the "Graph uses percent" box is checked in which case the units are percent of maximum stress counts.

If the "Quantitative rest, washout" box is checked, then the graph includes the washout profile.

“Background subtraction” can be average, linear, or quadratic. Linear is analogous to the commonly used interpolative method. A lower value of background than that calculated by interpolation can be selected with the “Reduce by” value. (“0” means interpolated value.)

“Core exclusion region” is used to allow the user to draw a region representing the heart chamber and outflow region. This region is eliminated when the percent difference and washout images are generated.

“Norm. region” allows the user to select the region in the stress image that is used to normalize the delayed image.

“Auto” defines a square region with dimensions equal to the value in the box in the stress image to be used to normalize the delayed image. (The computer finds the square with the highest pixel sum in the stress image.)

“Threshold (% max rest)” is used to define the part of the image that is used to generate the percent difference and washout images. Only those pixels with greater values than the threshold percent (based on the rest image) are displayed. Other pixels are zeroed out.

“Difference full scale (%)” sets the relationship between the pixel values (0 to 255) in the percent difference images and the value of the percent difference between the normalized delayed and stress images. For example, a value of 40% means that a pixel value in the percent difference image with a value of 255 equals 40% percent difference, a value of 128 equals 20%, etc. Note: percent difference = 100 x (normalized rest - stress) / normalized rest.

“Washout full scale (%)” sets the relationship between the pixel values (0 to 255) in the washout images and the quantitative value of the washout. For example, a value of 60% means that a pixel value in the washout image with a value of 255 (maximum screen brightness) equals 60% percent washout, a value of 128 equals 30%, etc. Note: percent washout = 100 x (stress - rest) / stress.

“Curve normalizer” value is used to specify how the delayed image curve is scaled to the stress image curve. The basic principle is to find the hottest value in the stress curve and then to find the same location in the delayed curve and scale it to the stress value. However, if only one point is used, then noise can cause scaling problems. Thus it is useful to find the hottest average of a small number of points in the stress curve, e.g. 5.

After defining the settings, bring up the windows containing the stress and delayed images. If necessary change the active frames so that the same view is seen for each time, e.g. the 45-degree LAO view. Using the scale bar in the Controls window, set the brightness of the images as desired (the brightness is maintained during the processing).

Select the **Run Script...** item from the **Analyze>Planar Thallium** submenu and follow the directions.

7.10 Bull's-eye Plotting

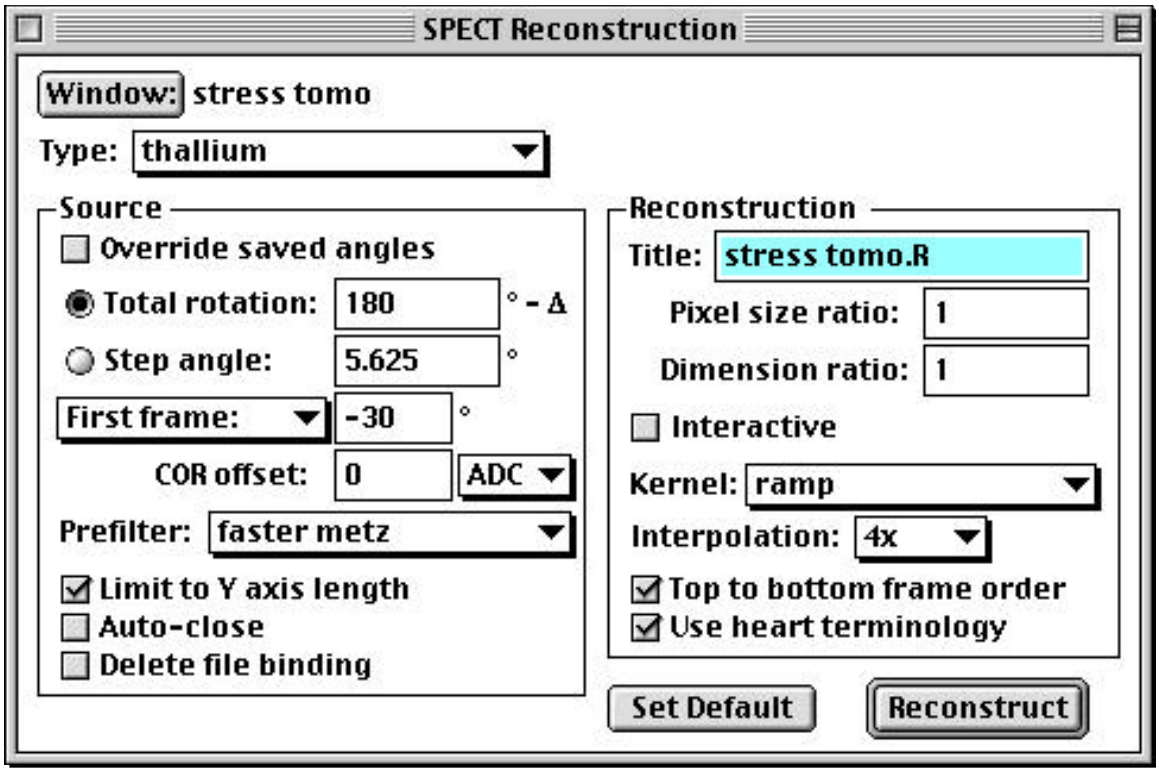
7.10.1 Basis

SPECT thallium scans are often viewed using bull's-eye plots. These plots project data from short axis slices of the left ventricle into an intuitive map in which the base is represented by the outer rings of the bull's-eye and the apex is represented as the inner rings.

Each ring on the bull's-eye represents the circumferential profile for a given short axis slice; the brightness in each part of the ring reflects the numeric value from the profile. Thus, as in planar thallium analysis, radial profiles of peak values, averages, or line integrals can be used to generate the bull's-eyes. The NucLear MAC lets the user choose among peak value plots, average value plots, and line integral plots to generate the bull's-eyes. The computer uses the data from the center of each image to the edge as defined by the user-generated boundary rings. The same relative advantages and disadvantages apply to the various approaches as described in the Planar Thallium section of the manual.

7.10.2 Processing

First, define the SPECT analysis settings for thallium scans by activating **SPECT...** from the **Analyze** menu. Note: when making these settings, have a window containing raw SPECT data open on the screen. Give the processing settings a name by adding it to the **Type:** popup menu. Different settings can be given different names. These names will later be called by the Bull's-eye Plotting.



Choose an appropriate prefilter from the Prefilter popup menu.

Choose an appropriate backprojection filter from the Kernel popup menu. This should be a ramp.

Turn on the “Limit to Y axis length” box if desired to limit the transverse slices that will be reconstructed to only those within the limits of the Y arm of the “Axis” tool.

If during the reconstruction process you wish to have the raw data file closed after it has been filtered (to save memory), then turn on the “Auto close” box.

Check the “Use heart terminology” box if desired.

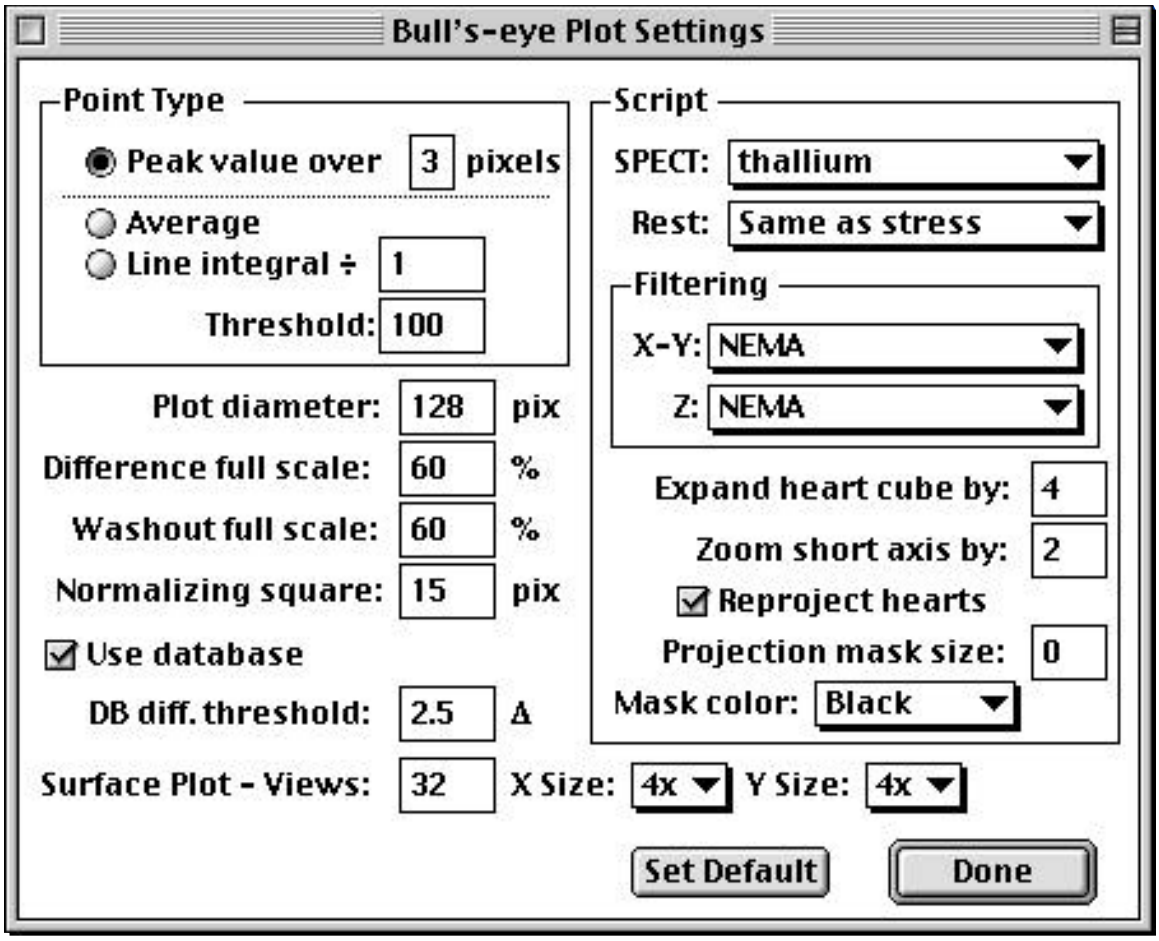
Finally, name the settings by adding the name to the **Type:** menu. This name will later be chosen when you edit the Bull’s-eye plot settings.

Hit “Set Default” if desired.

The SPECT reconstruction dialog box can then be closed.

Next, set the bull’s-eye parameters under the **Settings…** selection of the **Analyze>Bull’s-eye Plotting** submenu.

(The example below contains useful settings.)



"Point type" is used to choose between radial plots of the peak value, average, or line integrals. See the section **Thallium Scan Quantitative Analysis** for more details.

"Plot diameter" is used to set the size of the bull's-eye.

"Difference full scale:" sets the scale for the percent difference bull's-eye; percent difference = 100 x (normalized rest - stress) / normalized rest. The value represents the percent difference that will cause the bull's-eye to reach its maximum brightness. Use low values to increase sensitivity and high values to decrease sensitivity.

"Washout full scale =" sets the scale in the washout bull's-eye. The value in this box represents the washout value that will cause the bull's-eye to have maximum brightness. Washout above this value will be pinned to maximum brightness.

"Normalizing square" sets the dimension of the box that is used to find the brightest region in the stress bull's-eye.

The "SPECT" popup menu is used to pick the SPECT reconstruction parameters from the list of names of settings made with the SPECT Reconstruction dialog box, e.g. "thallium". (The specifications for the particular type of reconstruction are set under the **SPECT...** selection of the **Analyze** menu.)

"Rest" selects the SPECT reconstruction to use for the rest script. The default is to use the same parameters as the stress portion.

"X-Y" and "Z" filterings are used on the oblique coronals (short axis views) after reorientation.

"Expand heart cube by" defines the final size if the image window based on the apex-to-base stress axis length (set during processing). The data cube should be big enough to enclose all of the data but not so big as to waste a lot of computer memory.

"Zoom coronals by" sets the size of the final short axis images in the All Frames Window.

"Reproject hearts", when checked, will cause the program to generate stress and rest volume rendered images during the bull's-eye plotting analysis.

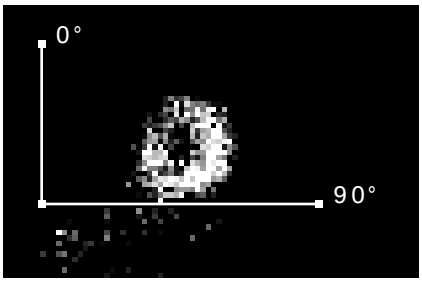
"Use database", when activated, will access the last selected reference database and generate comparison bull's-eyes between the database and patient being processed (See "Bull's-eye Databases" section for more details).

"DB Diff. Threshold" is used to set the point at which a difference between patient values and database values will cause the comparison bull's-eyes to reach maximum brightness. Units are in standard deviations from the mean database values. (See "Bull's-eye Databases" section for more details).

Note: be sure to click on the "Set Default" button when done to permanently store your selections.

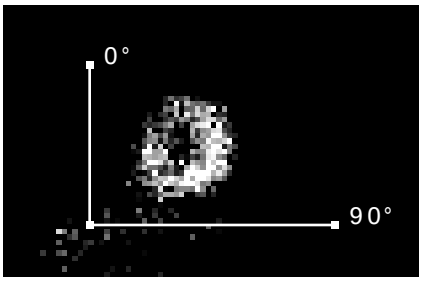
After loading the settings, the stress and rest scripts of the **Bull's-eye Plotting** selections from the **Analyze** menu are then used for the actual processing. Open the flood image and raw SPECT acquisition image files. Then select **Stress Script...** and follow the directions on the screen until the stress oblique coronal (short axis) images are generated. The following is a summary of the steps that you will encounter.

After clicking on the stress images and the flood correction image as requested by the script, the mouse track mode will be automatically activated for the stress image window. Move the mouse until the heart is approximately in the 45 degree LAO position, then click once to stop the mouse track mode, and then click on the stress image window. Axes will appear on the window. Adjust them so that you have a horizontal reference line near the inferior aspect of the LV to help your eye detect vertical displacements in the images.



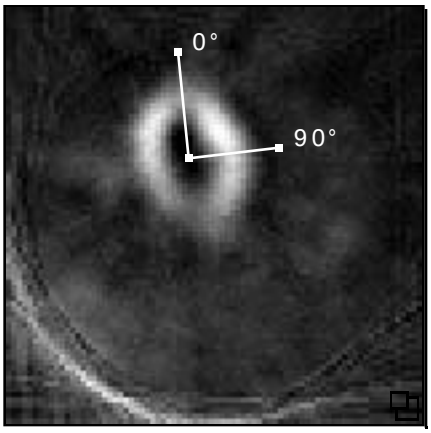
Then, click on "Frames" to reactivate the mouse track mode. Move the mouse to check the heart position in all of the frames and adjust their positions using the arrow keys when needed.

When you are finished adjusting the heart position in each of the frames, move to the frame with the heart apex aimed most outwards (the best pseudo-45 degree LAO view) and then click out of the mouse track mode. Then adjust the 0 degree axis length so that it covers the area that you wish to reconstruct. Be sure to go a few pixels beyond the LV in both the north and south directions. Note that the 90 degree axis has no affect in the reconstruction.

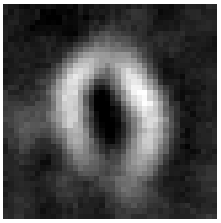


Then click on "Continue".

The program will prefilter the original data (if a prefilter was selected), do the filtered backprojection, and display a central transverse slice. The apex of the heart will be aimed at 12 o'clock based on the angle of the image that was left in the front of the movie window at the end of the motion correction. Click in the center of the LV and axes will appear. Place the origin of the axes in the center of the LV. Adjust the zero-degree axis so that it is several pixels beyond the apex as shown below. If the apex is tilted slightly from the 12 o'clock position, you can tilt the zero-degree axis slightly to compensate and the final short axis images will reflect the realignment.

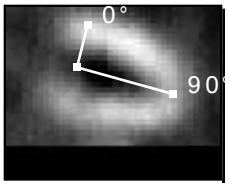


The length of the zero-degree axis will be used as half of the side dimension of a new box cut from the original box but without most of the background (to save memory). The new box will be displayed. Note that the 90-degree axis length has no effect at this point.



Also, a sagittal (tilted horizontal long axis) view through the center point will be generated.

Click on the sagittal view and axes will appear. Put the axes' junction point on the base and drag the 90 degree axis to the apex as shown below.



Click on "Continue" and the oblique coronals (short axis views) will be generated in a movie window.

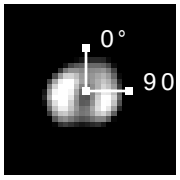
Because of the large amount of memory used for bull's-eye generation, it may be useful to close all of the windows except the oblique coronal images and flood image at this point (unless your MAC is running under System 7 and has more than eight megabytes of RAM available). You will also probably want to save the short axis slices for later viewing.

Next, open the rest images. Then select the **Rest Script...** and follow the screen directions. After clicking on the stress short axis slices, the rest original images, and the flood image as directed by the script, follow the directions to lead you to the point where the rest oblique coronals are generated as with the stress script summarized above.

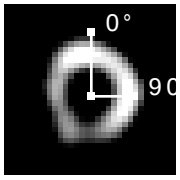
The program then brings up the apical slice from the stress coronals. Note that the apical slice will be determined from where you put the axes' junction point on the sagittal image. Set the center point and radius using the zero-degree axis. Note that the 90-degree axis length has no effect at this point. Click on "Continue".

(Sometimes, for example with an apical defect, you may wish to set the apex on a slice where you cannot easily define the center of the slice and/or its radius. The viewed slice can be changed by activating the mouse track mode with the "Frames" button if needed. Just note from the Controls box which frame represents the apex,

move to a slice where the myocardium can be seen well, set the axis position and radius, move back to the apex frame, then click on "Continue".)

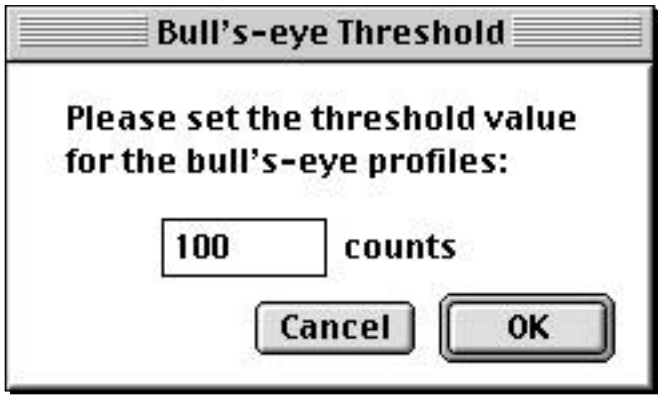


The program then brings up the base slice from the stress images based on where you put the end of the 90-degree axis.



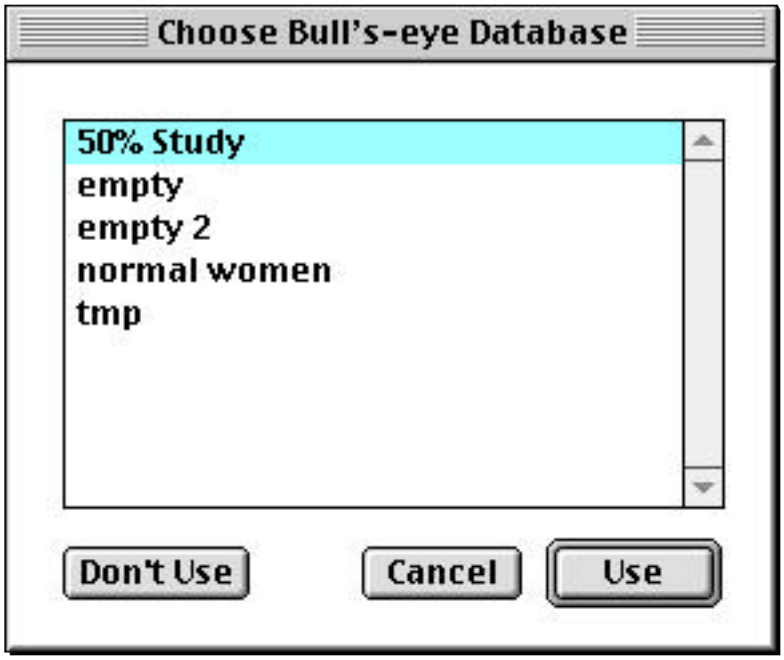
Again, change the slice if needed and then set the center and radius using the zero-degree axis. Click on "Continue" and the program will place the regions around the LV in all of the slices between the apex and base. Review and change them if needed by activating the mouse track mode by clicking on "Frames". Then hit "Continue" when all desired changes are done. (Note: the regions should lie just outside of the edge of the slices).

When the average or line integral methods are used, a dialog box appears asking for the "Threshold" value. This value is used to threshold out the background pixels so that they will not be included in the analysis. To determine the best value for any given data set, move the pointer around in the background near the myocardium and observe the number of counts in the Controls window. The number will vary so note a value near the maximum that is observed. Then move the pointer over myocardial regions and observe the count values. Choose a value for the threshold that is greater than most of the background values but less than the myocardial values. (This option does not occur with the maximum pixel value method.)



After setting the value, click on "OK". The stress bull's-eye is generated.

The program then moves to the rest apex and then the rest base. Set the center points and radii as with the stress images, review the regions, set the “Threshold” value if requested, then hit “Continue”. If the “Use Database” box was checked in the “Bull’s-eye Settings”, a window will appear that allows the user to pick a database.



If the database function is not activated, the program generates a window containing the stress bull’s-eye, the rest bull’s-eye, the normalized rest bull’s-eye, the washout bull’s-eye, and the difference bull’s-eye. (See the Bull’s-eye Databases section for the bull’s-eyes that are generated with the database function activated).

Stress: Bull’s-eye containing information regarding stress short axis profiles. The type of plot is determined from the parameters in the “Bull’s-eye Settings” window.

Rest: Bull’s-eye containing information regarding rest short axis profiles.

Normalized rest: Bull’s-eye in which the rest information is normalized to the stress information. The program finds the rectangular area (dimensions set with the “Normalizing square” value in the “Bull’s-eye Settings” window) in the stress bull’s-eye with the greatest activity. The program then finds the same area in the rest bull’s-eye and normalizes the rest bull’s-eye activity so that activities in the reference areas are the same.

Washout: Bull’s-eye in which each pixel value reflects [(stress-rest) / stress]

Difference: Bull’s-eye in which each pixel value reflects [(normalized rest - stress) / normalized rest]

Also, All Frames Windows containing the stress and rest oblique coronals (short axis views) are generated.

If the "Reproject hearts" box is checked, then the program generates stress and rest volume rendered image windows.

Notes regarding reprocessing of scans. At times, it may be desirable to reprocess a scan that has already been processed once. To facilitate this, it is useful to save the flood and motion corrected acquisition images after the first processing procedure. Then the flood and motion corrections do not have to be performed again. It is important to remember, however, not to flood-correct data that has already been flood corrected once, or to ever flood-correct data that has already been motion corrected (since the flood image would not match up to any frames that had been shifted during the motion correction process). To skip the flood correction step in the Bull's-eye Plotting script, just click on some non-image window within the NucLear MAC program (such as the color bar) when the script says to click on the flood image.

7.11 Bull's-eye Databases

7.11.1 Basis

In the evaluation of bull's-eye data from a particular patient, it is sometimes helpful to be able to compare them to reference values from normal patients. Towards this goal, programs have been developed by some vendors that include fixed internal reference patterns derived from series of normal patients. While these fixed references are sometimes helpful, their use is not without problems.

One possible problem with this approach is that the contrast between normal and abnormal areas is affected by the type of filtering performed during the SPECT reconstruction. Therefore, a user must obtain images with the same spatial resolution as those of the database and use the same types of prefiltering and backprojection filtering to make the comparisons valid. This requirement reduces flexibility in the analysis (and even the acquisition) of the SPECT data.

Another problem is that contrast between normal and abnormal regions in the stress images is affected by the duration of the acquisition. As redistribution of thallium can begin fairly rapidly, long acquisitions may reduce the magnitude of the difference in activities between ischemic and normally perfused tissue.

The NucLear MAC avoids these problems by allowing the user to generate databases from reference patients of their own choosing whose images are processed in the same manner that is used in the processing of the images being interpreted (See reference below for details). Rather than relying upon some fixed reference patterns, the user can process and store images to create databases that reflect particular needs. For example, if a user wishes to restrict the age or weight of the patients within a reference database, then the user can create the database accordingly. The user is free to create as many databases as he or she feels are needed to reflect different categories of patients.

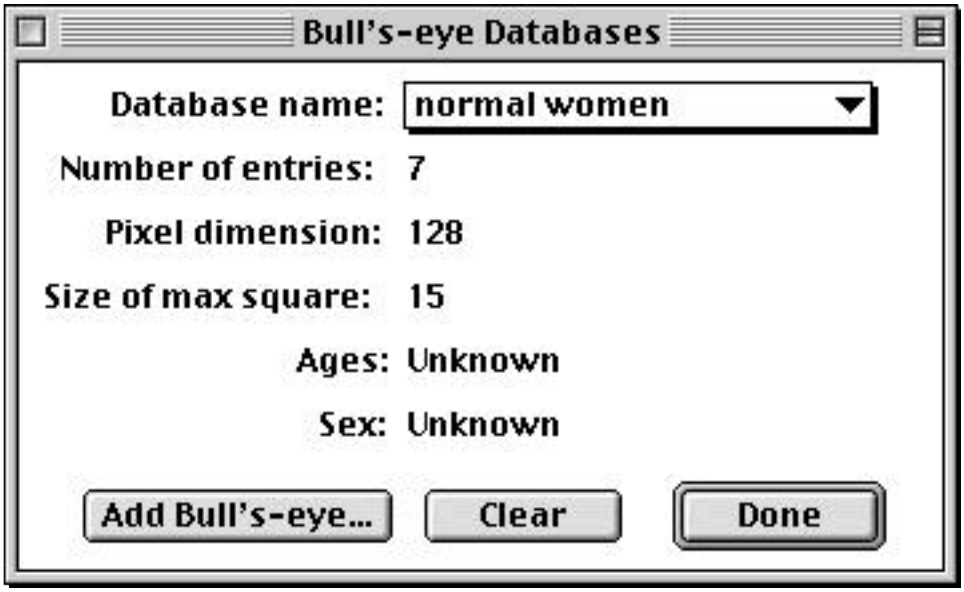
Lear J, Pratt J, Gregory C, Mallet J. User-definable bull's-eye database analysis. *Eur J Nucl Med* 21: 154-157, 1994

7.11.2 Processing

Databases contain information from the “Stress”, “Difference”, and “Washout” bull’s-eyes.

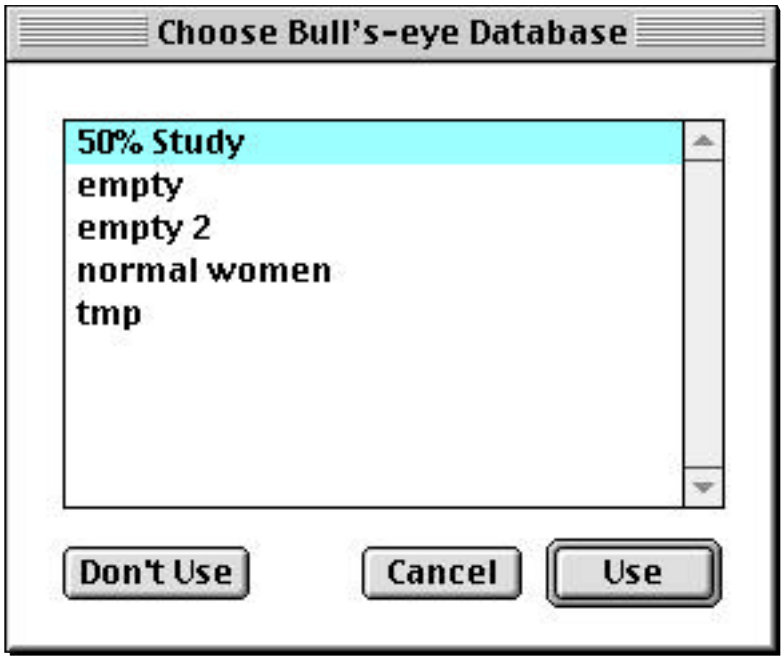
A new database can be generated or an existing database can be modified following the same basic procedure.

