

---

# Image SXM

---

These pages document the SXM macro commands and functions that have been added to those provided by NIH Image to extend the flexibility of Image SXM. If you are looking for information on NIH Image macros, try the documentation supplied with NIH Image – the 'Inside NIH Image' manual and the file 'Macro Notes'.

The macro commands and functions are grouped together according to their action, and are listed in order of the appearance of their corresponding menu items in the Image SXM menus. Functions with no corresponding menu items are listed at the end. The last page has some notes on the use of data arrays by macros.

---

Dr Steve Barrett  
Department of Physics and Surface Science Research Centre  
University of Liverpool  
Liverpool  
United Kingdom

S.D.Barrett@liv.ac.uk

April 2008



---

1

[next page >](#)

---

## Contents

---

<b>Options</b>		SetSizeDistOptions	9	<b>SPM/Files</b>	
SaveROlarray	4	SizeDistribution	9	BatchConversion	16
RestoreROlarray	4	ReCalcSizeDist	9	GetFinderComment	16
		StrandLenOptions	10	SetFinderComment	16
<b>Process</b>		StrandLengths	10		
Differential	5	StraightenCurv	11	<b>SPM/Compensation</b>	
Curvature	5	NonLinearity	11	Compensation	17
UnsharpMask	5				
HoughTransform	5	<b>Stacks</b>		<b>SPM/Dimensions</b>	
		AverageWithLimits	12	ChangeDimensions	17
<b>FFT</b>		MakeMontage	12		
SelectBigSquare	6	AutoRegister	13	<b>SPM/Images</b>	
SelectLongLine	6	DepthOfFocus	13	ReCompensate	18
MakeLEEDintoFFT	6	MakeMosaic	14	RotateToHorizontal	18
IFFTofLEED	6			MatchLUTs	18
		<b>SEM</b>		ChangeLUTs	18
<b>Analyze</b>		DispCalibBar	14	SetTipRadius	19
GradientValues	7			TipLocusEffect	19
GradientHistogram	7	<b>SPM</b>		TrimImage	19
PlotRadialDistrib	8	SetPlot3dOptions	15	AddBorder	19
PlotAzimDistrib	8	SetPlot3dGeometry	15	DisplayAxes	20
ImageCorrelation	8	Plot3d	15		

<b>SPM/Calculation</b>		<b>SPM/Spectroscopy</b>		<b>No Menu Equivalent</b>	
CalcStepHeights	2 1	MakeSTSmmap	2 5	GetStrFromWindow	2 8
SurfaceArea	2 1	MakeSTSstack	2 5	ResetGetStr	2 8
SurfaceRoughness	2 1	SetSTSoptions	2 5	ReDrawPlot	2 8
				GetZscale	2 9
<b>SPM/Calibration</b>		<b>SPM/Parameters</b>		GetXYscale	2 9
ReCalibrateXY	2 2	ParamDump	2 6	GetPlotScales	3 0
ApplyXYcalib	2 2	EnterComments	2 6	GetUserData	3 1
ReCalibrateZ	2 3	CreateCaption	2 6	PlotUserData	3 1
ApplyZcalib	2 3	AddCaptionToImage	2 6	GetXYfromPix	3 2
SetZscale	2 3	SaveCaption	2 6	GetPixFromXY	3 2
CalibDistortion	2 4			GetFolderInfo	3 3
ApplyCorrection	2 4	<b>Windows</b>		OpenFirstFile	3 3
GetDistCalib	2 4	ResizeWindow	2 7	OpenNextFile	3 3
SaveDefDistortion	2 4			OpenPrevFile	3 3
SaveDistCalib	2 4				
				CalcLineShift	3 5
				RequiresSXMversion	3 5
				Use of Data Arrays	3 6

## Save ROI

SaveROIarray (index)

parameter	type	valid	values	comment
index	integer	1...8		if a ROI already exists at specified index it will be overwritten

## Restore ROI

RestoreROIarray (index)

parameter	type	valid	values	comment
index	integer	1...8		

**Differential**

Differential

**Curvature**

Curvature

**Unsharp Mask**

UnsharpMask (kernelSize)

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
kernelSize	string	'7x7', '11x11', '15x15'		

**Hough Transform**

HoughTransform (shapeSize)

