

US Probe Calibration with LSQRREcipes
1.0

Generated by Doxygen 1.8.0

Thu May 17 2012 14:33:36

Contents

1	Namespace Index	1
1.1	Namespace List	1
2	Class Index	3
2.1	Class List	3
3	File Index	5
3.1	File List	5
4	Namespace Documentation	7
4.1	Ui Namespace Reference	7
5	Class Documentation	9
5.1	Calibration Class Reference	9
5.1.1	Detailed Description	10
5.1.2	Member Typedef Documentation	10
5.1.2.1	DataType	10
5.1.3	Member Function Documentation	10
5.1.3.1	Calibrate	10
5.1.3.2	ClearImagePoints	10
5.1.3.3	ClearTransformations	10
5.1.3.4	getEstimatedUSCalibrationParameters	10
5.1.3.5	InsertImagePoints	10
5.1.3.6	InsertTransformations	10
5.1.3.7	New	10
5.1.4	Member Data Documentation	10
5.1.4.1	data	10
5.1.4.2	estimatedUSCalibrationParameters	10
5.1.4.3	imagePoints	10
5.1.4.4	transformations	11
5.2	MainWindow Class Reference	11
5.2.1	Detailed Description	11
5.2.2	Constructor & Destructor Documentation	12

5.2.2.1	MainWindow	12
5.2.2.2	~MainWindow	12
5.2.3	Member Function Documentation	12
5.2.3.1	addImages	12
5.2.3.2	addLogText	12
5.2.3.3	displaySelectedImage	12
5.2.3.4	getDisplayWidget	12
5.2.3.5	print	12
5.2.3.6	probeCalibration	12
5.2.4	Member Data Documentation	12
5.2.4.1	Connections	12
5.2.4.2	displayWidget	12
5.2.4.3	imagesFileNames	12
5.2.4.4	textOnTextArea	13
5.2.4.5	ui	13
5.3	ProbeCalibrationWidget Class Reference	13
5.3.1	Detailed Description	14
5.3.2	Constructor & Destructor Documentation	14
5.3.2.1	ProbeCalibrationWidget	14
5.3.2.2	~ProbeCalibrationWidget	14
5.3.3	Member Function Documentation	14
5.3.3.1	calibrate	14
5.3.3.2	crop	14
5.3.3.3	cropProbeImage	14
5.3.3.4	getCoordinates	15
5.3.3.5	loadRotationsFile	15
5.3.3.6	loadTranslationsFile	15
5.3.3.7	saveCalibration	15
5.3.3.8	setImage	15
5.3.3.9	setImageStack	15
5.3.3.10	setMainWindow	15
5.3.4	Member Data Documentation	15
5.3.4.1	calibrationParameters	15
5.3.4.2	coords	15
5.3.4.3	image	15
5.3.4.4	imageStack	16
5.3.4.5	mainWindow	16
5.3.4.6	rotations	16
5.3.4.7	rotations_2	16
5.3.4.8	translations	16

5.3.4.9	workWithStack	16
5.4	QVTKImageWidget Class Reference	16
5.4.1	Detailed Description	18
5.4.2	Constructor & Destructor Documentation	18
5.4.2.1	QVTKImageWidget	18
5.4.2.2	~QVTKImageWidget	18
5.4.3	Member Function Documentation	18
5.4.3.1	displayImage	18
5.4.3.2	displaySelectedImage	18
5.4.3.3	getImageDisplayedIndex	18
5.4.3.4	getImageHeight	18
5.4.3.5	getImageSize	18
5.4.3.6	getImageStack	18
5.4.3.7	getImageType	19
5.4.3.8	getImageViewer	19
5.4.3.9	getImageWidth	19
5.4.3.10	getNumOfDimesions	19
5.4.3.11	getPixelType	19
5.4.3.12	getQVTKWidget	19
5.4.3.13	getXPicked	19
5.4.3.14	getYPicked	19
5.4.3.15	setAndDisplayImage	19
5.4.3.16	setAndDisplayImage	20
5.4.3.17	setAndDisplayMultipleImages	20
5.4.3.18	setAndDisplayMultipleImages	20
5.4.3.19	setImageProperties	20
5.4.3.20	setXPicked	20
5.4.3.21	setYPicked	20
5.4.4	Member Data Documentation	20
5.4.4.1	cornerAnnotation	20
5.4.4.2	imageDisplayedIndex	21
5.4.4.3	imageHeight	21
5.4.4.4	imageStack	21
5.4.4.5	imageType	21
5.4.4.6	imageView	21
5.4.4.7	imageWidth	21
5.4.4.8	isImageStackLoaded	21
5.4.4.9	itkImage	21
5.4.4.10	numDimensions	21
5.4.4.11	pixelType	21