Bring up the all frames window containing the bull’s-eye plots from the patient to be added to the database. Choose **Databases**… from the **Analyze>Bull’s-eye Plotting** submenu. The “Bull’s-eye Databases” dialog window appears.



Choose the name of the database to be modified from the “Database name” popup menu, or leave the selection untitled if a new database is to be started. Click on “Add Bull’s-eye”. A message appears that directs the user to click the mouse on the window containing the new bull’s-eye images. Click on the bull’s-eye window as indicated and the data from that patient is combined with that of the existing database. The “Database name” will change to “Untitled” to indicate that the modification has occurred. Add the same name back to the list if you wish to save the changes to the original database or enter a new name if you wish to make a new database without affecting the older one. Note: if you reenter the name of an existing database, a warning box appears that allows you to stop the action if it was inadvertent.

The database comparison information is automatically generated during the Bull’s-eye Reconstruction process if the “Use database” box is checked in the Bull’s-eye Settings window. The program uses whatever database is selected from the list that appears during the processing.



The following bull's-eyes are generated:

Stress: Bull's-eye containing information regarding stress short axis profiles. The type of plot is determined from the parameters in the "Bull's-eye Settings" window.

Rest: Bull's-eye containing information regarding rest short axis profiles.

Difference: Bull's-eye in which each pixel value reflects [normalized rest - stress) / normalized rest]

Washout: Bull's-eye in which each pixel value reflects [(stress-rest) / stress]

Database stress: Bull's-eye in which the mean values (maximum normalized to 1000) of the stress data from the database is displayed.

Normalized rest: Bull's-eye in which the rest information is normalized to the stress information. The program finds the rectangular area (dimensions set with the "Normalizing square" value in the "Bull's-eye Settings" window) in the stress bull's-eye with the greatest activity. The program then finds the same area in the rest bull's-eye and normalizes the rest bull's-eye activity so that activities in the reference areas are the same.

Database difference: Bull's-eye in which the mean values of the difference data from the database is displayed.

Database washout: Bull's-eye in which the mean values of the washout data from the database is displayed.

Stress comparison: Bull's-eye in which the difference between the database stress bull's-eye and the patient's stress bull's-eye is displayed. Units are in standard

deviations from the mean of the database stress bull's-eye. The "DB Diff. Threshold" value in the "Bull's-eye Settings" window is used to set the point at which the color scale reaches the maximum value.

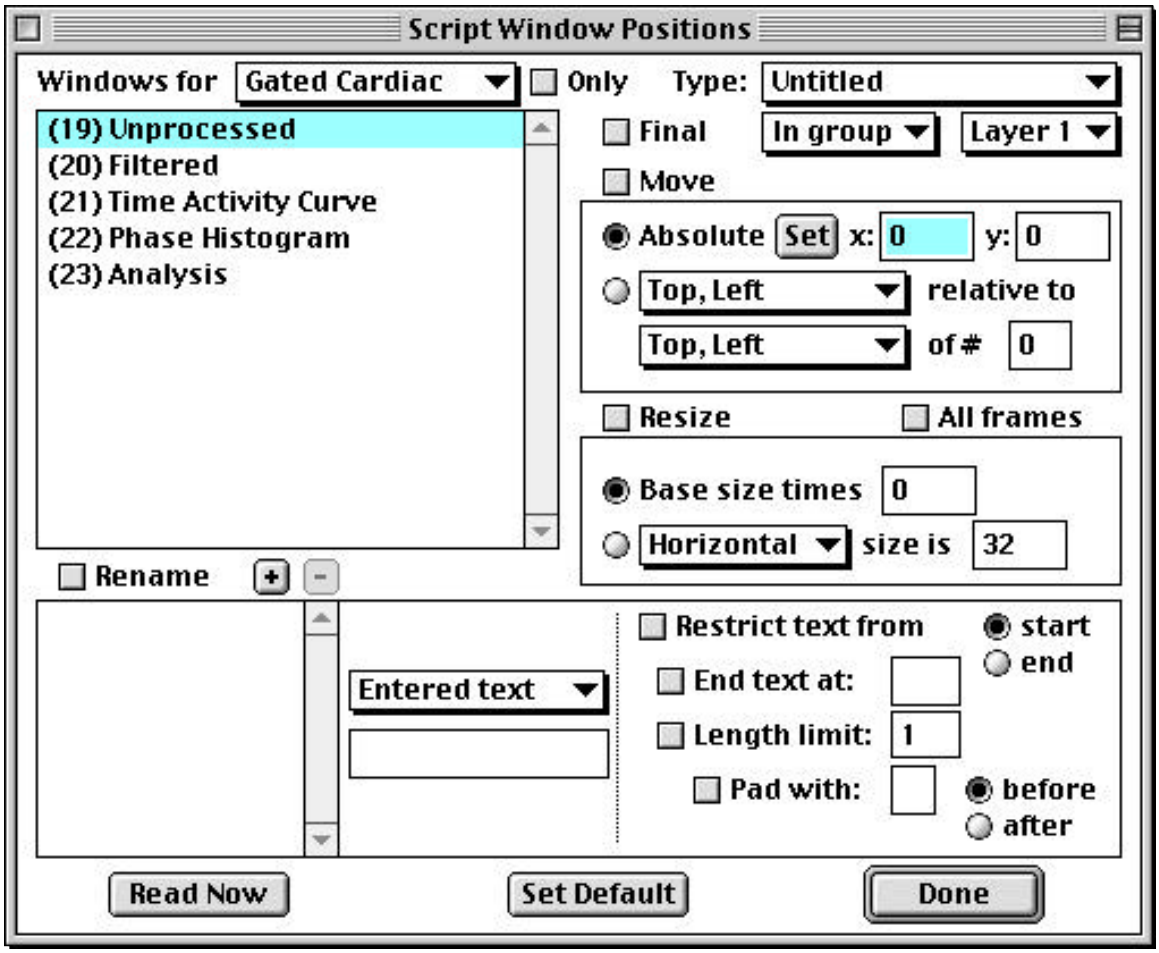
Stress masked: Bull's-eye in which the stress data is shown but with those pixels that are beyond the threshold for both stress and difference zeroed out.

Difference comparison: Bull's-eye in which the difference between the patient's difference bull's-eye and the database difference bull's-eye is displayed. Units are in standard deviations from the mean of the database difference bull's-eye. The "DB Diff. Threshold" value in the "Bull's-eye Settings" window is used to set the point at which the color scale reaches the maximum value.

Washout comparison: Bull's-eye in which the difference between the database washout bull's-eye and the patient's washout bull's-eye is displayed. Units are in standard deviations from the mean of the database washout bull's-eye. The "DB Diff. Threshold" value in the "Bull's-eye Settings" window is used to set the point at which the color scale reaches the maximum value.

7.12 Setting Window Positions for Processing Scripts

Because NucLear MAC supports the use of many different sizes of monitors, there is no one "best way" to set the locations at which the various windows appear during the clinical processing scripts. Therefore NucLear MAC provides a method by which a user can set the positions as desired. Both absolute (the user specifies a specific screen location for a window) and relative (the user specifies that a window should appear in some location relative to another window) positioning are supported. The location controls are activated through the **Script Windows...** selection of the **Edit>Preferences** submenu.



The use of the positioning controls is probably best described by an example. Consider a gated cardiac study.

Select Gated Cardiac from the **Windows for** popup menu. A list of windows that can be present during processing appears (Note: there may be items on the list that are not used).

As the first item in gated processing is the raw, unprocessed data select "19. Unprocessed". Click on the "Move" box to indicate that the computer should move the window during the processing. Click on "Absolute" to indicate that the window should be positioned in an absolute location. Click "Set" to set the location. The cursor turns into a hand. Click with the hand somewhere near the upper, left side of the monitor. This will be the location where the upper left corner of the gated raw data window is placed by the computer during the execution of the gated script.

Select "20. Filtered" from the list. For this window, we will use relative positioning. Click the "Move" button on if it is not on already. Select "Top, Left" relative to "Top, Right" of # "19" to indicate that the top, left of the Filtered window should be placed next to the top, right of the Unprocessed window.

Select “21. Time Activity Curve” from the list. Make sure the “Move” checkbox is on, then select “Top, Left” relative to “Top, Right” of # “20” to place the Time Activity Curve window to the right of the Filtered window.

Select “22. Phase Histogram” from the list. Make sure the “Move” checkbox is on, then select “Top, Left” relative to “Top, Right” of # “21” to place the Phase Histogram window to the right of the Time Activity Curve window.

Select “23. Analysis” from the list and make sure the “Move” checkbox is on. We will not place this window further to the right because there will not be room on most monitors. Rather we will place it below the other windows. Select “Top, Left” relative to “Bottom, Left” of # “19” to indicate that the window should be positioned below the Unprocessed window. Change the value in the “y” box from its default vale of 0 to 100. This will indicate that the top left corner of the Analysis (all) window should be 100 pixels below the bottom left corner of the Unprocessed window (so it will not overlay the other windows).

Add a name for these settings to the **Type:** list. Different names can be used to store different settings.

Hit "Set Default" if you wish these settings to be used as default positions.

Now, when the “Gated Cardiac” processing script is run, the windows are positioned as set. This gives the screen a less cluttered appearance during processing without restricting the flexibility of NucLear MAC.

8 Macros

8.1 Introduction

NucLear MAC has a powerful built-in macroing capability. This means you can add complex functions to the NucLear MAC program that are not part of its basic repertoire. Macros are a pre-entered series of actions that NucLear MAC performs automatically. You can write macros that behave like the scripts available under NucLear MAC's **Analyze** menu. For example, the gated cardiac script prompts the user to select a gated study, and then asks the user to draw various regions of interest (ROIs). The script then analyzes the data and displays the results.

Macros support most standard numerical functions and NucLear MAC menu commands. A typical use for a macro might be to perform special filtering on an image, have the user draw ROIs on the result, and then do some calculations based on the activity in the ROIs. The macro would report the results in the Analysis window.

You can use macros to perform many other tasks. Macros support a variety of file- and folder-manipulating functions. Thus, you can use macros to save and organize studies in specified locations, thereby avoiding the standard file dialog. The standard file dialog is what you get when you choose **Save As...** from the **File** menu. It requires you to find the desired folder and enter a file name for the study. Similarly, you can use macros to directly open frequently used images, such as flood correction matrices.

You can also use macros to customize NucLear MAC's screen display. Macros can show, hide and reposition the standard NucLear MAC windows, such as the color bars and the Controls window. Macros can reposition and resize windows containing studies.

In essence, NucLear MAC macros are computer programs. The programming language combines aspects of BASIC, Pascal and C. A macro consists of a series of statements, each of which must obey rules of syntax. When you run a macro, NucLear MAC interprets and executes one statement at a time, beginning with the first. Macros can perform *conditional branching*. That is, they can make decisions based upon current information, and perform different sets of actions depending on the results. For example, depending on the resolution of an image, a macro could apply different types of filters.

8.2 Creating Macros

You can create macros by two different methods. Most often you will use a combination of the two approaches. The first method is to "teach" NucLear MAC by having the program record a series of your actions. The second method is to type macro instructions directly into a macro window.

8.2.1 Recording a Macro

To record a new macro, select **Record New** from the **Macros>Create** submenu. This creates a new, empty macro window behind all other windows. The text of the **Record New** menu command changes to **End Recording**. Select the **End Recording** command when you are finished with the actions you want automatically entered in the macro. Between when you select **Record New** and when you select **End Recording**, macro statements are automatically written to the new macro window. During this period, you should perform only those actions you want to be part of the macro.

If you decide to keep a macro you have recorded, then select the macro window and choose **Save As…** from the **File** menu. You are prompted to select where and under what name the macro should be saved. It is a good idea to keep all of your active macros in one folder. That way you can make their names appear under the **Macros** menu for instant access (See the information about the **Set Directory…** command below).

Most menu commands can be recorded automatically. Additionally, actions such as drawing oval or rectangular ROIs and placing arrows can be recorded. However, there are a number of actions that are too complex or too common to record implicitly. These actions are recorded explicitly. They are listed in the **Commands** submenu under the **Macros** menu. When you want to record one of these actions, simply select it from the menu.

Change Frame…

Use the **Change Frame…** command to record an instruction to change the active frame of an image window. If an image window is front-most, then it is used for the command. Otherwise, you are prompted to select the image. You must set the frame of the image window prior to selecting the **Change Frame…** command. For example, suppose you want to ensure that the first frame of an image is active. First, bring the image to the front, then move to the first frame, either by stepping or by using the Controls window. Finally, select **Change Frame…** from the **Commands** submenu.

Move Window…

Use the **Move Window…** command to record an instruction to move a window to its current (at time of recording) location, regardless of where it is when the macro runs. To use this command, move the window of interest to the location where the macro should move it, then select the **Move Window…** menu command. You are prompted to select the window of interest, and the command is recorded.

Prompt for Image…

Use the **Prompt for Image…** command to record an instruction that prompts the macro user for an image window. With this instruction, the user can identify the image or images on which the macro should operate. When you select the **Prompt for Image…** command, you are first asked to enter text that will be presented to the user. This is typically something like “Please click on the image window to use for this function. Any other action cancels.” Next, you are given the prompt, and you must select the window to use for the recording.

Prompt for Frame…

Use the **Prompt for Frame…** command to record an instruction that prompts the macro user to choose a particular frame of an image window. If the front window is an image window, it is used for the command, otherwise you are asked to identify the target window. You are then asked to enter the text that is presented to the user when the macro runs. This is typically something like “Please move the mouse to select the desired frame. Then click to disengage the mouse, and press Continue to proceed.”

Prompt for Graph…

Use the **Prompt for Graph…** command to record an instruction that prompts the macro user for a graph window. With this instruction, the user can identify the graph or graphs on which the macro should operate. When you select the **Prompt for Graph…** command, you are first asked to enter text that will be presented to the user. This is typically something like “Please click on the graph window to use for this function. Any other action cancels.” Next, you are given the prompt, and you must select the window to use for the recording.

Prompt for Number…

Use the **Prompt for Number…** command to record an instruction that prompts the macro user to enter a number. You are asked to enter the prompt that will be given to the user. Note that the **Prompt for Number…** command records an assignment statement. In order to access the number in the macro, you must write additional statements using the variable from the assignment. That is, none of the automatically recorded statements can use numbers entered by the user without some programming on your part.

Prompt for Oval Region…

Use the **Prompt for Oval Region…** command to record an instruction that prompts the macro user to draw an oval region. If the front window is an image, then it is used for the command. Otherwise, you are asked to identify the image window. You are then asked to enter the prompt that will be given to the user. Finally, you are asked to draw the region, just as the macro user will be asked when the macro is run.

Prompt for Rect. Region…

Use the **Prompt for Rect. Region…** command to record an instruction that prompts the macro user to draw a rectangular region. If the front window is an image, then it is used for the command. Otherwise, you are asked to identify the image window. You are then asked to enter the prompt that will be given to the user. Finally, you are asked to draw the region, just as the macro user will be asked when the macro is run.

Prompt for Region…

Use the **Prompt for Region…** command to record an instruction that prompts the macro user to draw a freehand region. If the front window is an image, then it is used for the command. Otherwise, you are asked to identify the image window. You are then asked to enter the prompt that will be given to the user. Finally, you are asked to draw the region, just as the macro user will be asked when the macro is run.

Prompt for Region Move…

Use the **Prompt for Region Move…** command to record an instruction that prompts the macro user to reposition regions in a particular image frame. If the front window is an image, then it is used for the command. Otherwise, you are asked to identify the image window. You are then asked to enter the prompt that will be given to the user.

Prompt for String…

Use the **Prompt for String…** command to record an instruction that prompts the macro user to enter a string. You are asked to enter the prompt that will be given to the user. Note that the **Prompt for String…** command records an assignment statement. In order to access the string in the macro, you must write additional statements using the variable from the assignment. That is, none of the automatically recorded statements can use strings entered by the user without some programming on your part.

Resize Window…

Use the **Resize Window…** command to record an instruction to resize a window to its current (at time of recording) size, regardless of its size when the macro runs. To use this command, resize the window of interest to the desired size, then select the **Resize Window…** menu command. You are prompted to indicate the window of interest, and the command is recorded.

Select Window…

Use the **Select Window…** command to record an instruction to make a window active. That is, when the macro runs, the window is brought to the front. When you choose the **Select Window…** menu command, you are prompted to indicate the window to which the macro command applies.

8.2.2 Writing a Macro

In order to write a new macro, you must create a new, empty macro window. Do this by selecting **Record New** from the **Macros>Create** submenu. Then select **End Recording** from the same position in the **Create** submenu. You can now type in macro instructions. Follow the guidelines given in the section titled Macro Structure and Elements. Read the section on editing macros for an explanation of the various editing commands.

If you decide to keep a macro you have written, then select the macro window and choose **Save As…** from the **File** menu. You are prompted to select where and under what name the macro should be saved. It is a good idea to keep all of your active macros in one folder. That way you can make their names appear under the **Macros** menu for instant access (See the information about the **Set Directory…** command below).

8.3 Running Macros

One way to run a macro is to bring the macro window to the front and select the **Run Macro** command from the **Macros>Create** submenu. This requires that you have the macro open for editing. It is convenient if you are developing and testing a macro. However, this procedure is cumbersome if you are frequently using a macro and have no need to look at the macro instructions. Fortunately, there is an alternative way to run macros. You can ask NucLear MAC to monitor a particular folder, and display the names of all macros in the folder. These names appear at the end of the **Macros** menu, after the dotted gray line. When you choose a macro name from the **Macros** menu, the macro is automatically retrieved from disk and executed. As a shortcut, if you hold down the option key while selecting the macro, it is opened rather than executed.

Another way to run a macro is to export it as a user function. When you open the user function file, the macro runs. You cannot edit a user function file, so you should be certain your macro works properly before you export it. To export a macro, select the macro's window and choose **Export…** from the **File** menu. Enter an appropriate name in the standard file-save dialog. DO NOT replace the original macro file. If you do you will be unable to access the macro's text. Make sure you have a backup copy of the macro in case you need to change it. When you make changes, simply repeat the export procedure.

8.3.1 Setting the Macros Directory

To set the directory (folder) containing macros that NucLear MAC should automatically list, select **Set Directory…** from the **Macros>Create** submenu. Use the standard file dialog to choose the folder where you are storing your active macros. NucLear MAC will remember the folder, even if you quit and restart the program. With system software prior to 7.0, NucLear MAC will lose track of the macros folder if you rename the folder's volume, but not if you rename the folder itself.

8.4 Macro Structure and Elements

A macro consists of a series of statements on successive lines. It is important to distinguish between a new line created by a return character and one made by text wrapping around from the right side of the document. The way text wraps around (when the macro window is too narrow) is irrelevant to the execution of the macro. On the other hand, new lines created by pressing the return key are what separate macro statements.

8.4.1 Line Numbers

To facilitate branching during execution, each line can begin with an optional line number. A line number must be unique for a particular macro, and it must be in the range of zero to 65535. Line numbers are assumed to exist in the following situations 1) One or more numeric characters ('0' through '9') are at the beginning of the macro or 2) One or more numeric characters follow a return character, that is, appear at the start of a line. Line numbers are used as arguments to the **`goto`** and **`gosub`** functions.

8.4.2 Comments

You can enter comments in macros, but they must appear at the ends of lines. The semicolon ';' is used to begin a comment. The comment extends to the end of the current line or the end of the current macro, whichever comes first. Blank lines are ignored, so they can be used to set off comments and otherwise improve the appearance of your macros.

8.4.3 Types

Macros support various data types. These types range from the straightforward to the abstract. By keeping track of data types, the NucLear MAC can prevent inappropriate actions from being performed. The types are as follows:

T_ANY
This type encompasses any of the following types..

T_NO_INIT
This is the undefined type. You will only encounter it if you attempt to use the return value of a function that returns nothing. Using the undefined type generally causes an error during macro execution.

T_IMAGE
This type represents an image window. It can be either a movie-type window or an all-frames window, but not both.

T_NUMBER
This type indicates a number. Internally, the number is stored as a floating-point value. Numbers are written in macros in standard IEEE format. For example, 1, 3.14159, 2.0e-10

T_STRING
This type indicates a string of characters, usually plain text. This type can hold from zero up to a maximum of 255 characters. Strings are written as characters surrounded by double quotes. For example, "This is a string". You can embed the double quote in a string by prefacing it with the back-slash character. For example, "He said \"Hello.\"" You can include a return character in a string by typing '\r'. For example, "This is one line.\rThis is a new line."

T_ARRAY
The array type represents a vector of two or more elements. Each element can be of any type. You access individual array elements with an index in square brackets. The index can range from zero up to the number of array elements less one. For example, the third element of an array named An_Array would be accessed as An_Array[2].

T_GRAPH
This type represents a graph window.

T_REGION
This type represents a region of interest in a particular image window.

T_WINDOW
This type represents any NucLear MAC window. It is a superset of the image and graph types.

T_PRINTREC
This type represents the printing information for a window. Typically this includes page orientation and the like.

8.4.4 Variables

Macro variables are dynamic storage areas for any of the data types described above. A variable name consists of from one to 255 characters. It must begin with an alphabetic character or the underscore character '_'. Subsequent characters can be numeric, in addition to alphabetic or underscore. You cannot use a variable with the same name as a built-in function.

You do not need to formally declare variables. Any assignment statement defines a new variable, if that variable does not already exist. Although it is good programming practice to maintain one type of data in a given variable, this is not required; a variable of any type except an array can be reassigned to another type.

All variables are local to the macro in which they are defined. That is, a variable used in one macro cannot be directly used by a macro that is called by the first macro. This is a minor limitation, because a macro can pass any number of parameters to a macro it calls.

8.4.5 Operators

Macros support many operators, most of which apply to numbers. They are listed below in order of increasing precedence. Unless stated otherwise, operators are binary and evaluate from left to right.

,
The comma operator generates vector elements. The elements can be of any type. The arrays resulting from the comma operator are also used to pass arguments to functions.

=
The assignment operator places the value generated by the expression to its right in the variable reference to its left. For example, a[2] = cos(4.0). Any type can be assigned except the undefined type. The assignment operator returns the value that was assigned. Evaluation is from right to left.

== !=
These operators evaluate equality or inequality of two numbers or two strings. '==' returns one if the items on each side are the same, and zero if they are not. '!=' returns one if the items on each side are different, and zero if they are not.

<= < >= >
These operators evaluate the relative sizes of two numbers. '<=' returns one if the number to the left is less than or equal to the one to the right, and zero otherwise. '<' returns one if the number to the left is less than the one to the right, and zero otherwise. '>=' returns one if the number to the left is greater than or equal to the one to the right, and zero otherwise. '>' returns one if the number to the left is greater than the one to the right, and zero otherwise.

+ -
The add operator '+' returns the sum of the left and right numbers, or the concatenation of the left and right strings. The subtract operator '-' returns the difference between the left and right operands.

* /
The multiply operator '*' returns the product of the left and right numbers. The divide operator '/' returns the ratio of the left and right operands.

^
The power operator returns the left operand raised to the power of the right operand. The second operand must be greater than zero. This operator evaluates from right to left.

- !
The unary minus operator '-' returns the negated value of the number to the right. It evaluates from right to left. The unary logical negation operator '!' returns one if the number to the right is zero, and zero otherwise. It also evaluates from right to left.

( )
The parentheses are used to change the precedence of other operators, and to demarcate function arguments.

8.5 Editing Macros

If you have used any word processor on a Macintosh, then you should have little trouble learning to edit NucLear MAC macros. Text for macros is displayed in resizable windows. You enter macro text by bringing the macro window to the front, clicking on the desired spot for the insertion point, and typing the information.

The standard Macintosh editing features are available in macro windows. These include **Undo**, **Cut**, **Copy**, **Paste** and **Clear**, all of which are found under the **Edit** menu.

Undo

Choose **Undo** to cancel your last action. The name of the action is appended to the Undo command, so you can be sure of what you are undoing. For example, the Undo command can be "Undo Paste" or "**Undo Typing**." If you undo an action, the **Undo** menu command changes to "**Redo Action**," so that if you change your mind again, you can quickly repeat the action you canceled. The **Undo** menu command is enabled only when the window with the last undoable action is in front.

Cut

Use the **Cut** command to remove selected text from a macro and place it in the clipboard. Once text is in the clipboard, you can use the **Paste** command to place it in different locations in the macro, or in other macros. The **Cut** menu command is enabled only when one or more characters are selected in a macro.

Copy

Use the **Copy** command to place a copy of the selected text into the clipboard. As with the **Cut** command, once the text is in the clipboard, you can use the **Paste** command to place it in different locations in the macro, or in other macros. The **Copy** menu command is enabled only when one or more characters are selected in a macro.

Paste

Use the **Paste** command to replace the current selection with a copy of the text in the clipboard. The **Paste** command is enabled for a macro window only when there is text in the clipboard; you must have previously cut or copied some text.

Clear

Use the **Clear** command to delete the current text selection. The **Clear** menu command is enabled only when one or more characters are selected in a macro. Pressing the "delete" key is equivalent to choosing the **Clear** menu command.

Select All

Use the **Select All** command to expand the current text selection to include all the text in a macro.

8.6 Miscellaneous Text Editing

You have several options for enhancing the appearance of macro text. First, each character can be drawn in any of the fonts installed on your Macintosh. To change the font of the current text selection, locate the **Font** submenu under the **Edit** menu. Then select the new font in which to display the selection. You can reverse this action by choosing **Undo Formatting** from the top of the **Edit** menu. If you specify a new font when no text is selected, then any new text you type at the current insertion point will be in the newly selected font.

You can also change the font size of any text selection in a macro. Locate the **Size** submenu under the **Edit** menu and choose the desired point size. Any selected text is redrawn in the new size. You can reverse this action by choosing **Undo Formatting** from the top of the **Edit** menu. If you specify a new size when no text is selected, then any new text you type at the current insertion point will be in the newly selected size.

You can apply any combination of styles to a text selection. Select the text and locate the **Style** submenu under the **Edit** menu. Choose the style options you require for the selected text. You can choose among bold, italic, underline, outline and shadow. Choosing the **Plain** command removes all styling from the current selection. Individual style options can be removed as well as applied; if a style option applies to the current text, then the corresponding menu command has a check mark to its left. Choosing a checked style command removes that style from the selected text. You can reverse a style change by immediately choosing **Undo Formatting** from the top of the **Edit** menu. If you specify a new style when no text is selected, then any new text you type at the current insertion point will use the new style option.

8.7 File Operations

NucLear MAC macros are stored as text files. You can read macros with word processors outside of the NucLear MAC. However, you should not attempt to edit macro files without the NucLear MAC. This is because the font, size and style data are stored separately (in the resource fork), and most word processors cannot access them. Thus, changing macro text outside of the NucLear MAC can result in inconsistencies between the text and its formatting information. The exception to this is the SimpleText editor provided by Apple. SimpleText preserves style information in macros.

Many of the NucLear MAC file commands apply to macros, as well as images and other documents. The following commands are found under the NucLear MAC **File** menu.

Open...

Just as you can open images and graphs with the **Open...** command, you can also

select and open NucLear MAC macros. After you have opened a macro, you can edit it and run it. There is a shortcut to avoid the **Open...** command for macros that are installed in the **Macros** menu. Hold down the option key and select the name of the macro from the **Macros** menu. This opens the macro and does not run it.

Close

Use the **Close** command to remove a macro window when you are done with it. If the macro has never been saved, you are asked if you want to save it. Also, if the macro has been modified since it was last saved, you are asked if you want to save the changes.

Save

Use the **Save** command to save the changes to an existing macro, or to save a newly created macro to disk. In the latter case, a standard file dialog is presented so you can specify where and under what name the macro should be saved. See the **Save As...** command below.

Save As...

Use the **Save As...** command to save a newly created macro to disk, or to save an existing macro under a different name. A standard file dialog is presented so you can specify where and under what name the macro should be saved. Typically you will want to keep all of your active macros in one folder. That way you can have all the macros show up as commands under the **Macros** menu, available for instant access. For more information about this, see the description of the **Set Directory...** command.

Page Setup...

Use the **Page Setup...** command to access the standard Macintosh page setup dialog. The parameters you can set depend on the type of printer currently selected. Use the Chooser desk accessory to change printers. If you hold down the option key while selecting the **Page Setup...** command, then the resulting page setup is also made the default for all new macros.

Print...

Use the **Print...** command to print a macro on the current printer. Use the Chooser desk accessory to change printers. A standard Macintosh print dialog is presented. You can enter the number of copies you want to print and set various other printer-dependent parameters.