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
shapeSize	integer	2...100		the size of the shape to be used

[< previous page](#)

5

[next page >](#)**Select Largest Square**

SelectBigSquare

**Select Longest Line**

SelectLongLine

**Make LEED into FFT**

MakeLEEDintoFFT

**IFFT of LEED**

IFFTofLEED

**Gradient Values**

GradientValues (azRefDir, invertLUT, pixelAvg)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
azRefDir	string	'North', 'East', 'South', 'West'	azimuthal angle reference direction
invertLUT	boolean		true = white on black background false = black on white background
pixelAvg	integer	1...20	number of pixels over which to average

**Gradient Histogram**

GradientHistogram (angleRange, windowSize, addAxes, invertLUT, pixelAvg)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
angleRange	integer	1...90	angle range covered by histogram
windowSize	integer	1...4096	size of histogram window in pixels
addAxes	boolean		true = add x and y axes to histogram false = do not add axes to histogram
invertLUT	boolean		true = white on black background false = black on white background
pixelAvg	integer	1...20	number of pixels over which to average

**Radial Distribution**

PlotRadialDistrib

**Azimuthal Distribution**

PlotAzimDistrib

**Image Correlation**

ImageCorrelation

**Size Distribution and ReCalc Size Dist**

SetSizeDistOptions (options)

SizeDistribution (min, max, inc, lineWidth, threshold)

ReCalcSizeDist (threshold)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
options	string	'line', 'circle', 'ellipse' 'bright', 'dark' 'maskFound' 'showPlot' 'makeStack' 'showFound' 'listResults' 'listXY'	shapes to be found shapes are bright or dark mask out original image shapes found show plot of size distribution make stack of the Hough transforms show shapes found in new window show size distribution results window list all coordinates in results window
min	integer	2...100	minimum size of shape to be found
max	integer	2...100	maximum size of shape to be found
inc	integer	1...20	size increment
lineWidth	integer	1..9	line width in pixels of shape to be found
threshold	integer	1...255	threshold for shape discrimination

**DNA Strand Lengths**

StrandLenOptions (backThresh, anThresh, aspectRatio, showResults, resetCounter, showUnits, showTitle)

StrandLengths

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
backThresh	integer	1..99	background threshold usually 10-30%
anThresh	integer	1..99	analyse threshold usually < 50%
aspectRatio	integer	1..99	minimum value to be counted as strand
showResults	boolean		true = open Results window
resetCounter	boolean		true = reset measurements counter false = append results to existing set
showUnits	boolean		true = show units in Results window
showTitle	boolean		true = show title in Results window

**Straighten Curvilinear Object**

StraightenCurv (x1, y1, x2, y2)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
x1, y1	integers	0..8192	xy coord of start of object
x2, y2	integers	0..8192	xy coord of end of object

**Nanotube Non-Linearity**

NonLinearity (coeff, stddev)

Both parameters are set by the macro.

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
coeff	real	0.0..1.0	= 0 for randomly entangled network = 1 for linear array of objects
stddev	real	0.0..1.0	standard deviation (error) of coeff

**Average Within Limits**

AverageWithLimits (firstSlice, lastSlice, lowLimit, highLimit)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
firstSlice	integer	1..number of images in stack	slice number of first image to average
lastSlice	integer	1..number of images in stack	slice number of last image to average
lowLimit	integer	0..255	lowest pixel value to contribute to avg
highLimit	integer	0..255	highest pixel value to contribute to avg

**Make Montage**

MakeMontage (numCols, numRows, mag, first, last, incr, addNumbers, drawBorders)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
numCols	integer	1..255	number of columns in montage
numRows	integer	1..255	number of rows in montage
mag	real		magnification of images
first	integer	1..number of images in stack	slice number of first image
last	integer	1..number of images in stack	slice number of last image
incr	integer	1..number of images in stack	increment of slice numbers
addNumbers	boolean		true = label each image with number
drawBorders	boolean		true = draw borders around each image

**Auto Register**

AutoRegister

**Depth Of Focus**

DepthOfFocus (nearLimit, farLimit, weightedAvg, reduceNoise)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
nearLimit	integer	1..99	smallest detail visible (in pixels)
farLimit	integer	nearLimit..99	half the size of characteristic features
weightedAvg	boolean		true = take weighted average, not best
reduceNoise	boolean		true = reduce noise in images first