8.8 Functions

The rest of this document describes the functions available to NucLear MAC macros. Descriptions of most computer languages would include a separate section for programming statements. However, in NucLear MAC macros, all programming statements are implemented as functions. This explains why, unlike in other languages, the `goto` instruction requires its target line number to be in parentheses. For example, `goto` (20)

Function descriptions appear in the following format:

return_type **FunctionName**(arg1_type *arg1_name*,arg2_type *arg2_name*,…)

The optional return_type describes the type of value that results from calling the function. If the result of a function is assigned to a variable, then the variable has the type and value of the function result. The available return types and argument types are described above. If a function takes no arguments, then no parentheses are used in calling it.

8.8.1 Macro Execution

The functions described in this section are used to control the execution of macros. These functions represent the primary programming statements.

args(T_NUMBER *argument_number*)
The `args` function is used by a macro to access the parameters that were passed to it. When a macro is run directly by the user, no arguments are passed to it. However, any macro is capable of calling another macro with a list of arguments. The called macro accesses a particular argument by calling `args` with the index of the desired argument. The *argument_number* parameter can range from zero to the number of parameters less one. Use `countargs` to find out how many arguments were passed.

T_NUMBER **ArraySize**(T_ANY *argument*)
Call `ArraySize` to find out how many elements are in an array. If a value of one (1) is returned, then the argument is not an array. One use for this function is to find out how many values have been returned from a function that can return any number of items (see `OpenAll`, for example).

T_NUMBER **countargs**

Call `countargs` to find out how many arguments were passed to the macro. Use the `args` function to retrieve the arguments.

ClearArgs
Call `ClearArgs` to delete all saved macro function arguments. This call is typically made immediately after a macro function is called. It undoes the effects of all previous calls to `PutArg`.

T_ARRAY **DimensionArray**(T_NUMBER *array_size*)
Call `DimensionArray` to pre-allocate an array of a given size. For example, if you need to create an array with 20 elements, you can write: myArray = `DimensionArray`(20). You can access the array elements as myArray[0], myArray[1], … , myArray[19].

endsub

Use the `endsub` function to return from a subroutine. The point of execution returns to the most recently called `gosub` command. See the description of `gosub` below.

gosub(T_NUMBER *line_number*)
Use the gosub command to call a subroutine within a macro. The gosub function causes the macro execution point to be transferred to the line number given in the *line_number* parameter. Execution proceeds until an endsub statement is encountered. You can nest up to 16 levels of gosubs and endsubs.

goto(T_NUMBER *line_number*)
Use the goto function to transfer the execution point of the macro to the line number given in the *line_number* parameter. Typically goto is used in conjuction with the if command (see below) to effect conditional branches. For example: if (var > 20) goto (20).

if(T_NUMBER *condition*)
The if function allows a macro to make a decision based upon a condition. When the *condition* parameter is zero, the rest of the line is skipped, and execution continues on the next line. When the *condition* parameter is not zero, macro execution proceeds on the same line. For example, to limit a variable n to the range of $0 \leq n \leq 10$, you can write:

```
if (n < 0) n = 0
if (n > 10) n = 10
```

none

The none function causes no operation. Its intended use is to pass no arguments to a macro or user function. E.g., result = RunMacro("MyMacro",none).

Pause(T_NUMBER *seconds*)
Use the Pause function to introduce a delay in the execution of a macro. Pass the number of seconds to pause in the *seconds* parameter. During a pause, the macro can be aborted by typing a period while holding down the command (⌘) key or by pressing the esc key.

PutArg(T_NUMBER *which*, T_NUMBER *kind*,T_ANY *argument*)
Use the PutArg function to save an argument for a macro function. The *which* parameter specifies the particular argument, while the *kind* parameter indicates the format of the argument, which is passed in *argument*.. The *kind* parameter must be one of the following:

Value	Meaning
1	String is passed
2	Short integer is passed
3	Long integer is passed
4	Real number is passed

return(T_ANY *result*)
Use the return function to end the execution of a macro. If your macro does not need to return a result, then the use of return is optional; a macro automatically returns when there are no more lines to execute. If you want to end a macro early, and do not have a

value to return, then the *result* argument can be omitted. For example: `if` (done_early) return

T_ANY **`RunMacro`**(T_STRING *macro_name*, T_ANY *argument_list*)
Use the `RunMacro` function to call another macro. `RunMacro` attempts to execute the macro named by the *macro_name* parameter. `RunMacro` looks in the macros directory to find the macro. The macros directory is chosen with the Set Directory menu command. Information is passed to the submacro via the *argument_list* parameter. If no arguments are required, then use zero for *argument_list*. If more than one argument is needed, then use an array. For example:

result = `RunMacro`("A_SubMacro",(arg1,arg2,arg3))

The submacro accesses the arguments with the `args` function described above. The value returned by `RunMacro` depends on the submacro. If the submacro uses the `return` function described above, then `RunMacro` returns that value. Otherwise, `RunMacro` returns a value of type T_NO_INIT.

Recursion (a macro calling itself) is permitted, but due to heavy stack usage you will typically be limited to a depth of ten calls or fewer. Do not use recursion unless you know what you are doing; a macro that does nothing but call itself simply generates an error message.

A useful tip: quite often macros called by other macros are not intended to be called directly by users. In these instances, it is undesirable to have the submacros show up in the Macros menu. This seems hard to avoid, because `RunMacro` looks in the designated macros folder. Fortunately, all you need to do is create a folder within the macros folder. Place any submacros in this subfolder. Suppose you have a submacro named MySubMacro in a subfolder called MySubFolder. Then, to call the submacro from another macro, you would write:

result = `RunMacro`(":MySubFolder:MySubMacro",myArgs)

Note the placement of the two colons. The macro MySubMacro does not appear in the Macros menu, but can be called by any other macro. In fact, if you are familiar with Macintosh partial-path file names, then you can call submacros that are anywhere on the same volume as the macros directory. Note: in versions of the program beyond 2.3a2.0, *macro_name* can be a full path name.

8.8.2 Windows

These functions deal with general window manipulations. Additionally, access to some specialized NucLear MAC windows is provided.

T_WINDOW **`AnalysisWindow`**
This function returns a value representing the NucLear MAC Analysis window. The window value can be passed to window-manipulating and editing function calls.

T_WINDOW **`ColorsWindow`**(T_NUMBER *whichWindow*)
This function returns a value representing one of the NucLear MAC Colors windows.

You can use the window value to move, hide and select a Colors window. The *whichWindow* argument must be in the range of zero to the total number of Colors windows less one. You can obtain the total number of Colors windows with the `CountColorWindows` function.

T_WINDOW **`ControlsWindow`**
This function returns a value representing the NucLear MAC Controls window. You can use the window value to move, hide and select the Controls window.

T_NUMBER **`CountColorsWindows`**
This function returns the total number of Colors windows at the time of the call. There is always at least one Colors window.

T_STRING **`GetWindowName`**(T_WINDOW *theWindow*)
Given a parameter representing a NucLear MAC window, the `GetWindowName` function returns a string containing the current title of the window. If the window contents are stored in a file, then the name returned is also the name of the disk file.

`MoveWindow`(T_WINDOW *theWindow*, T_NUMBER *x_loc*, T_NUMBER *y_loc*)
Given a window value in *theWindow*, the `MoveWindow` function moves the window so that its top, left corner has the screen coordinates given by the *y_loc* and *x_loc* parameters. Keep in mind that the origin for screen coordinates is the top, left corner of the screen containing the menu bar. Also be aware that x coordinates increase from left to right and y coordinates increase from top to bottom.

T_WINDOW **`ProgressWindow`**
This function returns a value representing the NucLear MAC Progress window. You can use the window value to move, hide and select the Progress window.

T_WINDOW **`ToolsWindow`**
This function returns a value representing the NucLear MAC Tools window. You can use the window value to move, hide and select the Tools window.

`ResizeWindow`(T_WINDOW *theWindow*, T_NUMBER *x_size*,
T_NUMBER *y_size*)
Given a window value in *theWindow*, the `ResizeWindow` function resizes the window so that its active area is *x_size* pixels wide and *y_size* pixels high. The top, left corner of the window stays in the same position on the screen.

`SelectWindow`(T_WINDOW *theWindow*)
The `SelectWindow` function brings the window specified by *theWindow* to the front and makes it the active window. Palette windows are only brought to the front if an image window is active.

T_NUMBER **`WindowHeight`**(T_WINDOW *theWindow*)
`WindowHeight` returns the current height in pixels of the window specified by *theWindow*.

T_NUMBER **WindowWidth**(T_WINDOW *theWindow*)
WindowWidth returns the current width in pixels of the window specified by *theWindow*.

T_NUMBER **WindowXPosition**(T_WINDOW *theWindow*)
WindowXPosition returns the current horizontal screen coordinate of the top, left corner of window specified by *theWindow*. The origin for screen coordinates is the top, left corner of the screen containing the menu bar. Horizontal coordinates increase from left to right.

T_NUMBER **WindowYPosition**(T_WINDOW *theWindow*)
WindowYPosition returns the current vertical screen coordinate of the top, left corner of window specified by *theWindow*. The origin for screen coordinates is the top, left corner of the screen containing the menu bar. Vertical coordinates increase from top to bottom.

8.8.3 Program Information

Functions in this section deal with getting and setting information contained in the NucLear MAC preferences file.

DeleteAllPatientRecords
DeleteAllPatientRecords removes all patient records from the NucLear MAC program.

ExportPatientRecords(T_STRING *theFormat* T_STRING *fileName*)
Using the patient record format defined by *theFormat*, ExportPatientRecords exports all current patient records to the file defined by *fileName*.

T_ANY **GetProgramInfo**(T_NUMBER *valueIndex*,T_NUMBER *valueType*)
GetProgramInfo returns an informational value stored in the NucLear MAC preferences file. *valueIndex* is the index number of the information to return. *valueType* is the type of information the call should retrieve. Use the following for *valueType*:

0 Causes return value of one (1) if field exists, zero (0) if not
1 Returns a string field
2 Returns a two-byte integer field
3 Returns a four-byte integer field
4 Returns a four-byte real number field

ImportPatientRecords(T_STRING *theFormat*, T_STRING *fileName*,
T_NUMBER *replace*,T_NUMBER *asked*)
ImportPatientRecords attempts to input all patient records in the file named *fileName* using the patient record format defined by *theFormat*. *asked* and *replace* control whether the user is asked if it is OK to replace records with the same names. Use this table to control the behavior when duplicate record names exist:

replace	*asked*	Action
any value	0	Ask user for action (delete existing duplicates or skip new ones).

```
0         1      Always skip new duplicate records.
1         1      Always delete existing duplicate records.
```

PutProgramArg(T_NUMBER *theArg*)
PutProgramArg transfers a copy of the field value specified by *theArg* from the default program area into the macros argument list. With this function a macro can use default parameters from a particular NucLear MAC system rather than explicitly specifying all values.

8.8.4 Active Image Frame

These functions control cines, mouse tracking, and the selection of the active frame of a study window.

ChangeFrame(T_IMAGE *theImage*, T_NUMBER *new_frame*)
ChangeFrame changes the active frame of the image window specified by *theImage* to the frame number specified by *new_frame*. In the case of a movie window, the front-most frame is changed to *new_frame*. For an all-frames window, *new_frame* becomes the highlighted frame and is made visible by scrolling, if necessary. The *new_frame* parameter must be at least one and less than or equal to the number of image frames, which is returned by FrameCount (see below).

ClearMouseTrack
ClearMouseTrack clears the list of image movie windows used by the MouseTrackSome function.

T_NUMBER **CurrentFrame**(T_IMAGE *theImage*)
CurrentFrame returns the number of the current active frame of the image window specified by *theImage*. This is the front-most frame of a movie window, or the highlighted frame of an all-frames window.

T_NUMBER **FrameCount**(T_IMAGE *theImage*)
FrameCount returns the number of frames present in the image window specified by *theImage*.

FullSpeed
FullSpeed causes the Macintosh computer to devote almost all of its processing power to running cines. You can only exit this state by clicking the mouse button.

Halt(T_IMAGE *theImage*)
The Halt function stops the cine of the movie window specified by *theImage*.

HaltAll
HaltAll stops all cines that are currently running.

InMouseTrack(T_IMAGE *theImage*)
InMouseTrack sets a flag for the movie image window specified by *theImage* so that it will be included when the MouseTrackSome command is issued.

MouseTrack(T_IMAGE *theImage*)
The MouseTrack function enters a mode where moving the mouse changes the active frame of the movie image window specified by *theImage*.

MouseTrackAll
MouseTrackAll causes the active frames of all movie image windows to change with the motion of the mouse.

MouseTrackSome
MouseTrackSome causes the active frames of certain movie image windows to change with the motion of the mouse. The movie windows are included with the InMouseTrack function. Initially, the ClearMouseTrack function should be called to remove any unwanted movies from the tracking.

T_NUMBER **PromptForFrame**(T_IMAGE *theImage*, T_STRING *thePrompt*)
PromptForFrame asks the macro user to select a frame of the image window specified by *theImage*. The prompt given is specified by *thePrompt*. *theImage* can be either a movie or an all-frames window.

Run(T_IMAGE *theImage*)
Run starts a cine in the movie image window specified by *theImage*.

RunAll
RunAll starts cines in all movie image windows.

8.8.5 Study Information

Functions in this section deal with getting and setting information about study windows.

ClearFramesSubset(T_IMAGE *theImage*, T_NUMBER *startFrame*,
T_NUMBER *endFrame*)
ClearFramesSubset clears a range of the flags that indicate which image frames are to be extracted by the FramesSubset function. The command applies to the image specified by *theImage* and the range of frames defined by *startFrame* and *endFrame*.

T_IMAGE **FrontImage**

Provided the front-most NucLear MAC window is an image window, FrontImage returns a representation of that window. Before calling this function, you should call the FrontIsImage function to verify that the front window is an image.

T_NUMBER **FrontIsImage**

The FrontIsImage functions returns the value one if the front-most NucLear MAC window is an image window, and zero otherwise.

T_NUMBER **GetFrameTime**(T_IMAGE *theImage*, T_NUMBER *frame*)
For the image given by *theImage*, GetFrameTime returns the time associated with the specified frame. This time is in milliseconds. For most studies, GetFrameTime

indicates how long it took to acquire the particular frame. For gated studies, however, `GetFrameTime` returns the time for the portion of the heartbeat held by the given frame.

T_NUMBER **`GetFramePause`**(T_IMAGE *theImage*, T_NUMBER *frame*)
For the image given by *theImage*, `GetFramePause` returns the pause associated with the specified frame. This pause is in milliseconds. `GetFramePause` indicates the sum of the pauses occurring at the start of or during the acquisition of the frame.

T_NUMBER **`GetImageType`**(T_IMAGE *theImage*)
`GetImageType` returns the type of study of the image window represented by *theImage*. At this writing, the following types are defined:

0 STATIC
1 SEQUENCE
2 DYNAMIC
3 GATED
4 SPECT
5 PLANAR DURING SPECT
6 WHOLE BODY
7 GATED SPECT
8 RECONSTRUCTION
9 BULLSEYE
10 GATED RECONSTRUCTION

T_STRING **`GetPatientBirthday`**(T_IMAGE *theImage*)
For the image given by *theImage*, `GetPatientBirthday` returns a string containing the date of birth of the patient associated with the image. If the date was not entered in the image, an empty string is returned.

T_STRING **`GetPatientID`**(T_IMAGE *theImage*)
For the image given by *theImage*, `GetPatientID` returns a string containing the identification number of the patient associated with the image. If the number was not entered in the image, then an empty string is returned.

T_STRING **`GetPatientName`**(T_IMAGE *theImage*)
For the image given by *theImage*, `GetPatientName` returns a string containing the name of the patient associated with the image. If the name was not entered in the image, then an empty string is returned.

T_STRING **`GetStudyDateString`**(T_IMAGE *theImage*)
`GetStudyDateString` returns a string containing the acquisition date of the image, if any. The date format is determined by the operating system. This is user-defineable on newer systems. If no study date exists, then an empty string is returned.

T_NUMBER **`GetStudyDateTime`**(T_IMAGE *theImage*)
`GetStudyDateTime` returns a number containing the acquisition date and time of the image in Macintosh format. This is a number representing seconds elapsed since midnight, January 1, 1904. If no study date exists, then zero is returned.

T_ANY **GetStudyInfo**(T_IMAGE *theImage*,T_NUMBER *valueIndex*,
T_NUMBER *valueType*)
GetStudyInfo returns an informational value stored with *theImage*. *valueIndex* is the index number of the information to return. *valueType* is the type of information the call should retrieve. Use the following for *valueType*:

0 Causes return value of one (1) if field exists, zero (0) if not
1 Returns a string field
2 Returns a two-byte integer field
3 Returns a four-byte integer field
4 Returns a four-byte real number field

T_STRING **GetStudyTimeString**(T_IMAGE *theImage*)
GetStudyTimeString returns a string containing the acquisition time of the image, if any. The time format is determined by the operating system. This is user-defineable on newer systems. If no study time exists, then an empty string is returned.

T_NUMBER **ImageHeight**(T_IMAGE *theImage*)
ImageHeight returns the vertical resolution of the image window given by *theImage*. Note that image resolution need not be the same as the image window size. When these differ, the image is scaled before it is shown in a window on the screen.

T_NUMBER **ImageWidth**(T_IMAGE *theImage*)
ImageWidth returns the horizontal resolution of the image window given by *theImage*. Note that image resolution need not be the same as the image window size. When these differ, the image is scaled before it is shown in a window on the screen.

T_NUMBER **IsDecayCorrected**(T_IMAGE *theImage*)
Given an image window specified by *theImage*, IsDecayCorrected returns a value of one if the image has been decay corrected, and zero if not.

T_NUMBER **IsFloodCorrected**(T_IMAGE *theImage*)
Given an image window specified by *theImage*, IsFloodCorrected returns a value of one if the image has been flood corrected, and zero if not.

T_NUMBER **IsMovieWindow**(T_IMAGE *theImage*)
Given an image window specified by *theImage*, IsMovieWindow returns a value of one if the window is a movie window, and zero if it is an all-frames window.

T_IMAGE **PromptForImage**(T_STRING *thePrompt*)
PromptForImage prompts the macro user to select an image window. If the user selects an image, its representation is returned. Otherwise, the macro is canceled. The string given by *thePrompt* is shown to the user.

SetFrameFlags(T_IMAGE *theImage*, T_NUMBER *startFrame*,
T_NUMBER *endFrame*, T_NUMBER *whichFlag*, T_NUMBER *setFlag*)
SetFrameFlags sets a range of the flags that indicate which image frames are to be used by a particular function. The command applies to the image specified by *theImage* and the range of frames defined by *startFrame* and *endFrame*. If *setFlag* is

non-zero, then the flag is set, otherwise the flag is cleared. Use one of the following for *whichFlag*:

1 include frame in new composite
2 include frame in graphing function
3 include frame in frames-subset function

SetFramesSubset(T_IMAGE *theImage*, T_NUMBER *startFrame*,
T_NUMBER *endFrame*)
SetFramesSubset sets a range of the flags that indicate which image frames are to be extracted by the FramesSubset function. The command applies to the image specified by *theImage* and the range of frames defined by *startFrame* and *endFrame*.

8.8.6 Image Processing and Display

The following functions act on one or more image windows to generate a processed result. Because these functions all have corresponding menu commands, you will find more complete descriptions of their actions elsewhere in the manual.

T_WINDOW **AllAnglesCoronal**(T_IMAGE *theImage*)
AllAnglesCoronal returns an all-angles window based on the image window represented by *theImage*. *theImage* is assumed to be a coronal or short axis view.

T_WINDOW **AllAnglesTransverse**(T_IMAGE *theImage*)
AllAnglesTransverse returns an all angles window based on the image window represented by *theImage*. *theImage* is assumed to be a transverse or horizontal long axis view.

T_IMAGE **AllFramesSnapshot**(T_IMAGE *theImage*)
Given an all-angles image window, AllFramesSnapshot returns an image window with a single frame incorporating the visible area of *theImage*.

T_IMAGE **AllFramesWindow**(T_IMAGE *theImage*)
Given a movie image window, AllFramesWindow produces an all-frames window of the same image data.

T_ARRAY **AnalyzePhase**(T_IMAGE *theImage*,T_NUMBER *threshold*,
T_NUMBER *binDegrees*, T_NUMBER *start*)
Use the AnalyzePhase function to perform temporal Fourier analysis on *theImage*. The other arguments correspond to those in the Phase Settings dialog. Respectively, they are the threshold as a percent of maximum, number of degrees per histogram bin, and starting angle in degrees for the histogram The array returned by AnalyzePhase has the following values:

Element	Type and Value
0	T_IMAGE—Phase images
1	T_GRAPH—Graph of phase histogram

T_IMAGE **AnalyzeSlopes**(T_IMAGE *theImage*)
Use the AnalyzeSlopes function to perform point-by-point temporal slope analysis

on *theImage*. `AnalyzeSlopes` returns an image window containing minimum and maximum slope images, as well as a maximum-slope-frame image.

T_IMAGE **`Append`**(T_IMAGE *firstImage*, T_IMAGE *secondImage*)
Given two image windows having the same resolutions, `Append` creates a new image window that has the images of the *firstImage* parameter followed by the images of the *secondImage* parameter.

`ApplyShifts`(T_IMAGE *shiftedImage*, T_IMAGE *theImage*)
`ApplyShifts` shifts the pixels of *theImage* by the shifts that have already been applied to the pixels in *shiftedImage*. The number of frames in *theImage* should be an integer multiple of the number of frames in *shiftedImage*. Thus the shifts in each frame of *shiftedImage* can be applied to a number of frames in *theImage*. This facilitates motion correction in gated-SPECT images.

T_IMAGE **`BottomView`**(T_IMAGE *theImage*)
Given an image window, presumably a reconstructed data set, `BottomView` returns a new image window that views the original data from the bottom, as if the top of the original image had been rotated away 90°.

T_IMAGE **`ByteScale`**(T_IMAGE *theImage*)
`ByteScale` takes a word-mode image window and returns a byte-scaled version of the image data in a new window. If the image parameter is already byte-mode, then a duplicate of the original window is made.

T_IMAGE **`CLUTFilter`**(T_IMAGE *theImage*)
`CLUTFilter` takes the image data of the image parameter and passes it through the current color lookup table to generate a new image window. When the source image is word-mode, a byte-scaled version is used.

T_IMAGE **`Complement`**(T_IMAGE *theImage*)
`Complement` performs a logical complement of all the bits in the data making up the source image window.

T_IMAGE **`CoronalToSagittal`**(T_IMAGE *theImage*)
`CoronalToSagittal` uses an image window, presumed to be a coronal view, to generate a sagittal view. This also transforms a short axis view to vertical long axis.

T_IMAGE **`CoronalToTransverse`**(T_IMAGE *theImage*)
`CoronalToTransverse` uses an image window, presumed to be a coronal view, to generate a transverse view. This also transforms a short axis view to horizontal long axis.

T_IMAGE **`DecayCorrect`**(T_IMAGE *theImage*, T_NUMBER *half_life*)
Given a source image named by *theImage* and an isotope half-life in hours, `DecayCorrect` returns a decay-corrected image. Note that the source image must have timing information associated with it.

T_IMAGE **Double**(T_IMAGE *theImage*)
The Double function doubles the resolution of a source image through interpolation. Note that the resolution of the resulting image is double the source resolution *less one*.

T_IMAGE **EquationFilter**(T_IMAGE *theImage*, T_STRING *theFilter*)
EquationFilter returns an image that is the equation-filtered source image. The equation filter is specified by the string *theFilter*, which must have been previously defined in the dialog presented by the Equation Filter menu command.

T_IMAGE **ExtractCube**(T_IMAGE *theImage*)
The ExtractCube function effectively pulls a cube of image data from an image window. The dimensions of the cube are the length of the y axis in the source image. The center of the cube coincides with the origin of the axes in the source image.

T_IMAGE **ExtractSphere**(T_IMAGE *theImage*)
The ExtractSphere function effectively pulls a sphere of image data from an image window; areas outside the sphere are masked to zeros. The radius of the sphere is the length of the y axis in the source image. The center of the sphere coincides with the origin of the axes in the source image.

T_IMAGE **FlipHorizontal**(T_IMAGE *theImage*)
FlipHorizontal exchanges the left and right sides of an image. That is, a mirroring occurs about a vertical line drawn midway through the image.

T_IMAGE **FlipVertical**(T_IMAGE *theImage*)
FlipVertical exchanges the top and bottom portions of an image. That is, a mirroring occurs about a horizontal line drawn midway through the image.

T_IMAGE **FloodCorrect**(T_IMAGE *theImage*, T_IMAGE *floodImage*)
Given a source image named by *theImage* and a flood matrix of the same resolution, FloodCorrect applies an inverse of the flood image to the source to generate a corrected image.

T_IMAGE **FrameNormalize**(T_IMAGE *theImage*)
FrameNormalize scales the frames of the source image so that the frames of the resulting image all have the same maximum value. If the source image is byte-mode, then the resulting image frames have maximum values of 255. For word-mode source frames, the resulting frames are scaled to each have the maximum pixel value of the entire source image set.

T_IMAGE **FramesSubset**(T_IMAGE *theImage*)
Using the flags manipulated by the ClearFramesSubset and SetFramesSubset functions, FramesSubset makes an image window having a subset of the frames contained in the source window.

T_IMAGE **Halve**(T_IMAGE *theImage*)
By averaging groups of four pixels, the Halve function creates an image having half the resolution of the source image.

T_IMAGE **Interleave**(T_IMAGE *theImage*, T_NUMBER *grouping*)
Use the Interleave function to rearrange the frames of a source image. The *grouping* parameter determines the ordering of the frames. Suppose there are (n * *grouping*) source frames. Then the resulting frame order is 1, *grouping* + 1, 2 * *grouping* + 1,…, (n - 1) * *grouping* + 1, followed by 2, *grouping* + 2, 2 * *grouping* + 2,…, (n - 1) * *grouping* + 2,…, and ending with *grouping*, 2 * *grouping*,…, n * *grouping*.

T_ARRAY **InterleaveSeparate**(T_IMAGE *theImage*, T_NUMBER *grouping*)
Use the InterleaveSeparate function to rearrange the frames of a source image and split them into multiple image windows. The *grouping* parameter determines the ordering of the frames. Suppose there are (n * *grouping*) source frames. An array of *grouping* images is returned. The first image window has a frame order relative to the source image of 1, *grouping* + 1, 2 * *grouping* + 1,…, (n - 1) * *grouping* + 1. The second image window has a frame order of 2, *grouping* + 2, 2 * *grouping* + 2,…, (n - 1) * *grouping* + 2, and the last image window has a frame order of *grouping*, 2 * *grouping*,…, n * *grouping*.

Interpolate(T_IMAGE *theImage*, T_NUMBER *turnOn*)
The Interpolate function enables and disables interpolation in the display of *theImage*. Pass a value of one for *turnOn* to enable interpolation, and a value of zero to disable interpolation.

T_IMAGE **Invert**(T_IMAGE *theImage*)
Use the Invert function to exchange the values of high and low pixels. For byte-mode images, each pixel value is changed to (255 - old_pixel_value). For word-mode images, each pixel value becomes (max_image_pixel_value - old_pixel_value).

T_IMAGE **LeftView**(T_IMAGE *theImage*)
Given an image window, presumably a reconstructed data set, LeftView returns a new image window that views the original data from the left, as if the right side of the original image had been rotated away 90°.

MatchCropping(T_IMAGE *firstImage*, T_IMAGE *secondImage*)
If possible, the MatchCropping function changes the window size and viewing area of *firstImage* so that its pixel size matches that of *secondImage*. This allows a direct size comparison of the two images.

MatchScaling(T_IMAGE *firstImage*, T_IMAGE *secondImage*)
If possible, the MatchScaling function changes the brightness of *firstImage* so that it is on the same scale as *secondImage*. This allows a direct comparison of the brightnesses of the two images.

T_IMAGE **MovieWindow**(T_IMAGE *theImage*)
Given an all-frames image window, MovieWindow produces a movie window of the same image data.

T_WINDOW **NewAllViewsWindow**(T_IMAGE *stressImage*, T_IMAGE *restImage*,
T_NUMBER *fromCoronals*, T_NUMBER *compression*, T_NUMBER *slices*)
Given stress and rest image windows, NewAllViewsWindow produces an all-views

window of the image data. If the *fromCoronals* parameter is non-zero, the source images are assumed to be of coronal / short axis orientation. Otherwise, the source images are assumed to be of transverse / horizontal long axis orientation. The *compression* parameter is used to reframe the images prior to display. A value of one causes no compression. A value of two causes every pair of images to be summed, etc. The *slices* parameter determines how many slices will be displayed horizontally in the all-views window.