**Make Mosaic**

MakeMosaic (orientation, order, numSamples, FFTsize)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
orientation	string	'Landscape', 'Portrait'	
order	string	'AcrossDown', 'DownAcross'	
numSamples	integer	3..15	number of FFT samples per overlap
FFTsize	integer	16, 32, 64, 128	size of FFT to use in overlap calculation (any invalid value = 'Auto')

**Display Calib Bar**

DispCalibBar (corner, barSize, fontSize, barStyle, useAngstroms)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
corner	string	'TopLeft', 'TopRight' 'BottomLeft', 'BottomRight'	corner of image to display calib bar
barSize	string	'BarSizeSmall', 'BarSizeOpt' 'BarSizeLarge'	optimum size is in proportion to image small and large are relative to optimum
fontSize	string	'FontSizeFixed', 'FontSizeOpt'	optimum size is in proportion to image
barStyle	integer	0..3	see dialog box for styles
useAngstroms	boolean		true = display calib bar size in Å false = display calib bar size in nm

### 3d Plot

SetPlot3dOptions (options, LUTpc)

SetPlot3dGeom (thetaV, phiV, thetaL, phiL, viewDist, zMag)

Plot3d

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
options	string	'drawAxes', 'noAxes' 'useLUT', 'useShadows', 'useBoth' '2dPlot', '3dPlot', '4dPlot' 'faster', 'better'	
LUTpc	integer	1..99	% LUT when 'useBoth' selected
thetaV	integer	5..85	view angle (altitude)
phiV	integer	-90..90	view angle (azimuth)
thetaL	integer	5..85	illumination angle (altitude)
phiL	integer	-90..90	illumination angle (azimuth)
viewDist	integer	2..50	viewing distance
zMag	integer	2..50	magnification of z scale

Note that in versions of Image SXM prior to 165-1 the parameter 'viewDist' had an effect on the apparent perspective of 3d plots. This is now redundant as 3d plots are created without perspective, but the parameter is kept for backwards compatibility.

### Batch Conversion

BatchConversion (loadFormats, saveFormat, saveWhere, folderName)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
loadFormats	string	'loadPICT', 'loadTIFF', 'loadSXM'	user can choose more than one format
saveFormat	string	'saveAsPICT', 'saveAsTIFF'	user must choose one format only
saveWhere	string	'sameFolder', 'newFolder'	if same folder, names will have suffixes
folderName	string		must be unique name

### Get Finder Comment

str := GetFinderComment

### Set Finder Comment

SetFinderComment



**Compensation**

Compensation (compToApply)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
compToApply	string	'Tilt', 'Quad', 'LbyL' 'Const', 'Linear', 'Quad'	XY Tilt, Quadratic Sub or Line-By-Line order of fit if 'LbyL' is selected

**Change Dimensions**

ChangeDimensions (newUnits)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
newUnits	string	'pm', 'Å', 'nm', 'µm', 'cm' 'mm', 'meter', 'km'	

**ReCompensate**

ReCompensate

**Rotate To Horizontal**

RotateToHorizontal

**Match LUTs**

MatchLUTs

**Change LUTs**

ChangeLUTs (lowZ, highZ)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
lowZ	real		new z value that will be set to black
highZ	real		new z value that will be set to white

**Tip Locus Effect**

SetTipRadius (tipRadius)  
TipLocusEffect

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
tipRadius	real		tip radius in nm

**Trim Image**

TrimImage

**Add Border**

AddBorder (options, borderSize)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
options	string	'forRotation' '50%', '100%' 'specify'	border wide enough for image rotation
borderSize	integer	1..500	border size if 'specify' is selected

**Display Axes and Display Axes and LUT**

DisplayAxes (addLUT)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
addLUT	boolean		true = add LUT scale to side of image false = no LUT scale

**Step Height**

CalcStepHeight (thisProfile)

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
thisProfile	boolean			true = calculate for current profile plot false = take new profile, then calculate

**Surface Area**

x := SurfaceArea

**Surface Roughness**

x := SurfaceRoughness (zParam)

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
zParam	string	'zRMS'		returns rms value of Z roughness
		'zRa'		returns absolute value of Z roughness

**Recalibrate XY**

RecalibrateXY (x1, y1, x2, y2, x3, y3, actualA, actualB, actualTheta, unitCellsA, unitCellsB)

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
x1, ... , y3	integer			pixel coords defining two lattice vectors
actualA	real			actual length of lattice vector A
actualB	real			actual length of lattice vector B
theta	integer			angle between vectors A and B
unitCellsA	integer			number of unit cells
unitCellsB	integer			number of unit cells

**Apply XY Calib**

ApplyXYcalib

**Recalibrate Z**  
RecalibrateZ

---

**Apply Z Calib**  
ApplyZcalib

---

**Set Z Scale**

SetZscale (lowZ, highZ, units)

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
lowZ	real			value that will be made lowest z in image
highZ	real			value that will be made highest z in image
units	string			units of z scale

---

**Calibrate Distortion**  
CalibDistortion

---

**Apply Correction**  
ApplyCorrection

---

**Get Distortion Calib**  
GetDistCalib

---

**Save Default Distortion**  
SaveDefDist

---

**Save Distortion Calib**  
SaveDistCalib

---

**Make STS Map**

MakeSTSmapi (voltage)

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
voltage	real			value at which current values will be extracted from pixel STS spectra and converted into an image

---

**Make STS Stack**

MakeSTSstack

**STS Options**

SetSTSoptions (Imin, Imax)

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
Imin	real			min I value for STS image scale
Imax	real			max I value for STS image scale

---

**Parameter Dump**

ParamDump

**Enter Comments**

EnterComments

**Create Caption**

CreateCaption

**Add Caption To Image**

AddCaptionToImage

**Save Caption**

SaveCaption

**Resize Window**

ResizeWindow (percent)

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
percent	integer	20,25,33,50,100,200,400	any other value will be treated as 100%

The following commands and functions have no corresponding menu items

---

**GetStrFromWindow**

returns a string read from the selected text window. By calling GetStrFromWindow from within a loop, the user can access all of the text in a window. If the current window is not a text window then calling this function returns '<< Not a text window >>'. If all of the text in the window has been read then the next call returns '<< End of text >>'. A subsequent call will read from the beginning of the text window.

---

**ResetGetStr**

resets the pointer that keeps track of strings read from text windows. A subsequent call of the function 'GetStrFromWindow' will read from the beginning of the text window.

---

**ReDrawPlot**

enables a profile plot to be redrawn after the plot data has been modified by the user.

---

**GetZscale (greysPerUnit, unit)**

returns the z calibration of the current image, and has been added to complement the NIH Image macro 'GetScale' that returns the xy calibration.

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
greysPerUnit	real			value set by macro command
unit	string			string set by macro command

**GetXYscale (pixelsPerUnit, unit)**

is identical to the NIH Image macro command 'GetScale', and has been added to eliminate ambiguity in macros that also use the SXM macro command 'GetZscale'.

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
pixelsPerUnit	real			value set by macro command
unit	string			string set by macro command

**GetPlotScales (xScale, yScale, xOffset, yOffset, xUnit, yUnit)**

gets the scales and offsets of the plot (profile, FFT of profile, force curve, line scan, I-V spectrum, spot profile, etc) in the current window. The six parameters are all set by the macro.

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
xScale, yScale	real			scale factors = units per pixel
xOffset, yOffset	integer			pixel offsets of zeros of x and y axes
xUnit, yUnit	string			units of x and y axes of plot

## GetUserData (numPoints, xUnit, yUnit)

transfers data values from the plot (force curve, line scan, I-V spectrum, spot profile, etc) in the current window into the user arrays: rUser3 = x values, rUser4 = y values, rUser5 = second set of y values if the plot has two data sets. The three parameters are all set by the macro.

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
numPoints	integer	2..1024		number of points read into rUser arrays
xUnit, yUnit	string			units of x and y axes of plot

## PlotUserData (plotTitle, rUserNum, numPoints, xScale, xUnit, yUnit)

uses the Profile Plot routines to create a plot of user-defined data. Arrays rUser1 and rUser2 are for use exclusively by user macros. Arrays rUser3, rUser4 and rUser5 are used by some routines in Image SXM (see above and 'Use of Data Arrays' on the last page).