T_IMAGE **NewComposite**(T_IMAGE *theImage*)
NewComposite returns a single-frame image that is a sum of all the frames of the source image. The maximum resulting pixel value is 65535.

T_IMAGE **NewSubWindow**(T_REGION *boundsRegion*)
Given a rectangular region, NewSubWindow extracts the area of the region from all frames of the region's home image and puts the result in a new image window.

PhotoMode
The PhotoMode function hides the menu bar, the cursor, the desktop and the windows belonging to applications other than NucLear MAC. This function stays active until the user clicks the mouse or presses a key.

T_IMAGE **PurgeFrames**(T_IMAGE *theImage*)
The PurgeFrames function creates a copy of the source image set with blank frames removed. Blank frames are defined as those having no counts and no acquisition time.

T_IMAGE **Reconstruct**(T_IMAGE *theImage*,T_STRING *reconProtocol*)
The Reconstruct function reconstructs the SPECT projections in *theImage* using the SPECT reconstruction protocol named by *reconProtocol*.

T_IMAGE **Reframe**(T_IMAGE *theImage*, T_NUMBER *startFrame*,
T_NUMBER *endFrame*, T_NUMBER *group_size*,
T_NUMBER *extra*, T_NUMBER *doPrepend*,
T_NUMBER *doAppend*)
The Reframe command is used to sum groups of frames from a source image and place the results in a new image window. *startFrame* and *endFrame* determine the range of frames over which the summing takes place. *group_size* indicates how many source frames should be summed to make one destination frame. *extra* chooses whether remainder frames are ignored (0), appended (1), or rescaled (2). Set *doPrepend* to one if you want any frames prior to *startFrame* prepended to the result image, otherwise pass zero. Set *doAppend* to one if you want any frames after endFrame appended to the result image, otherwise pass zero.

T_IMAGE **ResliceFromView**(T_IMAGE *theImage*)
ResliceFromView takes the source image, which is assumed to be a reconstructed data set, and creates a new set of image slices based on the view angle set by the SetViewAngle function.

T_IMAGE **Reverse**(T_IMAGE *theImage*)
The Reverse function creates a copy of the source image set with the frame order in reverse.

ReverseImageScale(T_IMAGE *theImage*,T_NUMBER *reverse*)
The ReverseImageScale function controls whether an image is displayed with a standard or a reversed color table. If the *reverse* parameter is non-zero, the image scale is reversed if it is not already. If the *reverse* parameter is zero, the image scale is set to normal if it is not already. This function is equivalent to operating the reverse button in the Controls window—not the reverse button in a Colors window.

T_IMAGE **RightView**(T_IMAGE *theImage*)
Given an image window, presumably a reconstructed data set, RightView returns a new image window that views the original data from the right, as if the left side of the original image had been rotated away 90°.

T_IMAGE **Rotate**(T_IMAGE *theImage*)
Rotate creates a new image window containing a shifted and rotated copy of the source image. If interactive behavior is used, then Rotate waits for the user to enter rotation commands. The commands are:

command	result
move mouse	change to higher or lower image frame
hold down option key and move mouse	rotate current frame according to rotation preferences
press arrow key	shift frame in arrow direction according to rotation prefs
type 'a'	apply rotation and shift of current frame to all frames
press 'delete'	undo rotation and shift in current frame
click mouse	exit rotation command

The other arguments for this function are provided by the PutArg macro. The arguments are:

name	identifier	type	code	range
pixel_size_ratio	$00190380	real	4	0 < x ≤ 100
resolution_ratio	$00190384	real	4	0 < x ≤ 100
nudge_inc	$00190388	real	4	0 < x ≤ 100
nudge_limit	$0019038c	real	4	0 ≤ x ≤ 100
rotate_inc	$00190390	real	4	0 < x
non_interactive	$00190394	short int	2	0 or 1

The above arguments correspond to those that are entered in the Rotate Preferences dialog, except for non_interactive:

name	value	meaning
non_interactive	0	user interaction sets rotation and shifts
	1	rotation and shifts taken from SetRotateAmounts calls

T_IMAGE **RotateLeft**(T_IMAGE *theImage*)
RotateLeft creates a new image window containing the source image rotated counter-clockwise 90° about its center.

T_IMAGE **RotateRight**(T_IMAGE *theImage*)
RotateRight creates a new image window containing the source image rotated clockwise 90° about its center.

T_ARRAY **Separate**(T_IMAGE *theImage*, T_NUMBER *grouping*)
Use the Separate function to split the frames of a source image into multiple image windows. The *grouping* parameter determines the number of frames in the resulting images. Suppose there are (n * *grouping*) source frames. An array of n images is returned. The first image window has a frame order relative to the source image of 1, 2, 3,…, *grouping*. The second image window has a frame order of *grouping* + 1, *grouping* + 2, *grouping* + 3,…, 2 * *grouping*, etc.

SetRotateAmounts(T_NUMBER *startFrame*,T_NUMBER *endFrame*,
T_NUMBER *x_shift*,T_NUMBER *y_shift*,T_NUMBER *rotation*)
Use the SetRotateAmounts function to enter shifts and rotations to be used for a subsequent non-interactive call to Rotate. The frame parameters are zero-based, so the following should be true: 0 ≤ *startFrame* ≤ *endFrame* < (image frames)

T_IMAGE **SetViewAngle**(T_IMAGE *theImage*)
SetViewAngle projects the source image onto a plane from arbitrary angles that the user sets with the mouse. It returns a single-frame image containing the last projection before the user ended the rotation. Use the ResliceFromView function to slice the source image set from the new orientation.

T_IMAGE **SpatialConvolution**(T_IMAGE *theImage*, T_STRING *theFilter*)
SpatialConvolution returns an image that is the spatially convoluted source image. The convolution filter is specified by the string *theFilter*, which must have been previously defined.

T_IMAGE **SplitDual**(T_IMAGE *theImage*)
Use the SplitDual command to split a dual-energy image such that the different energy images appear in different frames of the resulting image window.

T_IMAGE **SubtractAsBackground**(T_REGION *theRegion*,
T_NUMBER *frame*, T_NUMBER *order*,
T_NUMBER *reduceBy*)
SubtractAsBackground subtracts background from the home image of *theRegion*. The order parameter determines whether the background region is modeled as constant (0), linear (1), or quadratic (2). *reduceBy* reduces the subtraction by the given percentage. When *order* is zero, all frames of the home image have background subtracted. A higher value of *order*, however, causes background to be subtracted only from the frame specified.

T_IMAGE **SubtractFromAll**(T_IMAGE *theImage*, T_IMAGE *subtractImage*)
SubtractFromAll subtracts the active frame of *subtractImage* from all frames of *theImage*, and returns the result. Both arguments must have the same resolution.

T_IMAGE **SubtractWeighted**(T_IMAGE *theImage*, T_IMAGE *subtractImage*)
The SubtractWeighted function performs an interactive subtraction of

subtractImage from *theImage*. Only the active frames of the two images are used. Both arguments must have the same resolution.

T_IMAGE **`TemporalConvolution`**(T_IMAGE *theImage*, T_STRING *theFilter*)
`TemporalConvolution` returns an image that is the temporally convoluted source image. The convolution filter is specified by the string *theFilter*, which must have been previously defined.

T_IMAGE **`ThreeDSmooth`**(T_IMAGE *theImage*)
`ThreeDSmooth` applies an unweighted three dimensional smooth to the multi-framed source image. Groups of 27 pixels are averaged.

T_IMAGE **`TopView`**(T_IMAGE *theImage*)
Given an image window, presumably a reconstructed data set, `TopView` returns a new image window that views the original data from the top, as if the bottom of the original image had been rotated away 90°.

T_IMAGE **`TransverseToCoronal`**(T_IMAGE *theImage*)
`TransverseToCoronal` uses an image window, presumed to be a transverse view, to generate a coronal view. This also transforms a horizontal long axis view to a short axis.

T_IMAGE **`TransverseToSagittal`**(T_IMAGE *theImage*)
`TransverseToSagittal` uses an image window, presumed to be a transverse view, to generate a sagittal view. This also transforms a horizontal long axis view to a vertical long axis view.

T_IMAGE **`VolumeRender`**(T_IMAGE *theImage*)
Given a reconstructed data set, `VolumeRender` returns a set of 32 volume rendered projections of the source data.

T_IMAGE **`Washout`**(T_IMAGE *firstImage*, T_IMAGE *secondImage*)
`Washout` computes a washout image based on *firstImage* and *secondImage*.

T_IMAGE **`XConcat`**(T_IMAGE *firstImage*, T_IMAGE *secondImage*)
The `XConcat` function returns an image that is a horizontal concatenation of *firstImage* and *secondImage*. Neither the source resolutions nor the number of source frames need match.

T_IMAGE **`YConcat`**(T_IMAGE *firstImage*, T_IMAGE *secondImage*)
The `YConcat` function returns an image that is a vertical concatenation of *firstImage* and *secondImage*. Neither the source resolutions nor the number of source frames need match.

8.8.7 Graphs

The following functions are used to create and get information about various types of graphs.

ClearCurveSymbols
ClearCurveSymbols deletes all string symbols that are associated with curves. This call undoes all previous calls to SetCurveSymbol. Use this call prior to defining new symbols for a curve math operation.

T_GRAPH **ColumnAverages**(T_REGION *theRegion*, T_NUMBER *frame*)
ColumnAverages returns a graph of the column averages of a rectangular ROI given by *theRegion* in its home image and in the frame given by *frame*.

T_GRAPH **ColumnSums**(T_REGION *theRegion*, T_NUMBER *frame*)
ColumnSums returns a graph of the column sums of a rectangular ROI given by *theRegion* in its home image and in the frame given by *frame*.

T_GRAPH **ConvolveCurves**(T_GRAPH *firstGraph*, T_NUMBER *firstCurve*,
T_NUMBER *firstStart*, T_GRAPH *secondGraph*,
T_NUMBER *secondCurve*, T_NUMBER *secondStart*,
T_NUMBER *nPoints*)
ConvolveCurves returns a graph containing a curve that is the convolution of the two curves specified by *firstGraph*, *firstCurve*, *secondGraph* and *secondCurve*. In addition, the starting points of the two curves are specified by *firstStart* and *secondStart* respectively. The number of points to convolve is given by *nPoints*.

T_NUMBER **CurveMath**(T_GRAPH *theGraph*, T_GRAPH *xGraph*,
T_NUMBER *xCurve*)
CurveMath performs calculations on one or more graph curves to generate a new curve in the graph specified by *theGraph*. The x-axis values are taken from the graph and curve specified by *xGraph* and *xCurve* respectively. The other arguments for this function are provided by the PutArg macro. The arguments are:

name	identifier	type	code	range
x_symbol	$001935b0	string	1	27 characters max
y_axis_title	$001935d0	string	1	27 characters max
y_axis_units	$001935c0	string	1	27 characters max
equation	$001935e0	string	1	255 characters max
use_y_units	$00193560	short int	2	0 or 1
x_mapping	$00193540	short int	2	0 or 1
x_range	$00193530	short int	2	0 or 1
point_type	$00193550	short int	2	0 ≤ x ≤ 6
point_connect	$00193520	short int	2	0 or 1
second_axis	$00193510	short int	2	0 or 1
x_start_val	$00193570	real	4	x < x_end_val
x_end_val	$00193580	real	4	x > x_start_val
x_mul_val	$00193590	real	4	x > 0
x_add_val	$001935a0	real	4	real numbers

The above arguments correspond to those that are entered in the Curve Math dialog:

name	value	meaning
use_y_units	0	take y-axis units from x curve
	1	take y-axis units from y_axis_units
x_mapping	0	do not remap x axis

```
              1      remap x axis using x_mul_val and x_add_val
x_range       0      do not limit the x-axis range
              1      limit x-axis range from x_start_val to x_end_val
point_type    0      circle
              1      +
              2      x
              3      triangle
              4      square
              5      dot
              6      bar-graph
point_connect 0      do not draw lines between points
              1      draw lines between points
second_axis   0      use primary axis
              1      use secondary axis if primary axis exists
```

T_GRAPH **DeconvolveCurves**(T_GRAPH *firstGraph*, T_NUMBER *firstCurve*,
T_NUMBER *firstStart*, T_GRAPH *secondGraph*,
T_NUMBER *secondCurve*, T_NUMBER *secondStart*,
T_NUMBER *nPoints*)

`DeconvolveCurves` returns a graph containing a curve that is the deconvolution of the two curves specified by *firstGraph*, *firstCurve*, *secondGraph* and *secondCurve*. In addition, the starting points of the two curves are specified by *firstStart* and *secondStart* respectively. The number of points to deconvolve is given by *nPoints*.

T_NUMBER **EquationPlot**(T_GRAPH *theGraph*)

`EquationPlot` generates a new curve from an equation and places it in the graph specified by *theGraph*. The other arguments for this function are provided by the PutArg macro. The arguments are:

name	identifier	type	code	range
y_axis_title	$00170010	string	1	27 characters max
y_axis_units	$00170020	string	1	27 characters max
x_var_name	$00170030	string	1	9 characters max
x_var_units	$00170038	string	1	27 characters max
y_var_name	$00170070	string	1	9 characters max
equation	$00170090	string	1	255 characters max
parm_names	$00170200+8i	string	1	9 characters max
integrate	$001700c0	short int	2	0 or 1
plot_every	$00170100	short int	2	0 < x ≤ 9999
number_parms	$001700f0	short int	2	0 ≤ x ≤ 32
point_type_int	$00170110	short int	2	0 ≤ x ≤ 6
point_connect_int	$001700d0	short int	2	0 or 1
second_axis	$001700b0	short int	2	0 or 1
x_start_val	$00170040	real	4	x < x_end_val
x_end_val	$00170050	real	4	x > x_start_val
x_step_val	$00170060	real	4	x > 0
y_start_val	$00170080	real	4	real numbers
parm_values	$00170201+8i	real	4	real numbers

The value i in the identifiers for parm_names and parm_values ranges from zero to number_parms less one. The above arguments correspond to those that are entered in the Equation Plotting dialog:

name	value	meaning
integrate	0	do not integrate
	1	integrate y values using y_start_val, x_step_val
point_type_int	0	circle
	1	+
	2	x
	3	triangle
	4	square
	5	dot
	6	bar-graph
point_connect_int	0	do not draw lines between points
	1	draw lines between points
second_axis	0	use primary axis
	1	use secondary axis if primary axis exists

T_GRAPH **FramesPlot**(T_IMAGE *theImage*, T_NUMBER *frame*)
FramesPlot attempts to make a graph of time activity curves of the image specified by *theImage*. The number of curves depends on the number of ROIs selected in the frame given by the *frame* parameter. There must be at least one selected ROI in the indicated frame.

T_GRAPH **FrontGraph**
Provided the front-most NucLear MAC window is a graph window, FrontGraph returns a representation of that window. Before calling this function, you should call the FrontIsGraph function to verify that the front window is a graph.

T_NUMBER **FrontIsGraph**
The FrontIsGraph functions returns the value one if the front-most NucLear MAC window is a graph window, and zero otherwise.

Histogram(T_REGION *theRegion*, T_NUMBER *frame*)
Given an ROI described by *theRegion*, Histogram creates a new window containing a histogram of the ROI in its home image and in the frame specified by the *frame* parameter.

T_NUMBER **IntegrateCurve**(T_GRAPH *theGraph*, T_NUMBER *theCurve*,
T_NUMBER *startPoint*, T_NUMBER *nPoints*)
IntegrateCurve returns the area of the specified portion of the curve given by *theGraph* and *theCurve*. The trapezoid method is used. *startPoint* is the index of the first point to integrate, and *nPoints* specifies how many points to integrate.

T_GRAPH **NewGraph**(T_STRING *title*, T_STRING *xName*, T_STRING *xUnits*)
NewGraph returns a new, empty graph with title, x-variable name, and x-variable units as specified.

T_NUMBER **NewVariable**(T_GRAPH *theGraph*, T_ARRAY *xVals*,
T_ARRAY *yVals*, T_STRING *yName*, T_STRING *yUnits*,
T_NUMBER *points*, T_NUMBER *connect*, T_NUMBER *secondAxis*)
NewVariable adds a new curve to the graph specified by *theGraph* and returns the number of the curve. *xVals* and *yVals* are numerical arrays of the x-values and y-values to be graphed. If *connect* is non-zero, then the graph points are connected by

lines, otherwise they are not connected. If *secondAxis* is non-zero and a primary axis already exists, then the graph points are scaled to the secondary y-axis, otherwise they are scaled to the primary y-axis. *points* determines the symbol used for drawing the points for the curve. Use one of the following:

0 circle
1 +
2 x
3 triangle
4 square
5 dot
6 bar-graph

T_GRAPH **`PromptForGraph`**(T_STRING *thePrompt*)
`PromptForGraph` prompts the macro user to select a graph window. If the user selects a graph, its representation is returned. Otherwise, the macro is canceled. The string given by *thePrompt* is shown to the user.

`PutGraphMessage`(T_GRAPH *theGraph*, T_STRING *theMessage*)
`PutGraphMessage` displays the text message given by *theMessage* in the message field of the graph given by *theGraph*.

T_NUMBER **`RegionToVariable`**(T_GRAPH *theGraph*, T_ANY *region*)
`RegionToVariable` adds a new curve to the graph specified by *theGraph* and returns the number of the curve. The graph must be one created by manual plotting or by a user function. The y-axis data for the curve are obtained from the *region* argument. If *region* contains a single region, then the y-axis data consists of an analysis of the region in all indicated frames of the region's image window. The *region* argument can also contain an array of regions from the same image window. In this case, the y-axis data are generated by analyzing each region in its home frame, provided that frame is in the graphing subset (See SetFrameFlags). The other arguments for this function are provided by the PutArg macro. The arguments are:

name	identifier	type	code	range
y_axis_name	$0019c830	string	1	27 characters max
y_axis_units	$0019c832	string	1	27 characters max
x_axis_type	$0019c83e	short int	2	0 ≤ x ≤ 2
x_axis_time_unit	$0019c84a	short int	2	0 ≤ x ≤ 2
x_axis_pause_exc	$0019c848	short int	2	0 or 1
x_axis_start_val	$0019c840	real	4	real numbers
x_axis_inc_val	$0019c842	real	4	x > 0
y_axis_type	$0019c83c	short int	2	0 ≤ x ≤ 5
point_type	$0019c834	short int	2	0 ≤ x ≤ 6
point_connect	$0019c836	short int	2	0 or 1
second_axis	$0019c838	short int	2	0 or 1

The above arguments correspond to those that are entered in the New Variable Settings dialog:

name	value	meaning
x_axis_type	0	increment from x_axis_start_val by x_axis_inc_val

```
                1     use frame number
                2     elapsed time in x_axis_time_unit
x_axis_time_unit 0    milliseconds
                1     seconds
                2     minutes
x_axis_pause_exc 0    add time of pauses to x-axis values
                1     exclude pause times from x-axis values
y_axis_type     0     total activity in region
                1     average activity in region
                2     count rate in region
                3     size of region
                4     minimum pixel value in region
                5     maximum pixel value in region
point_type      0     circle
                1     +
                2     x
                3     triangle
                4     square
                5     dot
                6     bar-graph
point_connect   0     do not draw lines between points
                1     draw lines between points
second_axis     0     use primary axis
                1     use secondary axis if primary axis exists
```

T_GRAPH **RowAverages**(T_REGION *theRegion*, T_NUMBER *frame*)
RowAverages returns a graph of the row averages of a rectangular ROI given by *theRegion* in its home image and in the frame given by *frame*.

T_GRAPH **RowSums**(T_REGION *theRegion*, T_NUMBER *frame*)
RowSums returns a graph of the row sums of a rectangular ROI given by *theRegion* in its home image and in the frame given by *frame*.

SetCurveSymbol(T_GRAPH *theGraph*, T_NUMBER *theCurve*,
T_STRING *symbol*)
Use SetCurveSymbol to associate a text symbol with a curve specified by curve number *theCurve* in the graph given by *theGraph*. The symbol is used in subsequent curve math operations.

ShowGraphText(T_GRAPH *theGraph*, T_NUMBER *show*)
Use ShowGraphText to show or hide the text box beneath graphs. Set the *show* parameter to one (1) to display the graph text if it is not already visible. Set the *show* parameter to zero (0) to hide the graph text if it is not already hidden.

8.8.8 Editing

These functions allow you to apply the standard editing commands to windows.

Clear(T_WINDOW *theWindow*)
Clear selects the window specified by *theWindow* and deletes the current selection, if any.

Copy(T_WINDOW *theWindow*)
Copy selects the window specified by *theWindow* and copies the current selection to the clipboard.

Cut(T_WINDOW *theWindow*)
Cut selects the window specified by *theWindow* and cuts the current selection, placing the selection in the clipboard.

Paste(T_WINDOW *theWindow*)
Paste selects the window specified by *theWindow* and pastes the contents of the clipboard into the window, if possible.

SelectAll(T_WINDOW *theWindow*)
SelectAll selects the window specified by *theWindow* and attempts to select everything in the window, be it objects or text.

8.8.9 Objects

These functions apply to all types of objects. The following sections describe functions dealing with specific types of objects.

HideAllObjects(T_IMAGE *theImage*)
HideAllObjects makes all objects in *theImage* invisible.

PlaceMask(T_IMAGE *theImage*, T_NUMBER *size*, T_NUMBER *color*)
PlaceMask places a colored mask in the image window specified by *theImage*. The mask covers the top and bottom *size* pixels of the image in all frames. Use these constants for the *color* parameter:

```
black   33
white   30
red     205
green   341
blue    409
cyan    273
magenta 137
yellow  69
```

ShowAllObjects(T_IMAGE *theImage*)
ShowAllObjects makes all objects in *theImage* visible.

8.8.10 Regions

These functions deal with regions of interest. Note that regions keep track of their home image windows, so by specifying a region, you are also choosing an image window.

T_ARRAY **Analyze**(T_REGION *theRegion*, T_NUMBER *frame*)
Use the Analyze function to analyze an ROI in its home image and in the frame specified by the *frame* parameter. The array returned by Analyze has the following values:

```
Element Type and Value
0       T_NUMBER—Maximum pixel value in the ROI
1       T_NUMBER—Minimum pixel value in the ROI
2       T_NUMBER—Number of pixels in theROI
3       T_NUMBER—Sum of the pixel values in the ROI
4       T_NUMBER—Sum of the squares of the pixel values in the ROI
```

Your macros will be easier to read if you do not use the element indexes directly. Instead, use the functions beginning with ROI_ to identify the desired index. For example, to access the number of pixels in an ROI, write something like this:

result = `Analyze`(myRegion,1)
number_of_pixels = result[`ROI_NumberOfPixels]`

T_ARRAY **`AnalyzeQARegion`**(T_REGION *theRegion*, T_NUMBER *frame*,
T_NUMBER *roiReduction*)
`AnalyzeQARegion` performs a quality assurance analysis of the specified region in the given image frame. The *roiReduction* parameter is displayed as the percent by which the region has been reduced from some base size. This value does not affect any of the calculations. Set *roiReduction* to 100 if there has been no previous reduction of the region. The array returned by `AnalyzeQARegion` has the following values:

```
Element Type and Value
0       T_IMAGE—A window showing horizontal and vertical processing of
the ROI
1       T_GRAPH—A graph of intra-group spread in the ROI
2       T_GRAPH—A graph of inter-group spread in the ROI
3       T_NUMBER—Total differential non-linearity (DNL)
4       T_NUMBER—Total integral non-linearity (INL)
```

T_REGION **`CopyRegionTo`**(T_REGION *theRegion*,T_IMAGE *theImage*,
T_NUMBER *frame*)
Given a region specified by *theRegion*, `CopyRegionTo` makes a copy of the region in the image and frame specified by *theImage* and *frame*..

T_NUMBER **`CountRegions`**(T_IMAGE *theImage*, T_NUMBER *frame*)
`CountRegions` returns the number of visible ROIs in the image window and frame given by the respective parameters.

`DeleteAllRegions`(T_IMAGE *theImage*)
Given an image in *theImage*, `DeleteAllRegions` removes all regions in all of the image frames.

`DeselectRegion`(T_REGION *theRegion*)
Given an ROI in *theRegion*, `DeselectRegion` ensures that the region is not selected.

T_REGION **`FrontRegion`**

Given that an image window is active, and there is a selected ROI in the active frame, `FrontRegion` returns a representation of the ROI.

T_REGION **FullOvalRegion**(T_IMAGE *theImage*, T_NUMBER *frame*)
Given an image and frame number, FullOvalRegion creates an oval ROI that is bounded by the visible rectangle of the image window.

T_REGION **FullRectangleRegion**(T_IMAGE *theImage*, T_NUMBER *frame*)
Given an image and frame number, FullRectangleRegion creates a rectangular ROI that fills the visible area of the image window.

T_STRING **GetRegionName**(T_REGION *theRegion*)
Given a region, GetRegionName returns the name of the region.

T_ARRAY **GetROIBounds**(T_REGION *theRegion*)
Given a region, GetROIBounds returns an array containing the rectangular bounding coordinates of the region. The array values are as follows:

Element	Type and Value
0	T_NUMBER—Top coordinate
1	T_NUMBER—Left coordinate
2	T_NUMBER—Bottom coordinate
3	T_NUMBER—Right coordinate

T_REGION **PlaceOvalRegion**(T_IMAGE *theImage*, T_NUMBER *frame*,
T_NUMBER *top*, T_NUMBER *left*,
T_NUMBER *bottom*, T_NUMBER *right*)
PlaceOvalRegion creates a new oval region in the specified image and frame with the coordinates of the bounding rectangle given by *top*, *left*, *bottom* and *right.*

T_REGION **PlaceRectangleRegion**(T_IMAGE *theImage*, T_NUMBER *frame*,
T_NUMBER *top*, T_NUMBER *left*,
T_NUMBER *bottom*, T_NUMBER *right*)
PlaceRectangleRegion creates a new rectangular region in the specified image and frame with the coordinates given by *top, left, bottom* and *right.*

T_REGION **PromptForOvalRegion**(T_IMAGE *theImage*, T_NUMBER *frame*,
T_STRING *thePrompt*)
Use PromptForOvalRegion to ask the user to draw an oval region on the given image window and frame. *thePrompt* is the text that is supplied to the macro user.

T_REGION **PromptForRectangleRegion**(T_IMAGE *theImage*,
T_NUMBER *frame*,
T_STRING *thePrompt*)
Use PromptForRectangleRegion to ask the user to draw a rectangular region on the given image window and frame. *thePrompt* is the text that is supplied to the macro user.

T_REGION **PromptForRegion**(T_IMAGE *theImage*, T_NUMBER *frame*,
T_STRING *thePrompt*)
Use PromptForRegion to ask the user to draw a freehand region on the given image window and frame. *thePrompt* is the text that is supplied to the macro user.

PromptForRegionMove(T_IMAGE *theImage*, T_NUMBER *frame*,
T_STRING *thePrompt*)
Use PromptForRegionMove to ask the macro user to move or resize any existing region or other object in the image and frame specified. *thePrompt* is the text that is displayed to the macro user.

T_NUMBER **ROIXCentroid**(T_REGION *theRegion*, T_NUMBER *frame*)
ROIXCentroid returns the x-centroid of a rectangular ROI given by *theRegion* in its home image and in the frame given by *frame*.

T_GRAPH **ROIXCentroidGraph**(T_REGION *theRegion*)
ROIXCentroidGraph returns a graph of the x-centroid of a rectangular ROI given by *theRegion* in its home image and across all frames.

T_NUMBER **ROIYCentroid**(T_REGION *theRegion*, T_NUMBER *frame*)
ROIYCentroid returns the y-centroid of a rectangular ROI given by *theRegion* in its home image and in the frame given by *frame*.

T_GRAPH **ROIYCentroidGraph**(T_REGION *theRegion*)
ROIYCentroidGraph returns a graph of the y-centroid of a rectangular ROI given by *theRegion* in its home image and across all frames.

T_NUMBER **ROI_Maximum**
Use ROI_Maximum as an index into the array returned by the Analyze function. This index yields the maximum pixel value.

T_NUMBER **ROI_Minimum**
Use ROI_Minimum as an index into the array returned by the Analyze function. This index yields the maximum pixel value.

T_NUMBER **ROI_NumberOfPixels**
Use ROI_NumberOfPixels as an index into the array returned by the Analyze function. This index yields the number of pixels in the ROI.

T_NUMBER **ROI_TotalCounts**
Use ROI_TotalCounts as an index into the array returned by the Analyze function. This index yields the sum of all pixel values in the ROI.

T_NUMBER **ROI_TotalCounts2**
Use ROI_TotalCounts2 as an index into the array returned by the Analyze function. This index yields the sum of the squares of all pixel values in the ROI.

T_REGION **SelectedRegionIn**(T_IMAGE *theImage*, T_NUMBER *frame*)
Use SelectedRegionIn to return a variable representing a selected region in the specified image and frame. If more than one region is selected, then it cannot be predicted which will be returned by SelectedRegionIn. An error message is generated if there are no regions selected in the designated image and frame. Use CountRegions to be sure a selected region exists.

SelectRegion(T_REGION *theRegion*)
Use SelectRegion to cause the specified region to be selected in its home window.

SetRegionName(T_REGION *theRegion*, T_STRING *name*)
Given a region, SetRegionName sets the name of the region to the string passed in the *name* parameter. The name is truncated to 27 characters.

8.8.11 Axes

These functions apply to the axes that can overlay any study window.

HideAxes(T_IMAGE *theImage*)
Use HideAxes to make invisible the axes overlaying *theImage*.

PlaceAxes(T_IMAGE *theImage*, T_NUMBER *x_loc*, T_NUMBER *y_loc*)
Use PlaceAxes to move and make visible the axes overlaying *theImage*. Specify the new origin with the *x_loc* and *y_loc* parameters.

ShowAxes(T_IMAGE *theImage*)
Use ShowAxes to make visible the axes overlaying *theImage*.

8.8.12 Text

PlaceText(T_IMAGE *theImage*, T_NUMBER *frame*, T_STRING *theText*,
T_NUMBER *x_loc*, T_NUMBER *y_loc*, T_STRING *fontName*,
T_NUMBER *fontSize*, T_NUMBER *fontStyle*, T_NUMBER *fontColor*,
T_NUMBER *backgroundColor*)
Use the PlaceText function to place a new text object in an image. Specify the image and frame in which to place the text, as well as the text and the coordinates of the top, left corner of the text rectangle. Specify the name of the font to use, e.g. "Chicago" and the point size, style, color and background color. Sum these constants to produce the desired style:

```
normal    0
bold      1
italic    2
underline 4
outline   8
shadow   16
condense 32
extend   64
```

Use these constants for font and background colors:

```
black   33
white   30
red     205
green   341
blue    409
cyan    273
magenta 137
yellow  69
```

8.8.13 Arrows

PlaceArrow(T_IMAGE *theImage*, T_NUMBER *frame*, T_NUMBER *top*, T_NUMBER *left*, T_NUMBER *bottom*, T_NUMBER *right*)
Use the PlaceArrow function to place a new arrow object in an image. Specify the image and frame in which to place the arrow, as well as the top, left, bottom, and right coordinates for the arrow.

8.8.14 Color Tables

AssignColorsWindow(T_IMAGE *theImage*,T_NUMBER *whichWindow*)
The AssignColorsWindow function causes the image window given by *theImage* to be redrawn in the colors appearing in the Colors window specified by *whichWindow*. *whichWindow* must be in the range of zero to the total number of Colors windows less one (see CountColorsWindows).

LoadColors(T_STRING *colorTable*,T_NUMBER *whichWindow*)
The LoadColors function loads the colors in the table specified by *colorTable* into the Colors window specified by *whichWindow*. *whichWindow* must be in the range of zero to the total number of Colors windows less one (see CountColorsWindows).

LoadColorSet(T_STRING *colorSet*)
The LoadColorSet function loads a set of color tables. If necessary, Colors windows are created or deleted to match the number of color tables specified by *colorSet*..

8.8.15 Dialogs

MessageDialog(T_STRING *theMessage*)
MessageDialog presents the macro user with a modal dialog containing the text specified by *theMessage*.

T_NUMBER **Query**(T_STRING *theQuery*, T_STRING *OK_String*, T_STRING *Cancel_String*)
Use the Query function to present the macro user with a dialog containing two choices, which are nominally OK and Cancel. *theQuery* is the question; *OK_String* is the text used for the highlighted button, which can be selected with the return key; *Cancel_String* is the text used for the other button, which can be selected with the esc key or command-period.

8.8.16 User Functions

T_ANY **RunUserFunction**(T_STRING *userFunctionName*, T_ANY *arguments*)
RunUserFunction loads and executes a user function with the given name from the user functions directory. The following are new in versions of NucLear MAC beyond 2.3a2.0: *userFunctionName* can also be a full path name. The *arguments* parameter has been added, which allows arguments to be passed to the user function. User functions can return values to the macro.

8.8.17 Strings

These functions deal with strings and text display.

`Display`(T_ANY *displayList*)
The `Display` function is one of the few that takes a variable number of arguments, from one on up. It is used to display text in the Analysis window. You can mix strings and numbers among the arguments. For example, Display("Here is a number: ",10,"\r") The format in which the numbers appear depends on the current number format and whether trailing zero suppression is enabled. These are controlled by the `SetNumberFormat` and `SuppressTrailingZeros` functions respectively. The font format can be changed with `SetDisplayTextFormat` .

T_STRING **`GetDateString`**

`GetDateString` returns a string containing the current date. The date format is determined by the operating system. This is user-defineable on newer systems.

T_NUMBER **`GetDateTime`**
`GetDateTime` returns the current date and time in Macintosh numerical format. This is a number representing seconds elapsed since midnight, January 1, 1904.

T_STRING **`GetTimeString`**

`GetTimeString` returns a string containing the current time. The time format is determined by the operating system. This is user-defineable on newer systems.

T_STRING **`NumberToString`**(T_NUMBER *theNumber*)
`NumberToString` converts a number to a text string. The format in which the number appears depends on the current number format and whether trailing zero suppression is enabled. These are controlled by the `SetNumberFormat` and `SuppressTrailingZeros` functions respectively.

T_STRING **`PromptForString`**(T_STRING *thePrompt*, T_STRING *default*)
The `PromptForString` function asks the macro user to enter a string. Pass the prompt string as *thePrompt* and a default string as the *default* parameter. If the user cancels the request, then the macro is aborted.

`SetDisplayTextFormat`(T_STRING *font*,T_NUMBER *fontSize*,
T_NUMBER *style*)
`SetDisplayTextFormat` is used to set the format of text sent to the Analysis window with the `Display` command. The *font* string parameter is the name of the font to be used, e.g. "Chicago". Set the *fontSize* parameter to the desired point size of the font. Sum these constants to produce the desired style:

```
normal    0
bold      1
italic    2
underline 4
outline   8
shadow   16
```

```
condense 32
extend   64
```

SetNumberFormat(T_NUMBER *format*)
SetNumberFormat is used to choose how numbers are converted to text. It takes an integer in the range of -20 to 20. A positive value causes numbers to be written in fixed point; *format* indicates the number of digits after the decimal point. A negative value causes numbers to be written in floating point; the absolute value of *format* determines the number of digits. When the *format* parameter is zero, only integers are written. The default *format* value is 4.

T_STRING **StringCharacter**(T_STRING *theString*, T_NUMBER *index*)
Use StringCharacter to examine a particular character of *theString*. The *index* parameter ranges from zero to the length of *theString* less one. StringCharacter returns a string of length one.

T_NUMBER **StringCompare**(T_STRING *firstString*, T_STRING *secondString*)
StringCompare returns a value of -1 if *firstString* is less than *secondString*, zero if the strings are identical, and 1 if *firstString* is greater than *secondString*.

T_STRING **StringConcat**(T_STRING *firstString*, T_STRING *secondString*)
StringConcat appends *secondString* to *firstString* and returns the result. The sum of the lengths of the two strings should not exceed 255.

T_NUMBER **StringEquals**(T_STRING *firstString*, T_STRING *secondString*)
StringEquals returns a value of one if *firstString* is identical to *secondString*, and zero otherwise.

T_STRING **StringLeft**(T_STRING *theString*, T_NUMBER *nchars*)
StringLeft returns a string consisting of the left-most *nchars* characters of *theString*. If *theString* has fewer than *nchars* characters, then StringLeft returns a string identical to *theString*.

T_NUMBER **StringLength**(T_STRING *theString*)
StringLength returns the number of characters currently in *theString*.

T_STRING **StringRight**(T_STRING *theString*, T_NUMBER *nchars*)
StringRight returns a string consisting of the right-most *nchars* characters of *theString*. If *theString* has fewer than *nchars* characters, then StringRight returns a string identical to *theString*.

T_NUMBER **StringToNumber**(T_STRING *theString*)
StringToNumber attempts to convert the string parameter into a number.

SuppressTrailingZeros(T_NUMBER *turn_on*)
SuppressTrailingZeros is used to enable and disable the suppression of zeros after the decimal point when numbers are converted to text. Zero suppression is enabled by default.

8.8.18 Numbers

These functions deal exclusively with numerical values.

T_NUMBER **abs**(T_NUMBER *theNumber*)
The `abs` function returns the absolute value of *theNumber*.

T_NUMBER **cos**(T_NUMBER *theAngle*)
The `cos` function returns the cosine of *theAngle*. *theAngle* is assumed to be in radians.

T_NUMBER **exp**(T_NUMBER *x*)
The `exp` function returns e^x.

T_NUMBER **int**(T_NUMBER *theNumber*)
The `int` function returns the integer portion of *theNumber*. For example, `int`(2.7) returns 2, and `int`(-2.7) returns -2.

T_NUMBER **log**(T_NUMBER *theNumber*)
The `log` function returns the natural logarithm of *theNumber*.

T_NUMBER **pi**
The function `pi` returns an approximation to π.

T_NUMBER **PromptForNumber**(T_STRING *thePrompt*, T_NUMBER *default*)
The `PromptForNumber` function asks the macro user to enter a number. Pass the prompt string as *thePrompt* and a default number as the *default* parameter. If the user cancels the request, then the macro is aborted.

T_NUMBER **sin**(T_NUMBER *theAngle*)
The `sin` function returns the sine of *theAngle*. *theAngle* is assumed to be in radians.

T_NUMBER **sqr**(T_NUMBER *theNumber*)
The `sqr` function returns the square of *theNumber*.

T_NUMBER **sqrt**(T_NUMBER *theNumber*)
The `sqrt` function returns the square root of *theNumber*.

T_NUMBER **tan**(T_NUMBER theAngle)
The `tan` function returns the tangent of *theAngle*. *theAngle* is assumed to be in radians.

8.8.19 Files

The following functions relate to the NucLear MAC file menu and the Macintosh file system. A note about file names is necessary: Many of these functions use full path names to identify files and folders. These have the following form:

VolumeName:*FolderName*:*SubFolderName*:…:*FileName*

Observe how semicolons separate the various directory names.

T_STRING **BootVolume**
BootVolume returns the name of the volume that was used to boot the Macintosh. This is the volume that contains the active System Folder.

T_NUMBER **BoundToFile**(T_WINDOW *theWindow*)
Given a NucLear MAC window, which must be an image or a graph, BoundToFile returns a value of one if *theWindow* has been saved to disk, and zero otherwise.

Close(T_WINDOW *theWindow*)
Given a NucLear MAC window, Close either hides or disposes of *theWindow*. Standard NucLear MAC window such as the color bar and the Analysis window are hidden. Images, graphs, and All Angles windows are disposed of. If images or graphs have not been saved, then the user can cancel the Close, which aborts the macro.

CloseAllPalettes
CloseAllPalettes hides all the NucLear MAC floating palettes. These include the color bar, the progress window and the controls window.

CloseAllWindows
CloseAllWindows attempts to close all windows other than palette windows. If there are images or graphs that have not been saved, then the user can abort the closing process.

DeleteFile(T_STRING *fileName*)
Given a full path name to a disk file, DeleteFile permanently removes the file. Use this command with extreme care, as you cannot recover any file after you have deleted it.

ExportAs(T_WINDOW *theWindow*, T_STRING *fileName*, T_NUMBER *format*)
ExportAs exports the window given by *theWindow* to the file specified by *fileName*. Use one of these values for the *format* parameter:

0 PICT
1 TIFF
2 TEXT
3 Siemens μ
4 ACR-NEMA
5 Interfile

T_NUMBER **FileOrFolderExists**(T_STRING *name*)
Given a full path name to a file or folder, FileOrFolderExists determines whether such a file or folder exists. If neither exists, then a value of zero is returned. If a file exists, then a value of one is returned. And, if a folder exists, then a value of two is returned.

T_STRING **GetFileDirectory**(T_WINDOW *theWindow*)
GetFileDirectory returns the path name of the directory containing the file that is a saved version of the specified window. This command applies to macros, images, and graphs. Use this function only if the function BoundToFile returns a value of one (1) for the specified window.

T_ARRAY **GetFileInfo**(T_STRING *fileName*, T_NUMBER *index*)
Given a full path name to a file or folder and an index, GetFileInfo returns an array of the following information about the item:

Element	Type and Value
0	T_STRING, name of item
1	T_STRING, four character type if file, empty string if folder
2	T_STRING, four character creator if file
3	T_NUMBER, length of file data fork
4	T_NUMBER, length of file resource fork
5	T_NUMBER, creation date
6	T_NUMBER, modification date

When *index* is zero, the information describes the object *fileName*. For values of *index* greater than zero, *fileName* must describe a directory, and *index* defines which object in the directory to describe. When scanning a folder using the index, check the size of the result using ArraySize. A non-existent object causes a numerical value rather than an array to be returned.

T_PRINTREC **GetWindowPageSetup**(T_WINDOW *theWindow*)
GetWindowPageSetup returns a copy of the page setup information associated with *theWindow*. An error results if the window has no such information, so you should first call WindowHasPageSetup to verify the record exists.

T_STRING **HomeVolume**

HomeVolume returns the name of the volume on which the NucLear MAC program is running.

MarkAsSaved(T_WINDOW *theWindow*)
Use MarkAsSaved to avoid a prompt when an unsaved graph or image window is closed. MarkAsSaved sets a flag telling NucLear MAC the window has been saved, even if it has not. This is a dangerous function. If you use it indiscriminately, you may cause someone to lose important data.

NewFolder(T_STRING *folderName*)
Given a full path name, NewFolder creates a new folder (directory) having that path name. It's a good idea to see if the folder already exists, because you will cause an error if it does, or if there is an existing file with the same name.

T_WINDOW **Open**(T_STRING *fileName*)
Use the Open command to open any document that NucLear MAC is capable of opening. *fileName* is a full path name of the file to be opened.

T_ARRAY **OpenAll**(T_STRING *directory*)
Use the OpenAll command to open all files in a directory. The *directory* string parameter contains the path name of the directory containing the files. The size of the array returned by OpenAll depends on how many files were opened. The array elements represent images, graphs, and macros.

Print(T_WINDOW *theWindow*,T_NUMBER *skipDialog*)
The Print command causes a window to be printed on the current printer. If the *skipDialog* parameter is non-zero, then no print job dialog is presented, and default values are used. Otherwise, the printer-specific dialog is presented to the user prior to printing. This allows the user to set items such as number of copies and page range.

PromptExportAs(T_WINDOW *theWindow*)
PromptExportAs prompts the user to export the window specified by *theWindow*.

PromptForPageSetup(T_WINDOW *theWindow*,T_NUMBER *setDefault*)
PromptForPageSetup prompts the user to edit the page setup for the window specified by *theWindow*. When the *setDefault* parameter is non-zero, the page setup selected will become the default for this type of window.

T_WINDOW **PromptOpen**
PromptOpen produces the same result as choosing Open from the NucLear MAC's File menu. You have no control other than suggestion over what file the user chooses to open, so do not assume it is a particular type.

T_STRING **PromptForDirectory**(T_STRING *promptString*)
The PromptForDirectory command brings up a standard file dialog with which the user can select a directory. The path name of the directory is returned in a string. The string given by the *promptString* parameter is displayed in the dialog.

PromptSaveAs(T_WINDOW *theWindow*)
The PromptSaveAs is equivalent to bringing *theWindow* to the front and choosing Save As from the File menu. This command applies only to images or graphs.

Save(T_WINDOW *theWindow*)
Use Save to directly save changes to *theWindow*. *theWindow* must have been saved previously. Use BoundToFile to verify this.

SaveAs(T_WINDOW *theWindow*, T_STRING *fileName*)
Use SaveAs to directly save *theWindow* as the file specified by the full path name in *fileName*. To avoid generating an error, you should check whether a file or folder of the same name already exists.

SetImportFormat(T_STRING *formatName*)
The SetImportFormat command changes the current image import format to the pre-defined format specified by the *formatName* parameter. An error is generated if the specified format does not exist—formats are defined in the Import File Formats dialog. If SetInputFormat executes successfully, subsequent Open commands will use the format when files with unknown types are encountered.

SetWindowPageSetup(T_WINDOW *theWindow*, T_PRINTREC *printRec*)
SetWindowPageSetup changes the printing information associated with *theWindow* to that contained in *printRec*.

T_NUMBER **`WindowHasPageSetup`**(T_WINDOW *theWindow*)
`WindowHasPageSetup` returns one (1) if *theWindow* has page setup information and zero (0) otherwise.

9 User Functions

9.1 Introduction

This chapter describes the process of creating external, “drop-in” functions for NucLear MAC. The external user function capability is provided so that NucLear MAC owners with programming experience can extend NucLear MAC with their own routines for image processing, graphing, and region-of-interest (ROI) analysis. External functions are compiled into Macintosh files that are placed in a directory accessible to NucLear MAC. The **Set Directory...** command in the **Analyze>User Functions** submenu is used to designate the user functions directory.

The interface to user functions is such that little Macintosh-specific programming knowledge is required to make external functions. Depending on image representation, functions can be developed on an entirely different platform, only to be compiled for the Macintosh as a last step.

Any programming language can be used to develop user functions, although the C language is strongly recommended, as it is the language of the host system. Because the calling conventions follow C standards, use of Pascal or Fortran requires additional “glue” code. Development of user functions in C requires the use of a single include file. Scientific Imaging supplies the following files to aid in the development of user functions:

User.h This C include file defines the data structures and macros for the call-back routines provided by NucLear MAC.

Extract Region.c This is the C source file of a working user function that demonstrates ROI processing. The image data in an ROI is extracted and placed in a new window.

Extract Region.π This file is a THINK C project that includes all settings for direct compilation of Extract Region.c into a user function.

Extract Region.π.rsrc This resource file contains the help text for the **Extract Region** user function.

Extract Region This file contains the compiled, ready-to-run, region-extraction user function.

Extremes.c This is the C source file of a working user function that demonstrates image access and graph generation. The minimum and maximum pixel values are plotted as a function of image frame.

Extremes.π This file is a THINK C project that includes all settings for direct compilation of Extremes.c into a user function.

Extremes.π.rsrc This resource file contains the help text for the **Extremes** user function.

Extremes This file contains the compiled, ready-to-run, min/max graphing user function.

9.2 The User Function Subroutine

A user function is a subroutine that takes a single argument and returns a short integer. In C, the declaration is

```
short UserFunction(NucMacCallBackProc callbackproc);
```

where `NucMacCallBackProc` is defined by

```
typedef long (*NucMacCallBackProc)(short selector,...);
```

The user function returns one (1) if its execution is successful, or zero (0) if not. The latter includes any case where the user cancels a prompt. The `NucMacCallBackProc` argument is the means provided for the user function to communicate with the NucLear MAC during its invocation. When the user function requires information from or action by the host, it calls the procedure pointed to by `callbackproc` with a function selector and a variable number of function-dependent arguments.

Typically the user function analyzes or transforms image data, so its first goal when called is to determine the source data window(s). The call-back selector PROMPT_FOR_DATA_WINDOW can be employed by the user function to prompt the user to select a window for a given operation. A convenient shortcut for the NucLear MAC user is to have the front (active) window automatically used as a source, so that a prompt is not necessary. This is supported by a call-back function selected with the constant FRONT_DATA_WINDOW. Hence a typical course of action is:

IF (There is a front window) **THEN**
  Do Stuff using front window
ELSE {
  Prompt for a window
  IF (User supplied a window) **THEN**
    Do Stuff using user-supplied window
  ELSE Give Up
}

A similar paradigm applies to operations requiring an ROI. In cases where it won't confuse the user, a pre-existing ROI can be used to perform a function. The selection constants for ROIs are PROMPT_FOR_ROI and GET_WINDOW_ROI. The use of these constants parallels the use of PROMPT_FOR_DATA_WINDOW and FRONT_DATA_WINDOW above:

IF (Window has selected ROI) **THEN**
  Do Stuff using existing ROI
ELSE {
  Prompt for an ROI in window

```
  IF (User created an ROI) THEN
    Do Stuff using user-supplied ROI
  ELSE Give Up
}
```

More involved ROI manipulations are possible with the selectors PROMPT_FOR_EXISTING_ROI, COUNT_ROIS, GET_UPDATED_ROI, and GET_ALL_WINDOW_ROIS. ROI names can be read and written with the selectors GET_ROI_NAME and SET_ROI_NAME.

Several selectors are provided to allow user functions to generate and edit graphs from arrays of real numbers. These include PUT_GRAPH_INFO, NEW_GRAPH, SET_GRAPH_X_LIMITS, ADD_GRAPH_DATA_SET, and SET_GRAPH_Y_LIMITS.

Because user functions cannot change existing window data without dire consequences, they must always perform data transformations by asking NucLear MAC for a new, blank window into which to place transformed data. In such cases, the characteristics for a newly created window are usually taken from the source window. For example, the destination window of a user function that sums all frames of a source window would match the source in height and width, but would contain a single frame, regardless of the number of source frames. In addition, the destination window would always use word-sized pixels, since byte-sized source pixels could easily overflow into words. The call-back selection constant for a new window request is NEW_DATA_WINDOW.

Call-back routines are provided that allow user functions to receive and return arguments. The arguments can be images, graphs, numbers, strings, and ROIs. It is strongly recommended that these routines be used wherever practicable. This will allow user functions to exchange data with macros, which in turn gives users greater flexibility in using NucLear MAC. A user function should thus be prefaced with the following logic:

```
IF (Valid arguments passed to user function) THEN
  Do Stuff using passed arguments
ELSE
  Look for or prompt for arguments, etc.
```

In addition, if a user function executes successfully, then it should return any useful items generated, be they images, graphs, numbers, strings, or ROIs. This gives a calling macro the opportunity to further manipulate these items in ways that would be difficult for the user function.

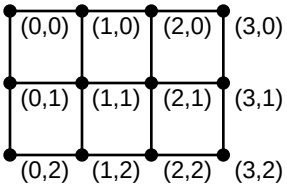
Many other call-back routines are provided. These and those mentioned above are described in detail in a separate section. Other calls provide a means of getting numbers and image coordinates from the user, as well as displaying numbers and text in the NucLear MAC program. Future expansion will be accomplished with the definition of new call-back routines and associated selection constants.

9.3 Data Representation and Structures

9.3.1 Coordinate System

NucLear MAC uses the Apple-defined Quickdraw coordinate system when referring to images and ROIs. This system is summarized here, and a detailed description can be found in Inside Macintosh, Volume I.

The basis for the Quickdraw coordinate system is a grid of points. Each point is given as an x-y pair of integers, for example, (1,1). Compared to Cartesian coordinates, the Quickdraw system appears to be upside down. That is, y coordinates increase in the downward direction:



Coordinate system

In C, a point is defined by

```
typedef struct {
 short v,h;
} Point ;
```

Note that 12 points are visible in Figure 1. Another important data structure is the rectangle, declared as

```
typedef struct {
 short top,left,bottom,right;
} Rect ;
```

or

```
typedef struct {
 Point topLeft,botRight;
} Rect ;
```

A rectangle is simply a frame that can be defined by two points. Rectangles that are not empty contain pixels. A pixel is a 1 x 1 square with a point on each corner. By convention, a pixel is identified by the point at its top, left corner. For example, in Figure 1 the top, left pixel shown is identified by the point (0,0). The remaining corner points of this pixel are (0,1), (1,1), and (1,0). The rectangle enclosing only this pixel is defined by the points (0,1) and (1,1) (or top=0, left=0, bottom=1, right=1). Similarly, the six pixels visible in Figure 1 are enclosed by the rectangle (0,0) (3,2).

In NucLear MAC, the top, left pixel of an image is always represented by the point (0,0). Note that this precludes the use of negative coordinates. In addition, this means that only two numbers (or one point) are required to define the bounding rectangle of an

image. Associating pixels, which are stored by row, with memory locations is complicated by the fact that the number of pixels in a given row must be rounded up to the next power of four. This rounding up is transparent to the user since only the pixels in the bounding rectangle are displayed. In C, the rounding calculation is written as

```
rowElements = hdim + 3 & 0xfffc;
```

where `hdim` is the width of the image bounding rectangle. The reason we define this as `rowElements` is that NucLear MAC supports both byte- and word-sized pixels. Hence a row element can be either one or two bytes long. Suppose you wish to randomly access an image pixel at the point (x,y). With a byte-mode image assume you have a variable “bytepointer” that points to the first (top, left) pixel of the image. Then the pixel at (x,y) is accessed as

```
*(bytepointer + (long) y * rowElements + x)
```

Note that if y were not extended to a long the product of the two short integers would be truncated to a short with undesirable results. In a word-mode image you would assume the existence of a “wordpointer” that also points to the top, left pixel, but which is defined as

```
unsigned short *wordpointer;
```

instead of

```
unsigned char *bytepointer;
```

It is assumed that sizeof(char) == 1 and sizeof(short) == 2.

As above, the pixel at (x,y) is accessed as

```
*(wordpointer + (long) y * rowElements + x)
```

Conveniently, the C language automatically recognizes that an integer number of elements added to wordpointer must be multiplied by sizeof(short), which is two. Extending this, we note that the total size in bytes of a byte-mode image is

```
(long) vdim * rowElements
```

and for a word-mode image it is simply double this. `vdim` is the height of the image bounding rectangle.

9.3.2 Study Windows

With the coordinate system defined, we can define the structures that NucLear MAC creates to describe study windows to user functions. In C, the declarations are

```
typedef struct {
  long reserved; /* Internal use only! */
  union {
    unsigned char *byteptr; /* Pointer to data bytes… */
```

```
    unsigned short *wordptr; /* or words... */
  } dp;
} FramePtrRecord;

typedef struct {
  long reserved[4]; /* Internal use only! */
  short hdim,       /* Horizontal image dimension */
  vdim,             /* Vertical image dimension */
  rowElements,      /* Row bytes/words in image  */
                    /* data = hdim + 3 & 0xfffc */
  nframes,          /* Number of frames with window */
                    /* (#Elements in frames array) */
  frontframe,       /* Visible frame in movie-type */
                    /* window. (Not all-frames) */
  wordmode;         /* Zero if each pixel is one byte */
                    /* (values 0-255) and one if each */
                    /* pixel is two bytes (0-65535) */
                    /* The Motorola system is followed, */
                    /* so most significant byte first. */
  unsigned short maxword, /* Value of maximum pixel in */
                    /* word mode images **only** */
  spare;            /* Internal use only! */
  long spare1;      /* Internal use only! */
  FramePtrRecord frames[]; /* One record for each window */
                    /* frame */
} DataWindowRecord,*DataWindowPtr;
```

The `DataWindowRecord` struct consists mainly of the variables discussed above: the image dimensions, row elements, and an indication of whether pixels are represented by bytes or words. The `nframes` field tells how many image frames are associated with the window. For each frame there is a `FramePtrRecord` in the `frames` array. The `FramePtrRecord` contains a union so that either a byte pointer or word pointer can be accessed, depending on the `wordmode` field of the `DataWindowRecord`. These pointers point to the top, left pixel locations of their respective frames.

NOTE! The word and byte pointers in the `FramePtrRecords` are subject to change with the use of any call-back function. What happens is that image memory may be shifted around during the call-back routine, but the pointers are correctly updated before returning to the user function. The bottom line is that user functions cannot save image pointers across call-back function calls and expect them to remain valid.

The `frontframe` field indicates which frame is currently visible and accessible to the user. When a user function prompts for an ROI, this is the frame that will be used. Note that `frontframe` can be used directly as an index in the `frames` array. The `frontframe` field ranges from zero to `nframes-1`. Call-back functions are provided that allow a user function to change the front frame or prompt the user for a desired frame. Note that a one-based frame count is expected by NucLear MAC users, rather than the zero-based count used in programming.