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
plotTitle	string			title of plot window
rUserNum	integer	1..5		number of the rUser array
numPoints	integer	2..8192		number of points to take from rUser
xScale	real			scaling factor = xUnits per data point
xUnit, yUnit	string			units for x and y axes of plot

## GetXYfromPix (h, v, x, y)

converts the coordinates of a pixel in the current plot window into the corresponding real values.

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
h, v	integer			coords of pixel in plot window
x, y	real			x and y values corresponding to coord

## GetPixFromXY (x, y, h, v)

converts x and y values into the coordinates of the corresponding pixel in the current plot window.

<u>parameter</u>	<u>type</u>	<u>valid</u>	<u>values</u>	<u>comment</u>
x, y	real			x and y values of plot
h, v	integer			coords of pixel corresponding to x and y

## Notes on plots

GetPlotScales, GetUserData and GetXYfromPix offer alternative ways to access the data values of the plot in the current window, and can be used independently of each other according to the needs of the user.



Folder := GetFolderInfo (fileTypes, numFiles);

returns the full path name of the folder of the current file (see notes below) and the number of files of the specified type(s) in the folder.

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
fileTypes	string	see notes on next page	user can choose more than one file type
numFiles	integer		number of files of specified type(s)

OK := OpenFirstFile (fileTypes)

opens the first file of the specified type in the same folder as the current file.

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
fileTypes	string	see notes on next page	user can choose more than one file type

OK := OpenNextFile (fileTypes)      or      OK := OpenPrevFile (fileTypes)

opens the next/previous file of the specified type in the same folder as the current file.

<u>parameter</u>	<u>type</u>	<u>valid values</u>	<u>comment</u>
fileTypes	string	see notes on next page	user can choose more than one file type

Notes on 'GetFolderInfo' and 'Open...File' macro functions

1. The 'current file' is taken to be (i) the file of the current image or text window or, if no window is selected, (ii) the file from which an image or text was loaded most recently. If no files have yet been loaded, a dialog box is displayed: For 'GetFolderInfo' or 'OpenFirstFile', the user selects a folder; for 'OpenNextFile' or 'OpenPrevFile' the user selects a file.
2. If the option key is pressed when calling 'GetFolderInfo', then a new folder can be selected independently of the current file. This selection remains in effect until completion of the macro, after which note 1 applies again.
3. Any number of valid file types can be specified, such as: 'PICT', 'TIFF', 'iSXM', 'TEXT', 'PICS', 'MooV'. A file type of 'All' will include any image or text file.
4. The 'Open...File' functions return 'false' if no file was opened.
5. OpenNextFile('All') is equivalent to the keyboard shortcut cmd-right arrow;  
OpenPrevFile('All') is equivalent to the keyboard shortcut cmd-left arrow.
6. The files are handled in alphanumerical order – text in the filenames is handled alphabetically and numbers embedded in filenames are handled numerically.

x := CalcLineShift (line1, line2)

calculates the shift in pixels between two lines of an image using a correlation function applied to the two corresponding line profiles. The result is +ve (-ve) if line 2 is shifted right (left) relative to line 1.

parameter	type	valid values	comment
line1	integer	0..image height - 1	y coord of reference image line
line2	integer	0..image height - 1	y coord of shifted image line

RequiresSXMversion (versNum)

parameter	type	valid values	comment
versNum	real	1.62 or above	earliest version of Image SXM that can run this macro. If the version number is given to 3 decimal places, then the Image SXM release number is checked.

In NIH Image there are two general-purpose data arrays, each holding 8000 real numbers. These arrays are named User1 and User2 in the source code and are accessed from macros as rUser1 and rUser2.

In Image SXM these are complemented by an additional four data arrays. Arrays User1 and User2 are for use exclusively by user macros. Arrays User3, User4 and User5 are used by the macro commands shown in the table below. All macro commands other than those listed do not use any of the User arrays. Array User6 holds strings rather than real numbers and cannot be accessed by macros.

Macro Command	User1	User2	User3	User4	User5	User6
PlotRadialDistrib					•	
PlotAzimDistrib					•	
CalcStepHeight			•	•		
Measure					•	•
ReCalcSizeDist			•	•		
SizeDistribution			•	•		
StrandLengths			•	•	•	•
GetUserData			•	•	•	

User

Image SXM