The `maxword` field is only applicable to word-mode images (`wordmode == 1`). In these cases, `maxword` is the value of the highest pixel in all of the image frames. The one exception is that `maxword` can never be zero, so in a blank image `maxword` will be set to one.

Pointers of type `DataWindowPtr` are passed back and forth between the user function and the NucLear MAC. User functions must never alter the structures to which these point. The only NucLear MAC-generated data a user function can change are the pixel values of image windows created by the user function (which are initialized to zero). When such changes are made, the user function must explicitly inform NucLear MAC of the changes through a call-back function with selector DATA_WINDOW_CHANGED. This allows the NucLear MAC to perform internal housekeeping, which includes invalidating caches and recomputing the maximum pixel value for word-mode images (*******and scaling them to byte-mode if possible*******). This notification must occur before the user function returns and before any other call-back function is used.

9.3.3 Graphs

User function-created graphs are identified by GraphPtrs, defined by

```
typedef void *GraphPtr;
```

The non-type `void` is used because user functions should not need to directly access graph structures; user functions supply all the parameters necessary to create graphs.

9.3.4 ROIs

The following structures are created by NucLear MAC to describe ROIs to user functions:

```
typedef struct {
  long reserved[4];/* Internal use only! */
  Rect bounds;     /* Boundary in pixels of region */
  short rowbytes;  /* Row bytes in regionflags = */
                /* bounds.right-bounds.left + 3 & 0xfffc */
  unsigned char regionflags[];
                   /* Array of bytes, one per pixel, */
                   /* with non-zero value if the pixel */
                   /* is in the ROI, and zero value if */
                   /* not. */
} ROIRecord, *ROIPtr;

typedef struct {
   long reserved[6];  /* Internal use only! */
   short number_ROIs; /* Number of ROIPtrs in following */
   ROIPtr theROIs[];  /* Array of ROIPtrs */
} ROIArrayRecord, *ROIArrayPtr;
```

The bounds field of the `ROIRecord` describes a sub-rectangle of an image boundary. An image boundary is always described by the rectangle (0,0) (`hdim`,`vdim`), thus the following conditions hold for ROI bounds:

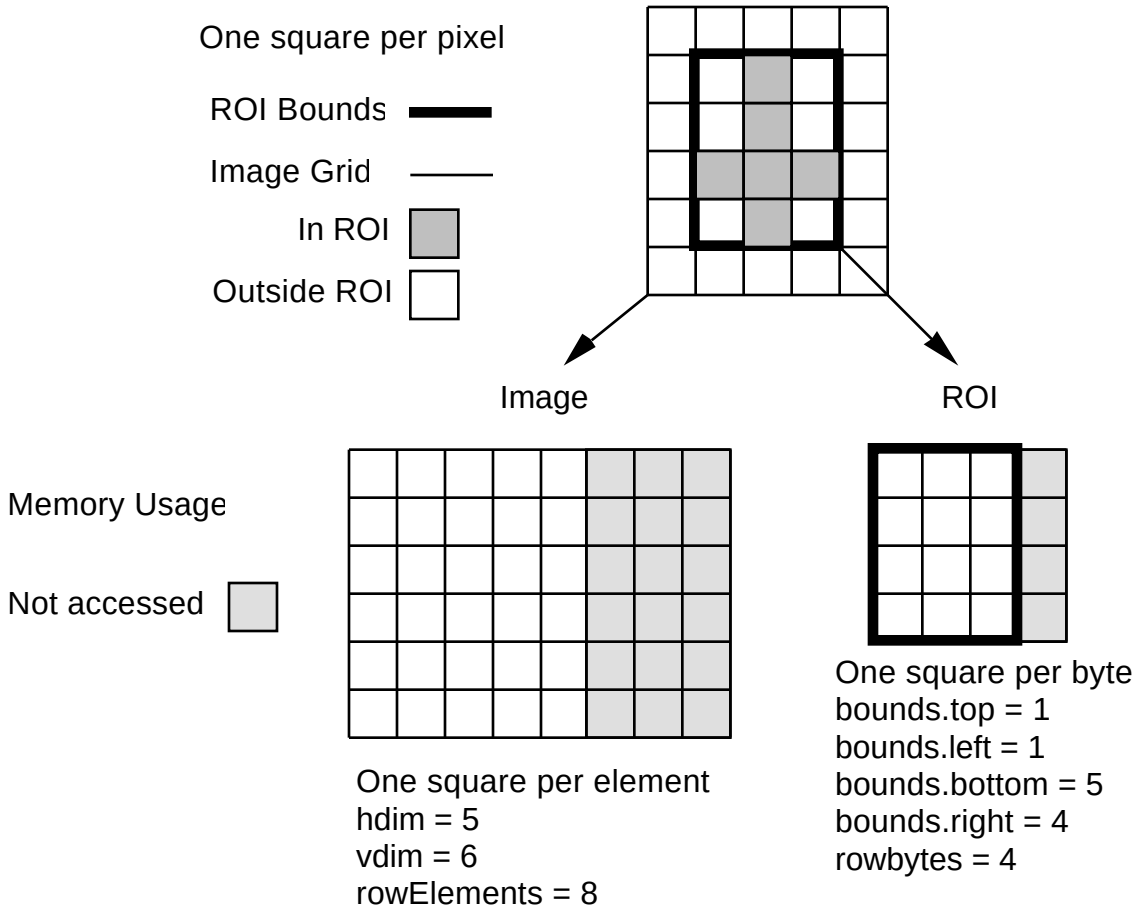
```
0 ≤ bounds.left ≤ bounds.right ≤ hdim

0 ≤ bounds.top ≤ bounds.bottom ≤ vdim
```

As with image data, the ROI mask data has the restriction of rounding its row pixels up to a multiple of four. The total size of the `regionflags` array in bytes is found from:

```
(long)(bounds.bottom - bounds.top) * rowbytes
```

Data in the `regionflags` array is set such that a zero value means that the corresponding pixel in the image is not in the ROI, whereas a non-zero value implies the pixel is in the ROI. It may be assumed that flags in an ROI `regionflags` array that lie outside the bounds rectangle are zero.

We conclude this section with an example showing a 3 x 4 ROI in a 5 x 6 image. Note that memory usage is from left to right, then top to bottom.



Memory usage

9.4 User Function File

All files in the Macintosh file system have identifying long-word parameters known as file type and file creator. The file type is used to identify the format of a file. The file creator identifies the application program associated with a file. In order for NucLear MAC to recognize a file as containing a user function, the file type must be set to ‘NUCG’, and the file creator must be ‘NUCM’. Most development systems can be instructed to automatically place this information after compilation.

Macintosh files have two locations, or forks, where information can be stored. These are termed the resource fork and the data fork. A user function file uses only the resource fork. The resource fork of a file is essentially an unsophisticated database. Individual pieces of information, or resources, are stored and retrieved according to their types and unique identifying integers. A resource type is analogous to the file type, in that it can identify the format of information stored as that type.

Two resources are required to be in a NucLear MAC user function file. The most important is a resource containing the subroutine(s) that implements the user function. The convention is for this resource to have a type of ‘NUCG’ and an identifying number that depends on the type of code. This resource consists of the compiled user function code with an entry point at the top. As described in the User Function Subroutine section, the subroutine obeys C calling conventions, accepts a single argument containing the address of a call-back procedure, and returns a short integer indicating the success or failure of the call.

There are currently different versions of NucLear MAC to support three types of computer code. These exist to support the various Macintosh computing environments. As a result, it may be necessary to include more than one ‘NUCG’ resource, depending on the intended environments for the user function. The currently defined ‘NUCG’ resources are as follows:

‘NUCG’ #128

Supply this resource for the original NucLear MAC. This resource consists of routines compiled into MC68020 op-codes. A math coprocessor (MC68881, etc.) is assumed and the size of a double real number is 12 bytes. Set the user function compiler to generate MC68020 and MC68881 instructions. In THINK C, enable native floating-point format.

‘NUCG’ #129

This resource is used with NucLear MAC SANE / PPC Emulation. This version will run on all Macintoshes, but it is the slowest because a math coprocessor is not assumed. Rather, numeric calculations are performed in the Standard Apple Numerics Environment (SANE). The size of a double real number is 10 bytes. Set the user function compiler to generate MC68020 instructions with no coprocessor access. In THINK C, enable native floating-point format.

‘NUCG’ #130

This resource is used with NucLear Power, which only runs in native mode on Power Macintoshes. In technical terms, this resource is a code fragment, so the requirement of a top entry point is dropped. The size of a double real number is eight bytes. Make sure that your user function is the starting entry point.

Note that ideally all three 'NUCG' resources should be included in a user function file. When all resources are present, the user function can safely be given to any other NucLear MAC user. Accessing a user function that lacks the resource for the Macintosh it is running on will result in a "Resource not found" error message.

The user function file also must contain a resource of type 'TEXT' with an identifying number of 128. This resource contains, as an ASCII C string, a concise description of what the user function does, and how to use it. Future versions of the NucLear MAC will display this information when the user requests help or additional information regarding the user function.

Most development systems can be configured to automatically configure a code resource as described above. A resource compiler or a resource editor like Apple's ResEdit can be used to create the help 'TEXT' resource. When developing native code for Power Macintoshes, you may need a utility to convert a shared library file into a 'NUCG' resource. This utility simply reads the data fork of a file and places the contents into a 'NUCG' resource.

9.4.1 Macros as User Functions

There is a special type of user function file that contains an execute-only macro. This type of user function file is created automatically when a macro is exported as a user function. A macro-containing user function lacks the 'NUCG' resources. Instead the macro text is stored in a 'TEXT' resource with an ID of 129. This kind of user function allows you to run a macro by simply opening the user function file.

9.5 Call-Back Functions

In this section, all currently defined call-back routines are described. In this discussion we shall assume that the macroing of the User.h file is used. The macros make the call-back routines appear to be standard C functions, when in fact they cause expansions such as

```
dp = PromptForDataWindow("\pClick on a window");
```

expanding to

```
dp = (DataWindowPtr)(*callbackproc)(PROMPT_FOR_DATA_WINDOW,
 (unsigned char *)("\pClick on a window"));
```

Note the use of '\p' in this example. Most of the call-back functions require text in the form of Pascal strings (which consist of a length byte followed by up to 255 characters). The '\p' directive causes a Macintosh C compiler to generate a Pascal rather than a C string.

define FRONT_DATA_WINDOW 1

```
DataWindowPtr FrontDataWindow(void);
```

If the front window contains image data, this function returns a pointer to a data structure describing the image data. If the front window doesn't contain image data or there is insufficient memory, then a value of NULL (0L) is returned.

define PROMPT_FOR_DATA_WINDOW 2

```
DataWindowPtr PromptForDataWindow(
  unsigned char *promptstring);
```

In response to this call, NucLear MAC puts up a window containing the Pascal string pointed to by `promptstring`. If the user clicks on a window containing image data, then this function returns a pointer to a data structure describing the image data. If the user chooses a different action or there is not enough memory, then a value of NULL (0L) is returned.

define NEW_DATA_WINDOW 3

```
DataWindowPtr NewDataWindow(
  unsigned char *namestring,
  short nframes,
  short hdim,
  short vdim,
  short wordmode,
  short format);
```

This call allows the user function to create a new image data window with the given parameters. `namestring` is a pointer to a Pascal string containing the title the window should be given. `nframes` is a short integer telling how many stacked images should be contained in the window. `hdim` is the horizontal dimension of the image(s). `vdim` is the vertical dimension of the image(s). If the `wordmode` parameter is zero (0), the image data will consist of one byte per pixel. If the `wordmode` parameter is one (1) then two bytes per pixel are allocated, with the high order byte stored first. If the `format` parameter is zero (0), the image window will be displayed in stacked format. When `format` is non-zero the image data is displayed in an all-frames window. If there is sufficient memory, then the function returns a pointer to a data structure describing the image data, otherwise NULL (0L) is returned. The data structure must be treated as read-only. The image data, however, can be set arbitrarily by the user function. Any time the image data is changed, the function `DataWindowChanged` must be called prior to the return of the user-function or any other call to the call-back routines. Other restrictions on `NewDataWindow` are: $32 \leq$ `hdim` ≤ 4096, $32 \leq$ `vdim` ≤ 4096, `nframes` ≤ 1000.

define GET_DATA_WINDOW_NAME 4

```
long GetDataWindowName(
  DataWindowPtr window,
  unsigned char *namestring);
```

Given a DataWindowPtr, this routine returns the name of the data window in the Pascal string pointed to by `namestring`. The function returns one (1L) if successful or NULL (0L) if the `DataWindowPtr window` is not recognized.

define DATA_WINDOW_CHANGED 5

```
long DataWindowChanged(
  DataWindowPtr window);
```

This routine must be called after data in an image window is altered. Among other things, it does internal housekeeping and sets flags indicating that the window should be updated. The function returns one (1L) if successful or NULL (0L) if the `DataWindowPtr window` is not recognized. **NOTE!** A side effect of this call is the conversion of a word-mode window to byte-mode if no pixel value is greater than 255.

define GET_WINDOW_ROI 6

```
ROIPtr GetWindowROI(
  DataWindowPtr window);
```

If a single ROI is selected in the image window described by the `window` parameter then a pointer to a data structure describing the ROI is returned. The data structure must be treated as read-only. If `window` is not recognized, or no region is selected, or there is insufficient memory, then a value of NULL (0L) is returned. Note that the flags array created by this function only covers the rectangle described in the bounds field of the `ROIRec`.

define PROMPT_FOR_ROI 7

```
ROIPtr PromptForROI(
  DataWindowPtr window,
  unsigned char *promptstring);
```

In response to this call, NucLear MAC puts up a window containing the Pascal string pointed to by `promptstring`. If the user draws an ROI in the designated window and clicks the "OK" button in the prompt window, this function returns a pointer to a data structure describing the ROI. The data structure must be treated as read-only. If the `window` parameter is not recognized, or the user chooses "Cancel", or there is insufficient memory, then NULL (0L) is returned. Note that the flags array created by this function only covers the rectangle described in the bounds field of the `ROIRec`.

define CHANGE_FRONT_FRAME 8

```
long ChangeFrontFrame(
  DataWindowPtr window,
  short newframe);
```

This call changes the visible frame of the window described by the `window` parameter to `newframe`. A value of one (1L) is returned when the call is successful. NULL (0L) is returned if `window` is not recognized, `newframe` is less than zero, or `newframe` is greater than or equal to the `nframes` field of `window` (`window->nframes`).

define TEXT_TO_ANALYSIS_WINDOW 9

```
long TextToAnalysisWindow(
  unsigned char *thetext,
  short textlength);
```

This call causes the text pointed to by `thetext` to be sent to the NucLear MAC Analysis window. The `textlength` parameter tells how many characters to send. A value of one (1L) is returned by the function. If any part of the Analysis window is not visible, the window is brought to the front.

define STRING_TO_MESSAGE 10

```
long StringToMessage(
  unsigned char *thestring);
```

This call puts up a modal dialog box containing the Pascal string pointed to by `thestring`. A value of one (1L) is returned.

define PROMPT_FOR_FRAME 11

```
long PromptForFrame(
  DataWindowPtr window,
  unsigned char *promptstring);
```

In response to this call, NucLear MAC puts up a window containing the Pascal string pointed to by `promptstring`. A mode is entered where, by moving the mouse horizontally, the user can change the visible frame in `window`. The function returns with a value of one (1L) after the user clicks the mouse or presses a key. The function returns NULL (0L) if `window` is not recognized. Inspect the `window->frontframe` field to determine the frame that was chosen.

define PROMPT_FOR_NUMBER 12

```
long PromptForNumber(
  unsigned char *promptstring,
  double *thenumber);
```

In response to this call, NucLear MAC puts up a modal dialog box containing the Pascal string pointed to by `promptstring`. The real number pointed to by `thenumber` is entered as a default in an editable text box in the dialog box. The user is then free to change the number. If the user clicks “OK” then the new number is stored in the location pointed to by `thenumber` and the function returns one (1L). If the user chooses “Cancel” the function returns NULL (0L). Note that sizeof(`double`) can be eight, 10, or 12 bytes.

define PROMPT_FOR_STRING 13

```
long PromptForString(
unsigned char *promptstring,
unsigned char *thestring);
```

In response to this call, NucLear MAC puts up a modal dialog box containing the Pascal string pointed to by `promptstring`. The Pascal string pointed to by `thestring` is entered as a default in an editable text box in the dialog box. The user is then free to change the string. If the user clicks "OK" then the new string is stored in the location pointed to by `thestring` and the function returns one (1L). If the user chooses "Cancel" the function returns NULL (0L).

define PROMPT_FOR_POINT 14

```
long PromptForPoint(
  DataWindowPtr window,
  unsigned char *promptstring,
  Point *thepoint);
```

In response to this call, the NucLear MAC puts up a window containing the Pascal string pointed to by `promptstring`. If the user clicks the mouse in the window designated by the `window` parameter the function places the local coordinates of the click in the Point structure pointed to by `thepoint`, and then returns one (1L). If `window` is not recognized, the user clicks outside of the designated window or presses any key, then the function returns NULL (0L).

define PLACE_ROI 15

```
long PlaceROI(
  DataWindowPtr window,
  short frame,
  RgnHandle theRgn);
```

This call instructs NucLear MAC to place the region designated by `theRgn` in the frame of the window indicated by the `window` and `frame` parameters. If the call is successful, NucLear MAC appropriates the region passed for its own use and returns one (1L). If you need to further manipulate the region you should operate on a copy generated by such Macintosh system calls as `CopyRgn` or `HandToHand` (you should have no references to `theRgn` after a successful call to `PlaceROI)`. If you want to access a bitmap of the region, then simply follow this function call with a call to the `GetWindowROI` macro. If `window` is not valid or `theRgn` is not valid or there is insufficient memory or `frame` is out of range then the function returns NULL (0L). Note that a reasonable amount of Macintosh-specific knowledge is required to use this function. A `RgnHandle` is a pointer to a pointer to a structure whose format is not publicly documented by Apple. `RgnHandles` are created indirectly through calls to the Macintosh Toolbox.

define DO_QUERY 16

```
long QueryUser(
  unsigned char *promptstring,
  unsigned char *OKstring,
  unsigned char *Cancelstring);
```

In response to this call, NucLear MAC puts up a modal dialog box containing the Pascal string pointed to by `promptstring`. There are two buttons in the dialog box. In the lower right is the OK button, which is the default choice, selected by typing return or enter. Immediately to the left is the Cancel button, selected by typing command-period or escape. The `OKstring` and `Cancelstring` parameters point to Pascal strings that are used as titles for the OK and Cancel buttons respectively. For example, to label the buttons “OK” and “Cancel,” your call might look like this: `QueryUser("\pDo you want to do this?","\pOK","\pCancel")` This function returns one (1L) if the OK button is selected, or zero (0L) otherwise.

define GET_FRAME_DURATION 17

```
long GetFrameDuration(
  DataWindowPtr window,
  short frame);
```

This call returns the time, if any, in milliseconds, associated with the frame and window specified by their respective parameters. If `window` is not a data window, `frame` is out of range, or there is no timing information, then zero (0L) is returned. `frame` ranges from zero to (`window->nframes - 1`).

define GET_FRAME_PAUSE 18

```
long GetFramePause(
  DataWindowPtr window,
  short frame);
```

This call returns the pause time, if any, in milliseconds, associated with the frame and window specified by their respective parameters. If `window` is not a data window, `frame` is out of range, or there is no pause, then zero (0L) is returned. `frame` ranges from zero to (`window->nframes - 1`).

define NEW_GRAPH 19

```
GraphPtr NewGraph(
  unsigned char *graph_title,
  unsigned char *x_title,
  unsigned char *x_units);
```

This call creates a new graph with the title, x title and x units specified. `graph_title`, `x_title`, and `x_units` are pointers to Pascal strings. The `GraphPtr` returned by `NewGraph` is used to specify the graph in other calls where the axes are modified and data sets are added. If there is not enough memory to make the graph, the `NewGraph` returns NULL (0L). `x_title` and `x_units` will be truncated to 27 characters.

define SET_GRAPH_X_LIMITS 20

```
long SetGraphXLimits(
  GraphPtr theGraph,
  double xmin,
  double xmax,
  short tics,
  short num_format);
```

The default condition for graphs is that the x limits and number of tic marks are automatically calculated. Use this call if you wish to override this behavior. `theGraph` specifies a graph previously created with `NewGraph`. `xmin` and `xmax` specify the new display limits for the x axis. `tics` specifies how many tic marks to place along the x axis. `num_format` is a short integer that determines whether the numbers drawn along the x axis are in fixed point or floating point format. A positive value is used for fixed point; the magnitude of `num_format` tells how many digits are placed after the decimal mark. If `num_format` is zero, then numbers are written as integers, rounding if necessary. If `num_format` is negative, then numbers are written in floating point format, with the number of significant digits equal to the absolute value of `num_format`. `num_format` should be in the range ($-10 \leq$ `num_format` ≤ 10). If `theGraph` is not recognized, or `xmin` is greater than or equal to `xmax`, then `SetGraphXLimits` returns zero (0L), otherwise one (1L) is returned.

define ADD_GRAPH_DATA_SET 21

```
long AddGraphDataSet(
  GraphPtr theGraph,
  unsigned char *y_title,
  unsigned char *y_units,
  double *xdata,
  double *ydata,
  short npoints,
  short point_type,
  short connect,
  short axis);
```

This call adds a data set to an existing graph created with `NewGraph` and specified by `theGraph`. The parameters `y_title` and `y_units` point to Pascal strings that are used to label the data set. The data set is specified by `xdata`, `ydata`, and `npoints`. `xdata` and `ydata` point to arrays of real numbers. Note that the size of the real numbers is 8, ten or twelve bytes. The number of entries in each array is given by the `npoints` parameter. If this is not the first data set added to the graph, then you can substitute NULL (0L) for the `xdata` term, provided this data set has the same number of entries as the first data set, and these y values correspond to the same x values as the first set. (This feature saves time and memory) `point_type` determines what symbol should be drawn for each point. The choices are given by the following enumeration:

```
typedef enum {
```

```
    CIRCLE_PTD=0,
    PLUS_PTD,
    X_PTD,
    TRI_PTD,
    SQUARE_PTD,
    DOT_PTD,
    BAR_PTD

  } PointDrawType;
```

The `connect` parameter specifies whether lines should be drawn between the points: zero (0) for no lines, one (1) for lines. The `axis` parameter determines whether the point set will be scaled to the primary or secondary y axis. For the first data set of a graph, `axis` is ignored, and the data set becomes the primary y axis. For subsequent data sets, when `axis` is zero (0), the set being added is scaled to the primary axis. When `axis` is one (1), the data set either becomes the secondary y axis, if none exists, or else is scaled to the existing secondary y axis. `AddGraphData` returns one (1L) if successful or zero (0L) otherwise. Reasons for failure include insufficient memory, an unrecognized graph, or other bad parameters. Note that the `xdata` and `ydata` arrays must be sorted so that `xdata` proceeds from lowest to highest.

define SET_GRAPH_Y_LIMITS 22

```
  long SetGraphYLimits(
    GraphPtr theGraph,
    double ymin,
    double ymax,
    short tics,
    short num_format,
    short axis);
```

The default condition for graphs is that the y limits and number of tic marks are automatically calculated. Use this call if you wish to override this behavior. `theGraph` specifies a graph previously created with `NewGraph`. `ymin` and `ymax` specify the new display limits for the y axis. `tics` specifies how many tic marks to place along the y axis. `num_format` is a short integer that determines whether the numbers drawn along the y axis are in fixed point or floating point format. A positive value is used for fixed point; the magnitude of `num_format` tells how many digits are placed after the decimal mark. If `num_format` is zero, numbers are written as integers, rounding if necessary. If `num_format` is negative, numbers are written in floating point format, with the number of significant digits equal to the absolute value of `num_format`. `num_format` should be in the range ($-10 \leq$ `num_format` ≤ 10). `axis` determines whether the call affects the primary or secondary axis. Pass zero (0) for the primary axis and one (1) for the secondary axis. If `theGraph` is not recognized, or `ymin` is greater than or equal to `ymax`, or there are no data sets in the graph, or `axis` is one when there is no secondary axis, then `SetGraphYLimits` returns zero (0L), otherwise one (1L) is returned.

define GET_ALL_WINDOW_ROIS 23

```
ROIArrayPtr GetAllWindowROIs(
  DataWindowPtr window,
  short only_in_visible_frame);
```

This call returns a pointer to an array of ROIPtrs describing ROIs in the window specified by the `window` parameter. The `only_in_visible_frame` parameter determines whether all visible ROIs in the window are described (`only_in_visible_frame` == 0) or only those that are visible in the active, visible frame of the window (`only_in_visible_frame` == 1). If there is not enough memory to complete the array, or `window` is invalid, then NULL (0L) is returned instead of the `ROIArrayPtr`. Note that this array is not dynamic; if a user function causes more ROIs to be placed, they are not added to existing `ROIArrayRecords`.

define GET_ROI_NAME 24

```
long GetROIName(
  ROIPtr theROI,
  unsigned char *name);
```

This call returns the name of the ROI described by `theROI` in the Pascal string pointed to by the `name` parameter. A successful call returns one (1L). If `theROI` is invalid then zero (0L) is returned.

define SET_ROI_NAME 25

```
long SetROIName(
  ROIPtr theROI,
  unsigned char *name);
```

This call changes the name of the ROI described by `theROI` to the Pascal string pointed to by the `name` parameter. A successful call returns one (1L). If `theROI` is invalid then zero (0L) is returned. `name` is truncated to 27 characters.

define PUT_GRAPH_INFO 26

```
long PutGraphInfo(
  GraphPtr theGraph,
  unsigned char *text,
  short textlength);
```

This call enters the text given by the `text` and `textlength` parameters into the information field of the graph specified by `theGraph`. If `theGraph` is not a graph made from `NewGraph` then zero (0L) is returned, otherwise one (1L) is returned.

define COUNT_ROIS 27

```
long CountROIs(
  DataWindowPtr window,
  short frame);
```

This call counts and returns the visible ROIs in the window and frame described by the `window` and `frame` parameters. Zero (0L) is also returned if `window` is not a valid `DataWindowPtr`, `frame` is less than zero, or `frame` is greater than or equal to `window->nframes`.

define PROMPT_FOR_EXISTING_ROI 28

```
ROIPtr PromptForExistingROI(
  DataWindowPtr window,
  unsigned char *promptstring);
```

In response to this call, NucLear MAC puts up a window containing the Pascal string pointed to by `promptstring`. If the user selects an existing ROI in the designated window and clicks the OK button in the prompt window, this function returns a pointer to a data structure describing the ROI. The data structure must be treated as **read-only**. If the `window` parameter is not recognized, or the user chooses “Cancel,” or there is insufficient memory, then NULL (0L) is returned. Note that the flags array created by this function only covers the rectangle described in the bounds field. This call is problematic in that it allows the user to move and resize existing ROIs as well as select them. The `ROIRecord` is a static structure which cannot be changed. Hence, if an ROI is altered and it has an `ROIRecord`, that record should be marked obsolete and a new record allocated. In order for you to update the ROI data, you should call `GetUpdatedROI`, described below.

define GET_UPDATED_ROI 29

```
ROIPtr GetUpdatedROI(
  ROIPtr theROI);
```

Use `GetUpdatedROI` when you have called `PromptForExistingROI` in order to be sure the `ROIRecord` you are using corresponds to the ROI the user may have edited. The general rule is to use `GetUpdatedROI` on any `ROIPtr` that came from the window prior to calling `PromptForExistingROI`. These `ROIPtrs` could come from `GetWindowROI`, `PromptForROI`, `GetAllWindowROIs, or PromptForExistingROI`. `GetUpdatedROI` examines the `ROIRecord` to see whether the ROI still corresponds to the flags. If it does, `GetUpdatedROI` simply returns the same `ROIPtr`. If the ROI doesn't match the existing record, `GetUpdatedROI` allocates a new `ROIRecord`, fills it out, and returns a pointer to it. In this latter case the original `ROIRecord` is marked obsolete, and calls using an `ROIPtr` to it (other than `GetUpdatedROI`) will fail. If `theROI` is not recognized or there is not enough memory to allocate a new record, `GetUpdatedROI` returns NULL (0L). In some situations you may intentionally avoid calling `GetUpdatedROI` so that your calculations will operate on the ROIs that were present prior to calling `PromptForExistingROI`. You can determine whether an ROI has changed by comparing the original `ROIPtr` to the one returned by `GetUpdatedROI`.

define PUT_FRAME_TEXT 30

```
long PutFrameText(
  DataWindowPtr window,
```

```
   short frame,
   unsigned char *thestring,
   short inallframes);
```

`PutFrameText` is used to overlay text in an image frame. The text is placed in the top, left corner of the image. Because of this, it is recommended that you only use this call on image windows you created with `NewDataWindow`. Otherwise, your text may overlap existing text overlays. The window and frame are specified by the `window` and `frame` parameters respectively. `thestring` should point to a Pascal string containing the text to overlay. The parameter `inallframes` determines whether the text is drawn in all frames of the image window, or only the specified frame. Set `inallframes` to one (1) for all-frames overlay or zero (0) for single-frame overlay. `PutFrameText` returns one (1L) if successful, or zero (0L) if `window` is not recognized, `frame` is less than zero, or `frame` is greater than or equal to `window->nframes`.

define COUNT_ARGUMENTS 31

```
 long CountArguments(void);
```

`CountArguments` returns the number of arguments that were passed to the user function. The arguments can come from a macro.

define GET_ARGUMENT_TYPE 32

```
 long GetArgumentType (
   short argument_number,
   short *typeptr)
```

`GetArgumentType` returns the type of the argument specified by `argument_number`. `argument_number` must be greater than zero and less than or equal to the value returned by `CountArguments`. User functions can recognize the following types:

T_IMAGE, T_NUMBER, T_STRING, T_GRAPH, and T_REGION

`GetArgumentType` returns one (1) if the type was successfully placed in the location pointed to by `typeptr`, and zero (0) otherwise.

define GET_ARGUMENT 33

```
 long GetArgument(
   short argument_number,
   void *where)
```

`GetArgument` returns the argument specified by `argument_number`. `argument_number` must be greater than zero and less than or equal to the value returned by `CountArguments`. You should have called `GetArgumentType` previously to be certain the argument returned is what you expect. Here are the types that can be returned:

type	'where' must point to	size
T_IMAGE	DataWindowPtr	4 bytes
T_NUMBER	real double	8, 10, or 12 bytes-depends on version
T_STRING	Str255	256 bytes maximum: length byte, string
T_REGION	ROIPtr	4 bytes

`GetArgument` returns one (1) if the specified argument was placed in the location pointed to by `where`, and zero (0) otherwise.

define RETURN_ITEM 34

```
long ReturnItem(
  short argument_type,
  void *where)
```

`ReturnItem` adds an item to the list of items to be returned by the user function. Use the following list as a guide to possible argument pairs:

argument_type	pass in 'where' parameter	size
T_IMAGE	DataWindowPtr	4 bytes
T_NUMBER	pointer to real double	4 bytes
T_STRING	pointer to Str255	4 bytes
T_GRAPH	GraphPtr	4 bytes
T_REGION	ROIPtr	4 bytes

`ReturnItem` returns one (1) if the specified item was added to the return list, and zero (0) otherwise.

10 Menu Items Summary

10.1 File

Import Format…
Activates a dialog box that allows you to define file structures for importing. The parameters can be saved for future use. After the parameters are set, they can be applied to a file when it is imported with the **Open…** selection. Except for the case of Interfile, only the image information is translated; any patient, study, etc. information is ignored.

New File Group…
Saves a file containing the names and screen locations of a group of images and graphs. The images and graphs must have been previously saved in the location where you save the group file. When you open the group file later, the images and graphs are automatically opened and moved to their original screen locations.

Open…
Lets you open images, graphs, and clinical protocols. Enable the Import check box to show all files, and to select the import protocol to use for unrecognized files.

Close
Closes the active window.

Close All
Closes all open windows, except palette windows. Hold down the option key to close only all the palette windows.

Save
Saves the active window. If the window was not previously saved, this command is the same as selecting **Save As…**

Save As…
Prompts you for where and under what name to store the contents of the active window.

Export As…
Allows the active window to be exported in a different format than the standard NucLear MAC format, e.g., PICT, TIFF, Interfile, TEXT. This is usual for sharing data with other programs. Because the export formats don't retain all information used by NucLear MAC, you should also save the window with the **Save As…** command if you plan to use the data again.

Save Protocols As…
Allows you to make a file containing selected acquisition and processing protocols. The protocols file can be moved to another machine, and when it is opened using the **Open…** command from the NucLear MAC program, it loads all of the protocols on to

the new machine. This command is also useful in saving protocols for replacement if the NucLear MAC Prefs file becomes damaged.

Page Setup…
Lets you set the page orientation and other parameters for printing the active window on the current printer.

Print…
Prints the active window on the current printer.

Print Options…
Brings up a dialog allowing you to set global printing parameters.

Quit
Quits the program.

10.2 Edit

Undo or **Redo**
Reverses the previous command in the active window. When you perform an action, you can use the undo command to reverse it. For example, if you cut a region of interest from an image, the text of the undo command changes to **Undo Cut**, and selecting it places the region back on the image and restores the clipboard. The undo command text then changes to **Redo Cut**.

Cut
Removes the selected item(s) from the active window and places it (them) in the clipboard for later use with the **Paste** command. Items that can be cut include text and arrow objects, text selections, and regions of interest.

Copy
Copies the selected item(s) from the active window and places the copy in the clipboard for later use with the **Paste** command. Items that can be copied include text and arrow objects, text selections, and regions of interest.

Paste
Copies the contents of the clipboard into the active window.

Clear
Removes the selected item(s) from the active window. Items that can be cleared include text and arrow objects, text selections, and regions of interest.

Select All
Selects all items in the active window. In a text window, all of the text is selected. In a study window, all of the visible and unlocked objects (text, arrows, ROIs, etc.) are selected.

Font (Submenu)
This submenu shows a selection of the fonts available in the system. Select the font desired for the current text selection or entry point.

Size (Submenu)
This submenu allows selection of different font sizes for the current text selection or entry point. The sizes are given in points.

Style (Submenu)
This submenu allows selection of different text styles for the current text selection or entry point.

Objects > (Submenu)
These commands operate only on the active study window.

Hide
Hides the selected objects.

Hide All
Hides all of the objects in the current frame.

Show All
Shows all previously hidden objects.

In All Frames
Displays the selected object(s) in all frames of the study window.

In Home Only
Displays the selected object(s) in only their original frame(s) of the study window.

Lock
Locks the selected object(s) so it (they) cannot be moved or deleted.

Unlock Frame
Unlocks all previously locked objects in the current frame.

Set Color (Submenu)
Sets the color of the selected object(s).

Show Axes or **Hide Axes**
Hides axes that are visible or shows axes that are hidden in the active study window.

Information > (Submenu)
These commands operate only on the active study window.

Patient...
Lets you enter and display patient data.

Study, General…
Lets you enter and display general study information.

Study, Specific…
Lets you modify and display study-specific data information.

Location…
Lets you enter and display data regarding the clinical site.

Color Table > (Submenu)

Attach
Attaches the current color table to the active study window.

Remove
Removes an attached color table from the active study window.

Display
If Display is enabled (checked), an attached color table is activated when its study window is activated.

Settings…: Lets you define color table sets.

1,2,3,4
Associates an active color table (bar) with the active study window. The study is displayed in the colors of the chosen color bar. A shortcut for this is to command-click on the color bar and then click on the study window(s).

Preferences > (Submenu)

Saving Prompts…
Lets you disable warnings regarding unsaved changes when windows are closed. Also lets you disable automatic saving of window positions.

Object Colors…
Lets you set default colors for text and other objects.

Measuring…
Lets you set the default scale and units of the measuring tool. The scale is used in images lacking calibrated pixel sizes.

Location…
Lets you set clinical site information that is to be stored with image data.

Image Defaults…
Lets you define default behavior for image display (Interpolation, inversion, zooming).

Exporting…
Lets you specify whether Interfile exporting should create separate header and image files or one composite file.

Rotate…
Lets you define parameters for mouse control of the **Rotate** function.

Script Windows…
Lets you set the positions of windows that appear in processing scripts.

All Views…
Lets you set parameters for making All Views Windows.

Analysis Window…
Lets you set amount of text saved in the Analysis window.

Purge Frames
Removes blank frames from the active study window.

10.3 Display

Step Forward
Advances the active movie window by one frame. Can also apply to certain all-angles and all-views windows.

Step Backward
Reverses the active movie window by one frame. Can also apply to certain all-angles and all-views windows.

Run or **Halt**
Run or halts a cinegram of the active movie window. Can also apply to certain all-angles and all-views windows.

Mouse Track
Lets you set the current frame of the active movie window by mouse movement. Can also apply to certain all-angles and all-views windows.

Halt All
Stops all currently running cinegrams.

Run All
Runs all windows containing cinegrams. (Movie windows and certain all-angles and all-views windows).

Step All
Advances all windows containing cinegrams by one frame. (Movie windows and certain all-angles and all-views windows).

Step All Back
Reverses all windows containing cinegrams by one frame. (Movie windows and certain all-angles and all-views windows).

Mouse Track All
Applies the mouse-tracking to all windows containing cinegrams. (Movie windows and certain all-angles and all-views windows).

Mouse Track Some
Applies mouse-tracking to selected windows. When you select this command, the cursor turns into a "hand". Click on the windows to be controlled. End the window-selection process with two clicks on the final window.

Synchronize
Synchronizes selected cinegrams. When you select this command, the cursor turns into a "hand". Click on the windows to be synchronized. End the window-selection process with two clicks on the final window. The selected windows are run, each starting from its first frame.

All Frames Format or **Movie Format**
Converts a window in movie format into all-frames format. Alternatively, converts a window in all-frames format into movie format. Does not apply to windows with only one image frame.

All Views Window…
Creates an All-Views window based on one or more source study windows. The active study window is used first, and you are prompted for additional windows to use, if any. In a typical case, you would apply this command to the window containing stress short-axis slices and then double-click on the window having the rest short-axis slices.

All Angles Window
Makes an All-Angles window from the active study. You should apply this command to gated or non-gated reconstructed data having either transverse (horizontal long axis) or coronal (short axis) orientation.

Views > (Submenu)
These commands operate only on the active study window. They all create a new study window.

Transverse to Coronal or **H. Long Axis to Short Axis**
Generates coronal SPECT slices from transverse slices.

Transverse to Sagittal or **H. Long Axis to V. Long Axis**
Generates sagittal SPECT slices from transverse slices.

Coronal to Transverse or **Short Axis to H. Long Axis**
Generates transverse SPECT slices from coronal slices.

Coronal to Sagittal or **Short Axis to V. Long Axis**
Generates sagittal SPECT slices from coronal slices.

Set View Angle/VR
Activates interactive volume rendering. Use the mouse to control the angle at which the original data is viewed. Exit by clicking the mouse. The final angle is stored for use by the Reslice From View command.

Set View Angle/SR
Activates interactive surface rendering. Use the mouse to control the angle at which the original data is viewed. Hold down the key to adjust the surface threshold. Exit by clicking the mouse. The final angle is stored for use by the Reslice From View command.

Reslice From View
Generates oblique-axis images from original SPECT images based upon the Set View Angle final position (front to back from point of view).

Other > (Submenu)

Right
Reslices reconstruction study window from the right side.

Left
Reslices reconstruction study window from the left side.

Top
Reslices reconstruction study window from the top.

Bottom
Reslices reconstruction study window from the bottom.

Volume Render
Creates 32 volume rendered projection images of reconstruction study window.

Flip Horizontally
Flips all frames of a study window left-to-right.

Flip Vertically
Flips all frames of a study window top-to-bottom.

Rotate Right 90°
Rotates all frames of a study window clockwise 90°.

Rotate Left 90°
Rotates all frames of a study window counterclockwise 90°.

Double
Doubles the size of an image by pixel interpolation. Differs from normal zooming in that the final image has nearly four times the number of data pixels as the original image. Each dimension is increased to (2n - 1), where n is the original image dimension.

Halve
Halves the size of a study window by pixel averaging.

Byte Scaled
Scales a word-depth study to a byte-depth study using the current scaling. Duplicates a byte-mode study.

Reverse Order
Reverses the order of frames in a study window.

Frames Subset
Generates a new study window with only the selected frames of the original study window (selected by dragging with "floppy disk" tool in controls window).

Frame Normalize
Normalizes all frames of an image to have the same maximum pixel value, either 255 for byte-mode images, or the original maximum pixel value for word-mode images.

Match Cropping To…
Matches the window cropping of the active window to the cropping of a window selected by clicking on it with the "hand" cursor.

Match Scaling To…
Matches the brightness scaling of the active window to the scaling of a window selected by clicking on it with the "hand" cursor.

Extract Sphere
Extracts an image sphere from a study window based on the Y axis length and position of the Axis tool.

Extract Cube
Extracts an image cube from a Movie window based on the Y axis length and position of the Axis tool.

Rotate
Turns on the in-plane rotation function to allow the user to rotate individual or all frames in a study window.

Split Dual
Separates image frames that are in dual-head or dual-energy format. Each single frame is divided into two frames in the resulting study window.

Full Speed
Makes all running cinegrams cycle at maximum speed.

All Frames Snapshot
Converts an All-Frames window into a byte-scaled single-frame window. This process is needed for printing images with some printers.

Interpolate
Turns the interpolation state of the front window on or off. Images are interpolated when they are displayed in a larger format than the actual size of the image data matrix.

Photo Mode
Hides the menu bar and desktop until you click the mouse or press any key.

10.4 Acquire

Any Study...
Lets you enter study information prior to beginning an image acquisition.

Patient Records...
Lets you enter patient records prior to beginning an acquisition.

Single Image...
Brings up the acquisition dialog box for single images.

Gated Images...
Brings up the acquisition dialog box for gated images.

Dynamic Time Series...
Brings up the acquisition dialog box for dynamic images.

Image Sequence...
Brings up the acquisition dialog box for sequential images.

SPECT Series...
Brings up the acquisition dialog box for SPECT images.

Gated SPECT...
Brings up the acquisition dialog box for Gated-SPECT images.

Siemens Whole Body...
Brings up the acquisition dialog box for Siemens RS232 whole body scans.

Truvel Film Scan...
Brings up the acquisition dialog box for digitizing x-ray film images with Truvel scanner.

Vision Ten Film Scan…
Brings up the acquisition dialog box for digitizing x-ray film images with Vision Ten scanner.

Settings…
Lets you change parameters for study acquisitions.

Hardware…
Lets you change hardware settings.

Pixel Sizes…
Lets you calibrate pixel sizes for your camera and specific collimators.

Serial Port…
Lets you set parameters for serial (RS232) communication with camera controllers.

Pause Acquisition
Pauses the data acquisition in progress.

Stop Acquisition
Stops the data acquisition in progress.

Display Updating…
Lets you set the automatic image brightness and update frequency for image acquisitions.

10.5 Regions

Analyze
Displays statistics about the active region of interest. These include total number of pixels, total activity, minimum pixel value, etc.

Histogram
Generates a histogram of the pixel values from the active region of interest.

Frames Plot
Plots the total counts within the active region(s) of interest across specified frames of a study window. The default is for all frames to be used, but by using the "graph" tool in the Controls window you can specify a subset of frames for graphing.

Other Graphs > (Submenu)

Region to Variable
Graphs a curve of data for the selected region of interest across specified frames of a study window. The default is for all frames to be used, but by using the "graph" tool in the Controls window you can specify a subset of frames for graphing. The values used for the X and Y axes are specified in the New Variable dialog. The curve is drawn in

the current destination graph, if one exists. If there is no default destination graph, then a new one is created.

New Variable…
Lets you change the parameters for manual creation of curves. You can also enter X and Y values directly.

Curve Math…
Lets you create new graph curves that are functions of existing curves. The new curves are drawn in the current destination graph, if one exists. If there is no default destination graph, a new one is created.

Equation Plotting
Lets you plot equations or integrals of equations. The new curves are drawn in the current destination graph, if one exists. If there is no default destination graph, then a new one is created.

New Graph
Lets you create a new, empty graph. This graph becomes the default destination for new, manually generated curves.

Select Current
Activates the current curve-destination graph, if one exists.

Make Current
Makes the active graph window the current curve-destination graph.

Row Values > (Submenu)
These commands operate only on a selected rectangular region of interest in the active study window.

Averages
Plots the average row pixel value of the ROI.

Sums
Plots the sum of row pixel values of the ROI.

Y Centroid
Plots the Y centroid of the ROI.

Column Values > (Submenu)
These commands operate only on a selected rectangular region of interest in the active study window.

Averages
Plots the average column pixel value of the ROI.

Sums
Plots the sum of column pixel values of the ROI.

X Centroid
Plots the X centroid of the ROI.

Select > (Submenu)
Selects the chosen region of interest in the active study window. If necessary, the current image frame is changed.

Visibility > (Submenu)
Controls visibility of regions of interest in the active study window.

Hide
Hides the selected regions.

Hide All
Hides all of the regions in the current frame.

Show All
Shows all previously hidden regions.

In All Frames
Displays the selected region(s) in all frames of the study window.

In Home Only
Displays the selected region(s) in only their original frame(s) of the study window.

Delete All
Removes all regions from the study window.

Names > (Submenu)
These commands apply to regions of interest in the active study window.

Change...
Lets you rename the selected region.

Show
Makes visible the names of the selected regions.

Hide
Hides the names of the selected regions.

Show All
Makes visible the names of all regions in the study window.

Hide All
Hides the names of all regions in the study window.

Black
Draws the names of the selected regions in black.

White
Draws the names of the selected regions in white.

Heart > (Submenu)
Lets you perform basic manual ejection-fraction calculations. These commands apply to the selected region of interest in the active study window.

Diastole
Designates the diastolic region.

Systole
Designates the systolic region.

Background
Designates the background region.

Lung > (Submenu)
Lets you perform basic left / right activity quantification. These commands apply to the selected region of interest in the active study window.

Left
Designates the left region.

Right
Designates the right region.

Background
Designates the background region.

Kidney > (Submenu)
Lets you perform basic left / right activity quantification. These commands apply to the selected region of interest in the active study window.

Left
Designates the left region.

Right
Designates the right region.

Background
Designates the background region.

Injected Dose
Designates the injected-dose region.

Clearances…
Lets you compute clearances for DTPA, Hippuran, or MAG3.

First Pass > (Submenu)
Allows ejection fraction calculations from a first pass study window. These commands apply to the selected region of interest in the active study window.

Right Ventricle
Designates the right ventricular region.

Background
Designates the background region.

Utilities > (Submenu)
These commands apply to the selected region of interest in the active study window.

Subtract as Background...
Uses region of interest for subtraction of background activity.

Band...
Generates a band-shaped region from the region of interest.

Polynomial Fit...
Fits a polynomial to the activity in the region.

New Window
Makes a new window based upon a rectangular region.

Detect Edges...
Detects edges within the selected region in the active frame.

Detect Edges/All Frames...
Detects edges within the selected region in all frames.

10.6 Analyze

Image Math > (Submenu)

Set First Term
Makes the active image the first term in an image manipulation. Use this function before the specific image manipulation functions.

Subtract Weighted
Takes the active window and subtracts a weighted value of it from a previously selected first term image. Weight factor is controlled by up - down mouse movement and brightness is controlled by left - right movement. Click to exit the function.

Invert
Inverts the active image.

Complement
Complements the active image

Reframe…
Lets you reframe image frames.

Washout
Computes a washout image based on the first term image and the active image.

Append
Attaches the active movie window to the first term window. The dimensions must match.

Subtract From All
Subtracts the active image from all frames in first term window.

X Concat
Horizontally concatenates the selected image window with the first term window.

Y Concat
Vertically concatenates the selected image window with the first term window.

Regroup…
Lets you regroup or interleave image frames.

Apply Shifts
Applies X and Y shifts from first term window to the active window.

New Composite
Makes an image representing the sum of a group of images from a movie or all-frames image window. The "+" tool in the Controls window is used to select the images to be summed by dragging across the numbers in the top of the Controls window.

Spatial Convolution > (Submenu)
Applies the selected two-dimensional kernel to the active study window.

Temporal Convolution > (Submenu)
Applies the selected Z-axis kernel to the active study window.

Other Filtering > (Submenu)
These commands apply to the active image window.

3D Smooth
Performs three-dimensional unweighted smooth (3 x 3 x 3).

CLUT Filter
Maps images through current color table to produce new images.

Interactive…
Lets you perform interactive two-dimensional filtering in the Fourier domain.

Equation…
Lets you filters images through any equation.

SPECT…
Activates SPECT processing dialog box.

ERPF…
Activates plasma sample analysis.

Renal Functions…
Activates renal region-of-interest analysis.

Bull's-eye Plotting > (Submenu)

Stress Script…
Activates stress bull's-eye plotting script.

Rest Script…
Activates rest bull's-eye plotting script.

Set Apex
Applies to active image window containing short axis slices. Designates the current frame and axis location as the apex of the heart. The Y axis length sets the radius.

Set Base
Applies to active image window containing short axis slices. Designates the current frame and axis location as the center of the base of the heart. The Y axis length sets the radius.

Place Regions
Applies to active image window containing short axis slices. Using the previously-set apex and base locations and sizes, this command places oval profiling regions in all frames from apex to base.

Stress Plot
Applies to active image window containing stress short axis slices. Generates a bull's-eye plot based on existing regions and designated apex and base. The current settings determine the type of profiling.

Rest Plot
Applies to active image window containing rest short axis slices. Stress plot generated previously must still be open. Generates several bull's-eye plots based on comparisons to Stress and existing regions and designated apex and base. The current settings determine the type of profiling.

Surface Plot
Applies to active image window containing short axis slices. Generates a surface profile plot based on existing regions and designated apex and base. The current settings determine the type of profiling.

Place Mask
Applies to active image window containing raw SPECT projections. Masks top and bottom of image window to facilitate motion correction. Mask size is specified in Bull's-eye Settings dialog.

Databases…
Lets you create and edit databases for bull's-eye plotting.

Settings…
Lets you set the type of profiling used for bull's-eye plotting as well as other script parameters.

Overlay…
Lets you edit overlays that can be applied to finished bull's-eye plots.

Planar Thallium > (Submenu)

Run Script…
Activates planar thallium processing script.

Stress Profile
Creates a stress profile based on the region and axes in the active image window.

Rest Profile
Creates a rest profile based on the region and axes in the active image window. Adds the profile to the stress profile and generates a washout curve.

Settings…
Lets you set the type of profiling used for Planar Thallium processing and other script parameters.

First Pass…
Activates the first pass processing script.

Gated Cardiac > (Submenu)

Run Script…
Activates the gated cardiac processing script.

Settings…
Lets you change parameters for the gated cardiac script.

Phase & Slope > (Submenu)

Analyze
Performs temporal Fourier analysis on the active image window. Yields amplitude and phase images and a phase histogram.

Settings…
Lets you change parameters for phase and slope calculations.

Slopes
Generates slope images from source gated images.

Flood Correct…
Flood corrects the active image window. After you select this command, you are instructed to click on the flood image.

Shunt…
Activates the shunt quantification script.

Quality Assurance > (Submenu)

Run Script…
Activates the QA processing script. Expects you to have a QA flood available.

Analyze Region
Performs uniformity analysis of the selected region in the active image window.

Settings…
Lets you change the parameters used for uniformity analysis.

COR > (Submenu)

Auto-Scale Script…
Activates the Auto-Scale processing script. Expects you to have the COR calibration images available. The best study is a 360° SPECT series with an even number of views, using a line source near the center of the field of view. Computes the COR value to be used with all auto-scaled SPECT acquisitions.

Graph X & Y
Computes COR offsets from the active SPECT series of a point or line source.

Settings…
Lets you change the parameters used for uniformity analysis.

User Functions > (Submenu)

Set Directory…
Lets you set the directory where active user functions are located. The names of the active user functions are placed in this submenu. Select the name of the user function you wish to run. User functions are separate pieces of code that add functionality to

NucLear MAC. Any programmer with an appropriate C compiler can write user functions.

Gated SPECT > (Submenu)

Run Script…
Activates gated-SPECT processing script. Expects you to have raw gated-SPECT projections available.

Compute EF/Auto…
Computes ejection fraction semi-automatically based on short-axis, gated-SPECT reconstruction data.

Compute EF/Manual…
Computes ejection fraction based on short-axis, gated-SPECT reconstruction data. You are prompted to draw regions that define the heart wall at end-systole and end-diastole.

Surface Plot…
Displays a surface plot of the short-axis, gated-SPECT reconstruction data.

Script Settings…
Lets you change the parameters used for the gated-SPECT processing script.

Surface Plot Settings…
Lets you change the parameters used for surface plotting.

Attenuation Correct…
Activates attenuation correction script. Use on reconstructed SPECT data.

Decay Correct…
Corrects images for decay of radionuclides.

10.7 Windows

Tools
Activates the Tools window.

Controls
Activates the Controls window.

Colors
Activates the Colors window.

Analysis
Activates the Analysis window.

Kernels
Activates the Kernels window.

Progress
Activates the Progress window.

Custom Colors
Activates the Custom Colors window.

10.8 Macros

Create > (Submenu)

Record New or **End Recording**
Starts the recording of a new macro or stops the recording when done (this item automatically changes from Start Recording to End Recording after a macro recording session is begun.). The completed Macro should then be saved into the macro folder using the **Save** command from the **File** menu.

Run Macro
Starts the macro shown in the active window.

Set Directory…
Lets you set the directory where active macros are located. The names of the active macros are placed at the bottom of the **Macros** menu. Select the name of the macro you wish to run.

List All
Lists all macro commands along with their arguments and return values.

Commands > (Submenu)
Allows you to insert certain user-interaction steps into the macro you are recording.

Change Frame…
Records a command to change the active frame of an image window.

Move Window…
Records a command to move a window.

Prompt For Image…
Records a command to prompt the macro user for an image window.

Prompt For Frame…
Records a command to prompt the macro user for a frame of an image window.

Prompt For Graph…
Records a command to prompt the macro user for a graph window.

Prompt For Number…
Records a command to prompt the macro user for a number.

Prompt For Oval Region…
Records a command to prompt the macro user to draw an oval region.

Prompt For Rect. Region…
Records a command to prompt the macro user to draw a rectangular region.

Prompt For Region…
Records a command to prompt the macro user to draw a free-hand region.

Prompt For Rgn Move…
Records a command to prompt the macro user to move an existing region.

Prompt For String…
Records a command to prompt the macro user for a string.

Resize Window…
Records a command to resize a window.

Select Window…
Records a command to activate a window.

11 APPENDIX A Clinical Acquisition Protocols

The following protocols are suggested for clinical use, but you may wish to modify them per specific needs.

11.1 Image Sequences

Lung Scan - Perfusion
(each view; ant, lao, l lat*, lpo, post, rpo, r lat*, rao)

```
Width        100
Height       100
Depth        Word
Scaling Auto 1.0
Update       2
Brightness   30
Count Limit  300000, *250,000 for these views
Time Limit   na
```

Lung Scan - Ventilation (aerosol)

```
Width        100
Height       100
Depth        Word
Scaling Auto 1.0
Update       2
Brightness   30
Count Limit  400000, *300,000 for these views
Time Limit   na
```

Bone - Spots
(each view)

```
Width        150
Height       150
Depth        Word
Scaling Auto 1.0
Update       2
Brightness   30
Count Limit  500000
Time Limit   5 min
```

Bone - Whole Body
(each view)

```
Width        200
Height       800
Depth        Word
```

Scaling Auto 1.0.
Update 5
Brightness 30
Count Limit na
Time Limit 30 min

Gastric Emptying
(each view)

Width 100
Height 100
Depth Word
Scaling Auto 1.0
Update 2
Brightness 30
Count Limit na
Time Limit 60 sec

Thyroid
(each view)

Width 100
Height 100
Depth Word
Scaling Auto 1.0
Update 2
Brightness 30
Count Limit 100000
Time Limit 5 min

COR on GE 400T
(each view of 4; anterior, left lateral, posterior, right lateral)

Width 80
Height 80
Depth Word
Scaling 11 (about 35% zoom)
Update 2
Brightness 30
Count Limit 50000
Time Limit na

11.2 Dynamic Time Series

Renal DTPA with Flow

Width 100
Height 100
Depth Word

Scaling Auto 1.0
Brightness 30
Frames 1 30
Duration 1 2 secs
Frames 2 30
Duration 2 60 secs

Renal DTPA for GFR

Width 100
Height 100
Depth Word
Scaling Auto 1.0
Brightness 30
Frames 1 1 (syringe)
Duration 1 60 secs
pause
Frames 2 30 (patient) (begin when bolus first reaches kidney)
Duration 2 60 secs (use the 3nd frame of this part of the acquisition for renal ROIs)

Renal DTPA with Lasix

Width 100
Height 100
Depth Word
Scaling Auto 1.0
Brightness 30
Frames 1 25
Duration 1 60 secs
pause
Frames 2 25
Duration 2 60 secs

Renal MAG3 with Flow

Width 100
Height 100
Depth Word
Scaling Auto 1.0
Brightness 30
Frames 1 30
Duration 1 2 secs
Frames 2 30
Duration 2 60 secs

Renal Hippuran

Width 100

Height 100
Depth Word
Scaling Auto 1.0
Brightness 30
Frames 1 30
Duration 1 60 secs

Renal Hippuran with ERPF

Width 100
Height 100
Depth Word
Scaling Auto 1.0
Brightness 30
Frames 1 1 (syringe)
Duration 1 60 secs
pause
Frames 2 30 (patient) (begin immediately after injection)
Duration 2 60 secs (use the 2nd frame of this part of the acquisition for renal ROIs)

Liver Function - HIDA

Width 100
Height 100
Depth Word
Scaling Auto 1.0
Brightness 30
Frames 1 60
Duration 1 60 secs

Cardiac First Pass for RV Ejection Fraction

Width 64
Height 64
Depth Byte
Scaling Auto 1.0 on SMOV, Auto 1.4 on LFOV
Brightness 30
Frames 1 400
Duration 1 50 millisecs

Cardiac First Pass for Shunt Quantification

Width 64
Height 64
Depth Byte
Scaling Auto 1.0 on SMOV, Auto 1.0 or 1.2 on LFOV
Brightness 30
Frames 1 150
Duration 1 200 millisecs

Lung Xenon

```
Width         100
Height        100
Depth         Word
Scaling Auto  1.0
Brightness    30
Frames 1      1
Duration 1    10 secs
 pause
Frames 2      1
Duration 2    10 secs
 pause
Frames 3      12
Duration 3    20 secs
```

GI Bleed

```
Width         100
Height        100
Depth         Word
Scaling Auto  1.0
Brightness    30
Frames 1      90
Duration 1    60 secs
```

11.3 Gated Studies

Standard List Mode - Fixed Interval Forward 2/3, Backward 1/3

```
Width            100
Height           100
Depth            Word
Scaling Auto     1.0 with SMOV, Auto 1.4 with LFOV
Brightness       30
Frames           16
Averaging Beats  20
Display Update   30
Brightness       50
Counts           na
Time Limit       10 min
Settings
 Minimum -20 (% of median)
 Maximum 20 (% of median)
 Fixed Bins (on)
 67% at start
```

Exercise - Each Level

```
Width            100
```

```
Height          100
Depth           Word
Scaling Auto    1.0 with SMOV, Auto 1.4 with LFOV
Brightness      30
Frames          16
Averaging Beats 20
Display Update  30
Brightness      50
Counts          na
Time Limit      150 secs
Settings
 Minimum -20 (% of median)
 Maximum 20 (% of median)
 Fixed Bins (on)
 67% at start
```

12 APPENDIX B Manual Processing Examples

This appendix gives some examples of manual, step-by-step, processing.

12.1 First Pass Examinations of the Right Ventricle

1) Open the file containing the first pass study.

2) Using the "+" tool in the Controls window, select a group of frames (approximately 50 to 100) by dragging on their numbers. Then use the **New Composite** item of the **Analyze>Image Math** submenu to produce a composite image. The composite image can be resized for convenience.

3) Draw a region around the right ventricle in the composite image. Move the region to the window containing the first pass data using the **Copy** and **Paste** commands in the **Edit** menu.

4) Do the same for a background region.

5) Use the "Graph" tool in the control window to select all of the frames of the first pass study. This can be done by dragging the tool over the frame numbers or by holding the "shift" key down while clicking on the first and last numbers.

6) Select the background region in the first pass window by clicking on it with the Arrow cursor. Select **Background** from the **Regions>First Pass** submenu.

7) Activate the RV region and select **Right Ventricle** from the **Regions>First Pass** submenu.

8) NucLear MAC generates a background-corrected time-activity curve.

9) Use the "Large circle" tool (on the right side of the graph window) to mark several peaks (diastolic periods) near the beginning of the curve, then use the "Small circle" tool to mark corresponding valleys (systolic periods).

10) The calculated ejection fraction (EF) based on the average of values from the chosen diastole-systole pairs is printed in the bottom of the window. Note: EF is defined as [(diastole - background) - (systole - background)] / (diastole - background).

11) Click on the small heart icon in the graph window to perform variable region size analysis if desired, then follow the directions on the screen.

12.2 Gated Examination of the Left Ventricle

1) Open the file containing the gated study.

2) View the data in cinegraphic mode by selecting **Run** from the **Display** menu. Contrast can be controlled while the cinegram is playing. If you prefer to view smoothed data, perform the following optional step.

2a) Smooth the data spatially using the desired filter in the **Analyze>Spatial Convolution** submenu.

2b) Smooth the resulting data temporally using the desired filter in the **Analyze>Temporal Convolution** submenu.

2c) Alternatively, instead ot 2a) and 2b), select **3D Smooth** from the **Analyze>Other Filtering** submenu and the study images are spatially and temporally filtered using 27-point unweighted smoothing.

3) Using the pointer tool in the Controls window, activate frame one by pointing to the number "1". Draw a region around the left ventricle (LV) using the irregular ROI tool from the Tools window. Select the **Diastole** item from the **Regions>Heart** submenu. Draw a background region, then select the **Background** item from the **Regions>Heart** submenu.

4) Advance to the systolic frame by clicking on its number with the pointer tool in the Controls window. You can toggle between frames to pick the best one. Draw a region around the LV. Select the **Systole** item from the **Regions>Heart** submenu.

5) The ejection fraction (EF) is printed in the Analysis window. Note: EF is defined as [(diastole - background) - (systole - background)] / (diastole - background).

6) If you wish to perform phase analysis, activate the window containing the smoothed cinegram, then choose **Phase** from the **Analyze>Gated Cardiac** submenu.

12.3 Thallium Scan Quantitative Analysis

1) Draw a background region around the LV in the stress image. You can use any of the ROI tools, but the "Oval" tool is probably the most convenient.

2) Choose the **Band...** selection from the **Regions>Utilities** submenu. A dialog appears that allows you to draw the band inward or outward from the previously drawn region. Typical choices are "4" pixels "Outward". Click "Make Band".

3) The band shaped region appears on the image. This region is sampled for the background subtraction using the least-squared fit method.

4) Select **Subtract as Background...** from the **Regions>Utilities** submenu. A dialog window appears. Choose between average, linear, or quadratic for the type of interpolation. The amount of subtraction can be reduced by changing the fraction in the "Reduce subtraction by" field. Typical choices are 0% reduction and linear interpolation. Click the "Subtract" button.

5) A background-subtracted image appears in a new window.

6) Draw a tight-fitting region around the LV in the new window.

7) Place axes in the center of the LV using the “Axes” tool from the Tools window. You can set the zero degree direction by clicking and dragging on the selection box at the end of either axis.

8) Select **Settings...** from the **Analyze>Planar Thallium** submenu. A dialog appears allowing you to choose between maximum value profiles, average profiles, or line integral profiles. Select a number of points, e.g. 180 and click “Done” or “Set Default”.

9) Select **Stress Profile** from the **Analyze>Planar Thallium** submenu.

10) The computer plots the circumferential profile.

11) Repeat the above procedure for the corresponding delayed image, this time selecting **Rest Profile** from the **Analyze>Planar Thallium** submenu in step 9).

12) The computer plots the circumferential profiles and washout curve. Note: washout is defined as (stress activity - delayed activity) / (stress activity).

13 APPENDIX C Revision History

This appendix contains a revision history for the NucLear MAC software. This list is available in NucLear MAC: Select **About NucLear MAC…** from the Apple menu, then click on the "What's New" button.

13.1 Version 4.1

1. User Functions Execute on Open Opening a user function file is now allowed as a shortcut for running the user function. Previously, any attempt to open a NucLear MAC user function file did nothing; user functions were executed only through the Analyze>User Functions… submenu.

2. Macros can be User Functions You can export a macro as a user function. The resulting run-only file executes when opened. Thus macros can be run by double-clicking on desktop icons.

13.2 Version 4.0

1. All Views and All Angles Commands The All-Views and All-Angles commands have been simplified. There is now only one command for each. The target study is inspected to determine whether its orientation is coronal or transverse. If the results are not as expected, then you should set the correct study orientation via the Edit>Information submenu.

2. Live Thumb Controls The thumbs (indicators) of standard scroll controls are now live. This means changes occur instantly while you move the indicators. Move the cursor away from the control to revert to the initial state.

3. Control of All-Views and All-Angles Windows The All-Views and All-Angles windows are now positioned underneath the floating palette windows. You can use the Controls window to adjust cine speed and current frame of these windows.

4. Gated-SPECT Surface Window When computing gated-SPECT ejection fraction automatically, NucLear MAC creates a window containing a surface rendering of the data. Use the Controls window and standard Display commands to adjust the window. You can customize the display with the Analyze>Gated SPECT>Surface Plot Settings… menu command.

13.3 Version 3.2

1. Gated-SPECT Filtering Revision You can now apply linear filtering (one dimensional kernels) to gated-SPECT data in two different ways: 1) To the gated component or 2) To the temporal or z-dimensional component. Scientific Imaging recommends that no filtering of the gated component be performed if the resulting data will be used for calculation of ejection fraction. The special three-d smooth filter has been modified so that it does not smooth the gated component of gated-SPECT studies.

2. Automatic EF Calculation for Gated SPECT There is now a semi-automatic method for computing the ejection fraction from reconstructed gated-SPECT slices. Use the **Analyze>Gated SPECT>Compute EF/Auto...** menu command. You are prompted to draw an oval region that encloses the heart in all frames. The pixel sizes of the source data must be correct, and the x, y, and z pixel sizes must be nearly equal. Be sure to inspect the chosen mid-myocardial surfaces, as no automated EF calculation is infallible. The calculation used is based on the one described by G. Germano, et al. in "Automatic Quantification of Ejection Fraction from Gated Myocardial Perfusion SPECT," J Nucl Med 1995; 36:2138-2147.

3. Curve Math Functions Certain functions of one curve that evaluate to constants can be entered in the curve math equation. For example, the integral of a curve labeled c1 is integral(c1). Other functions include: n_points, max_x, min_x, max_y, min_y, x_sum, y_sum, x2_sum, y2_sum, x_y_sum, y_at_max_x, y_at_min_x, x_at_max_y, x_at_min_y, left_x, right_x, y_at_left_x, y_at_right_x.

4. Curve Fitting Overhaul You can now open a separate curve fitting window for each graph window. You can enter any expression in the range fields, including the constant curve functions listed above (without an argument or parentheses). You can enter constant subexpressions in the curve fitting equation; simply put the subexpression between single quotes. Initial value fields also accept any constant expression. The macro language now supports curve fitting procedures.

13.4 Version 3.1

1. Protocols Window When you open a NucLear MAC Protocols file to incorporate the protocols it contains into your preferences file, a window is displayed that lists all the protocols that were read in along with their types.

2. Photo Mode A **Photo Mode** command has been added to the **Display** menu. This command hides the menu bar as well as all icons and windows that are behind NucLear MAC. Click the mouse or press a key to exit the photo mode.

3. Color Component Exchanger You can now exchange red, green and blue color components when designing color tables with the Custom Colors dialog.

4. Auto-Scale COR Enhancement The auto-scale COR parameter now applies to all modes of data acquisition in which auto-scaling is enabled. Previously only the SPECT and gated-SPECT acquisition modes were affected. This change simplifies the acquisition of floods for uniformity correction.

5. Bull's-eye Rest SPECT You can now specify a SPECT reconstruction method for the rest portion of the bull's-eye plot script that differs from the stress script. Use this ability with caution, as any significant differences in SPECT processing can make comparisons of stress and rest images meaningless.

6. Printing Windows In All-Frames Format The **Print** command now prints the entire contents of a study window in All-Frames Format rather than just the visible portion.

7. DICOM 3.0 Rudimentary support for the DICOM 3.0 file format has been added. You can import and export images in this format. In this release it may be necessary to manually regroup and/or separate image frames before processing the following types of imported studies: multiple-energy, multiple-detector, gated-SPECT, gated-reconstruction, multiple R-R-interval.

8. New File Group The **New File Group...** command in the **File** menu saves a file containing the names and screen locations of a group of images and graphs. The images and graphs must have been previously saved in the location where you save the group file. When you open the group file later, the images and graphs are automatically opened and moved to their original screen locations. The **New Composite** command has been moved to the **Image Math** menu;

9. Surface Rendering The **Set View Angle** command in the **Display>Views** submenu has been split into two functions. The **Set View Angle/VR** command performs volume rendering as previously. The **Set View Angle/SR** command performs surface rendering. Hold down the key to adjust the surface threshold.

13.5 Version 3.0

1. Simultaneous Gated Acquisitions Gated and Gated-SPECT studies can now be acquired in the background, which allows you do other processing concurrently. For best performance, adjust the List Mode Buffer size in the **Acquire>Settings...** dialog to reflect the maximum number of counts expected in any view. Reduce the List Mode Buffer size if you run out of memory.

2. Progress Cursor For many types of processing, there is now a progress cursor. The cursor is an hourglass whose bottom fills while the top empties.

3. More Import Types Several special import types have been added to accommodate importing filters, color tables and patient records directly through the **File>Open...** command.

4. Expanded Exporting You can now use the **File>Export...** command to export filters and color tables. Previously you could only export these items via a button in the associated dialog.

5. New Macros The following macros have been added: PromptExportAs, ExportAs, GetFileInfo, GetDateTime, and GetStudyDateTime.

13.6 Version 2.9

1. Enhanced Cine Capability It is now possible to run All-Angles and All-Views windows as movies. This is especially useful for viewing reconstructed gated-SPECT data.

2. Views Functions Most of the functions under the **Views** submenu have been modified to properly handle gated data. This means that you can easily compute other

view angles for gated reconstruction data. In addition, the **Set View Angle** command creates a cinegram of gated data.

3. Gated-SPECT Reconstruction You can now reconstruct gated-SPECT projections directly, without having to use the gated-SPECT script. Use the **Analyze>SPECT...** menu command

4. Macros The following macro functions have been added: PromptForPageSetup, SetWindowPageSetup, and WindowHasPageSetup.

5. Gated-SPECT Script The gated-SPECT script has been simplified. Its primary function now is to generate the gated short-axis slices. Generating the volume-rendered images is optional, as you can now set the number of views to zero. The All-Angles and All-Views windows should be used to examine the resulting short-axis data.

13.7 Version 2.8

1. Curve Convolution/Deconvolution The code to convolve and deconvolve curves has been updated and now includes support for macros.

2. Macros The macro operators '+', '==' and '!=' can now be used with strings. These can be used in place of the functions StringConcat and StringEquals. A StringCompare macro has been added. Other new macro functions include ClearCurveSymbols, PutGraphMessage, and IntegrateCurve.

3. Gated-SPECT Surface Plotting A new option under the **Gated SPECT** submenu allows you to make a profile plot on a surface. The shape of the surface is determined by the elliptical regions used for the profiles.

4. Quality Assurance Definitions The definitions of INL and DNL have been modified to make them less sensitive to random noise. The computations can now ignore a user-specified percentage of the respective histogram areas.

13.8 Version 2.7

1. Script Region Prompts In a script, when you are asked to draw a free-hand region, you can now edit the region without completely redrawing it. Use the union tool to enlarge the region and the difference tool to reduce the region. The union tool is accessed by holding down the shift key when clicking. The difference tool is accessed by holding down the shift and control keys when clicking.

2. Bull's-eye Plotting Change After you have identified them, the locations of the apex and base are now saved with the short axis slices. This makes it easier to recompute a bull's-eye plot at a later time.

3. Surface Bull's-eye Plotting A new option under the **Bull's-eye Plotting** submenu allows you to make a profile plot on a surface. The shape of the surface is determined by the elliptical regions used for the profiles. The size and number of views of the surface can be changed in the Bull's-eye Plot Settings dialog.

13.9 Version 2.6

1. Graphing Changes Many graphing functions have been simplified and enhanced. Among the changes and additions: the **Region To Variable** command uses the settings in the non-modal New Variable dialog; a curve can be generated from multiple regions in an image set; any manually-created graph can be selected as the destination for new curves; there are now macro commands that perform the various graphing functions; manually-created graphs are now made only with complete curves, thus the **Region To Point** command has been removed—the New Variable dialog can be used to enter individual points.

2. ROI Drawing Addition When drawing a free-hand region-of-interest, you can include straight line segments by doing the following: hold down the option key, move the cursor to the desired location, and release the option key. The mouse button must still be held down until the ROI is complete..

3. Bull's-eye Plot Overlay You can now define a set of regions to be drawn as an overlay on any frame of a completed bull's-eye plot. When a bull's-eye database is used, the regions are analyzed for extent and severity of defects. The results are displayed in the Analysis window, and they are also stored with the bull's-eye plot, where they can be accessed with the **Edit>Study Information>Specific** command.

4. Projection Masks The bull's-eye-plot and gated-SPECT scripts can be instructed to automatically place upper and lower masks on the projection data to aid with motion correction. These masks can be resized as needed. There is also a separate menu command to place these masks on other SPECT data.

5. Gated-SPECT Script Additions The volume change between diastole and systole is now computed when this option is enabled in the gated-SPECT settings dialog. Using the Script Windows dialog, you can now define the sizes and locations of the windows generated by the gated-SPECT script.

13.10 Version 2.5

1. Region Histograms Region histograms are now drawn in standard NucLear MAC graphs.

2. Complement and Invert Change The Complement and Invert image math commands now create new study windows, and no longer alter the original data.

3. All-Frames and Movie Windows Image sets are now displayed in single windows. These windows can be toggled between movie and all-frames formats.

4. Script Windows The code to automatically reposition, resize, and rename script windows has been upgraded to allow greater flexibility. Because the internal format has changed, existing records must be re-entered.

5. Information-Window Text You can now select, copy and export the text in informative windows.

6. Renaming Preferences There are now preference windows to allow arbitrary renaming of the following items: 1) All-views window captions. 2) Studies acquired from the Any-Study dialog. 3) Imported patient records.

7. Region Analysis Analyzing a region now displays the top, left, bottom and right boundaries of the region in addition to what was shown previously.

13.11 Version 2.4

1. New Color Table Tools Three features have been added to the Custom Colors dialog: 1) You can now specify a range of x values for the color table equation. This makes it easier to build a color table in piece-wise fashion. 2) Using the same range of x values, you can specify initial and final colors, and the program interpolates colors for all the intermediate x values. The "Pick" command uses the system color-picking dialog. 3) Any color table can be scaled so that its average intensity matches the specified function f(x) over the indicated range of x values. Thus you can build a color table from any set of colors and then force the response to have linear intensity (f(x) = x). Another useful scaling function is f(x) = x*x/255. This causes extra suppression of background.

2. Palette Manager Conformance NucLear MAC is now Palette Manager-friendly. This means that there should be less disruption of the system color table when other graphics applications are operating concurrently. A side effect of this is that a system version of at least 6.0.5 is required to run NucLear MAC.

13.12 Version 2.3a5.0

1. Auto-Scale COR In the past, it was unwise to acquire a SPECT study with auto-scaling enabled. The user had to keep track of the center-of-rotation offset, and then enter it when the study was reconstructed. This is no longer the case, as the NucLear MAC now retains an absolute center of rotation for use in auto-scaled SPECT studies. This parameter can be entered directly in the COR Settings dialog, but most often it is computed by running the Auto-Scale script under the **Analyze>COR** submenu. The script requires a previously acquired calibration study, which can be either of the following: 1) A 360° SPECT study with an even number of views. 2) One or more pairs of images. Each pair is acquired such that the camera is rotated 180° between the two images. The pairs need not be acquired in order, so, for example, you could acquire an image sequence with the camera at these successive angles: 0°, 90°, 180°, 270°. A vertically-oriented line source is recommended for either type of acquisition, but a point source near the center of the field of view is adequate.

2. Bull's-eye Databases Due to their large sizes, bull's-eye databases are now stored as separate files. Databases in the preferences file are automatically converted and stored in a folder named "NucLearMAC Bull's-eye Databases". You can change the directory for bull's-eye plot databases by choosing the new "Set Directory…" command under the "Database name" popup menu. The **Save Protocols As…** menu command no longer applies to bull's-eye databases. Instead, simply use the Macintosh Finder to make copies of the bull's-eye database files. The new bull's-eye database files may have generic icons until you rebuild your desktop. With newer

system software, this is accomplished by holding down the command and option keys during system start-up.

3. Improved All-Views Windows All-views windows are no longer limited to displaying two image sets. They can now show one, two, three, or four sets of images. As before, the front window is taken as the first image set. Subsequent mouse clicks are used to add additional image sets to the all-views window. Selecting an image twice ends the image entry.

4. Centroid Functions Two new ROI functions have been added. The **X Centroid** and **Y Centroid** commands are found under the **Regions>Column Values** and **Regions>Row Values** submenus respectively. These functions compute the X and Y centroids of rectangular ROIs. If the window containing the ROI has one frame, then the centroid is displayed in the Analysis window. For studies with multiple frames, a plot of the centroids is created, with the frame numbers along the X axis.

13.13 Version 2.3a4.0

1. Pixel Saturation Warning A warning is now presented after an image acquisition in which at least one pixel has saturated. Saturation occurs when more than 255 counts accumulate in any byte-mode pixel, or when more than 65535 counts accumulate in a word-mode pixel. Because 255 and 65535 are the respective maximum pixel values, additional counts are discarded, and the image becomes inaccurate. This warning can be disabled in the Acquisition Settings dialog. In addition, when saturation is detected during acquisition, elapsed counts in the progress window are replaced by “Saturated”.

2. Gated Acquisition Information A modal dialog is no longer used to display the final information about gated acquisitions. Instead, this information is written to the Analysis window.

3. Flip Horizontal/Vertical Change The Flip Horizontal and Flip Vertical views commands now create new study windows, and no longer alter the original data.

4. Improved Text Printing Printing of text windows now includes page numbers and appropriate page breaks. This affects printing of macros, information windows, and the Analysis window.

13.14 Version 2.3a3.0

1. Preference Changes Several settings dialogs have been modified. You will need to re-enter your preferences in these dialogs if you were not using the default values. These include Bull’s-eye Plotting and Gated Cardiac script preferences, as well as SPECT serial control settings.

2. About Box The NucLear MAC About dialog is no longer modal. In addition, when it is front-most, the dialog displays the amount of free memory.

3. What’s New Window The What’s New window has been added to list recent additions to the NucLear MAC program. This window is accessed through a button in the About NucLear MAC dialog.

4. Measuring When the length of an arrow is measured, the length conversion factors are now taken from the image, if they are available, and otherwise from the Measurements preferences dialog.

5. Analyzing Arrows The **Regions>Analyze** menu command now applies to arrows as well as to regions of interest. Previously, you could only obtain the length of an arrow by double-clicking on it.

6. Patient Records It is now possible to import and export patient records in text formats that you define. Additionally, a Delete All button has been added to the Patient Records dialog.

7. Hardware Calibration Software has been added to automatically calibrate the analog-to-digital converter prior to data acquisition. As a result, the hardware settings dialog has been simplified. The fields containing the table ratios—typically around 512/168—have been removed. It is recommended that the X and Y table increments each be set to a value of two, which is the default value. A Ratios button has been added so you can display the calculated X and Y table ratios. You must perform some data acquisition, even if only in the P-Scope, in order to see the ratios. As with any data acquisition, you should allow the Macintosh to warm up for at least ten minutes after a cold power-up. To compare the automatic ratios with previous values, multiply them by the table increment. For example, a ratio of 1.524 times a table increment of two is 3.048, which compares favorably to the nominal 512/168 (3.0476).

8. Spot-Test Acquisition A new mode of data acquisition has been added. In addition to 'Random test' and 'Standard', there is now a 'Spot test'. The spot test verifies that the hardware calibration described above is functioning properly. A typical test protocol is: Single image, 64x64 pixels, word-mode, scale factors set to one with centering, and acquisition limited to 65000 counts. Process the resulting spot image as follows: enclose the spot in a rectangular region (~12 x 12 pixels), then generate row- and column- sum plots of the region. The full width at half-maximum (FWHM) for the graph peaks should be less than two pixels. You can use curve fitting to obtain the exact FWHM, although it is sufficient to observe that the graph peaks consist of three or fewer pixels.

9. Macros List A list of all available macros can be displayed by choosing **List All** from the **Macros>Create** submenu. This list also contains the types of arguments the macros expect and the types of values they return

10. Hexadecimal In Macros Macros can now accept numbers in hexadecimal format. These numbers must be preceded by a '$' character. For example, $a20 is equivalent to 2592.

11. Import Study Type Previously, most imported images were given a default study type of Static. A field has been added to the import file format dialog to allow any NucLear MAC study type to be assigned to an imported image set. This does not apply to Interfile, which should have the study type in the file.

12. Replace Protocol Shortcut It had been somewhat tedious to modify a protocol, in that you had to select the protocol, make the changes, and then re-add the protocol by typing in the full name. Now, after you have modified a protocol, you can quickly

replace the original by holding down the option key while choosing the protocol to replace in the relevant popup menu. The last item of the menu changes from "Add…" to "Replace choice." This reminds you that the protocol you choose will be replaced by the untitled protocol. It is recommended that you enable the "Prompt before replacing a protocol" option in the Save Prompts dialog. This can prevent you from inadvertently replacing the wrong protocol.

13. Locked Disks It is now possible for NucLear MAC to access user functions, macros and protocols files that reside on locked disks.

14. All Frames Command The '/' character has been added as the command key for the **All Frames Window** / **Movie Window** menu command.

13.15 Version 2.3a2.0

1. Page Setups Individual documents, as well as classes of documents, can now maintain page setup information. In the past, NucLear MAC had one set of page setup information that was applied when any document was printed. Now you can assign page setup information to any document (image, graph, etc.). Simply bring the document to the front and choose **Page Setup…** from the **File** menu. If you save the document, then the page setup information is also saved. Similarly, in order to set a default page setup for a class of documents, such as all graphs, do the following: bring one of these documents to the front, then choose **Page Setup…** from the **File** menu while holding down the option key.

2. Printing Printing from the Finder is now properly supported. That is, you can now select NucLear MAC documents in the Finder and choose the **Print…** command. NucLear MAC is opened and a single print dialog is presented. The information in the print dialog is applied to all documents that were selected for printing.

3. Importing It is now possible to select an import file type from within the standard file open dialog. Previously, the dialog contained a check box titled "Show all types." That title has been changed to "Import." Now, when the import check box is enabled, in addition to displaying all files, the dialog presents a popup menu containing all import formats that have been previously defined in the Import Formats dialog. When a format is selected, it is used to open any type of file that NucLear MAC does not recognize. The default import format type is taken from the default in the Import Formats dialog.

4. User Functions and Macros A macro can now access user functions and other macros with full path names. That is, it is no longer necessary for the locations of the user functions or macros to be specified relative to the User Functions folder or Macros folder respectively.

5. Macro Array Indexing It is now legal in a macro to index a scalar, provided the index is zero. Previously, only arrays with two or more elements could be indexed. The change should simplify macros that must process variables containing an arbitrary number of elements. It is no longer necessary to treat one element as a special case.

6. Macro Array Overwriting It is now legal for a macro to overwrite an existing array variable in most cases. Previously, after an array variable was defined, you could not arbitrarily reassign it.

7. Curve Math The curve math dialog is no longer modal. It contains a single list of all curves currently available. Curves in the same graph are grouped in sections separated by gray lines. Double-clicking on a curve brings the graph containing that curve to the front.

13.16 Version 2.2a1.2

1. Data Acquisition The data acquisition code has been modified to permit much faster simultaneous acquisition. This means it is possible to acquire much higher count rates without having the acquisition switch to the non-simultaneous mode. In addition, at normal count rates, other operations performed during acquisition are not slowed as much as before.

13.17 Version 2.1a1.0

1. Controls Window In Scripts During the various processing scripts (Gated Cardiac, for example), full access to the Controls window is now available. This makes it easier to adjust image brightness, etc. You still must open the Controls window prior to running the script.

2. Gated Cardiac Script Ejection Fraction In the Gated Cardiac processing script, the ejection fraction is now supplied as a text overlay on the processed images.

3. Default Cine Speed It is now possible to set a default cine speed in the Image Defaults dialog. This is accessed under the **Edit>Preferences** submenu.