5.4.4.12	qvtkWidget	21
5.4.4.13	renderer	21
5.4.4.14	renwin	22
5.4.4.15	rgbltkImage	22
5.4.4.16	vtkImage	22
5.4.4.17	xPicked	22
5.4.4.18	xPosition	22
5.4.4.19	yPicked	22
5.4.4.20	yPosition	22
5.5	QVTKImageWidgetCommand Class Reference	22
5.5.1	Detailed Description	23
5.5.2	Constructor & Destructor Documentation	23
5.5.2.1	QVTKImageWidgetCommand	23
5.5.2.2	~QVTKImageWidgetCommand	23
5.5.3	Member Function Documentation	23
5.5.3.1	Execute	23
5.5.3.2	New	23
5.5.3.3	SetAnnotation	23
5.5.3.4	SetImageWidget	23
5.5.3.5	SetPicker	23
5.5.4	Member Data Documentation	23
5.5.4.1	Annotation	23
5.5.4.2	ImageWidget	23
5.5.4.3	Picker	24
6	File Documentation	25
6.1	Calibration.cpp File Reference	25
6.2	Calibration.h File Reference	25
6.3	main.cpp File Reference	25
6.3.1	Function Documentation	25
6.3.1.1	main	25
6.4	mainwindow.cpp File Reference	26
6.5	mainwindow.h File Reference	26
6.6	ProbeCalibrationWidget.cpp File Reference	26
6.6.1	Variable Documentation	26
6.6.1.1	setCoordsSize	26
6.7	ProbeCalibrationWidget.h File Reference	27
6.8	QVTKImageWidget.cpp File Reference	27
6.9	QVTKImageWidget.h File Reference	28
6.9.1	Typedef Documentation	28

6.9.1.1	ImageType	28
6.9.1.2	RGBImageType	28
6.9.1.3	RGBPixelFormat	28
6.10	QVTKImageWidgetCommand.cpp File Reference	29
6.11	QVTKImageWidgetCommand.h File Reference	29

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Ui	7
----------	---

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Calibration	
Implements LSQRRecepies methods	9
MainWindow	
Main Window for user Interaction	11
ProbeCalibrationWidget	
Obtain data for calibration process	13
QVTKImageWidget	
Display VTK images	16
QVTKImageWidgetCommand	
Interaction with mouse	22

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Calibration.cpp	25
Calibration.h	25
main.cpp	25
mainwindow.cpp	26
mainwindow.h	26
ProbeCalibrationWidget.cpp	26
ProbeCalibrationWidget.h	27
QVTKImageWidget.cpp	27
QVTKImageWidget.h	28
QVTKImageWidgetCommand.cpp	29
QVTKImageWidgetCommand.h	29

Chapter 4

Namespace Documentation

4.1 Ui Namespace Reference

Chapter 5

Class Documentation

5.1 Calibration Class Reference

Implements LSQRRecepies methods.

```
#include <Calibration.h>
```

Public Types

- typedef
IsqrRecipes::SingleUnknownPointTargetUSCalibrationParametersEstimator::DataType [DataType](#)

Public Member Functions

- void [InsertTransformations](#) (vnl_matrix< double > rotationMatrix, vnl_vector< double > translation)
insert the rotation matrix of an image to transformations
- void [ClearTransformations](#) ()
clear transformations
- void [InsertImagePoints](#) (double p[2])
insert the crosswire point of an image to imagePoints
- void [ClearImagePoints](#) ()
clear imagePoints
- bool [Calibrate](#) ()
estimate calibration parameters
- std::vector< double > [getEstimatedUSCalibrationParameters](#) ()

Static Public Member Functions

- static [Calibration](#) * [New](#) ()
Constructor of the class.

Private Attributes

- std::vector< IsqrRecipes::Frame > [transformations](#)
- std::vector< IsqrRecipes::Point2D > [imagePoints](#)
contains the crosswire point in all images
- std::vector< [DataType](#) > [data](#)
contain the data of all images
- std::vector< double > [estimatedUSCalibrationParameters](#)

5.1.1 Detailed Description

Implements LSQRRecepies methods.

This classs have the calibration methods implemented in LSQRRecipes to calibrate an Ultra Sound Probe with a cross wire phantom.

5.1.2 Member Typedef Documentation

5.1.2.1 `typedef IsqrRecipes::SingleUnknownPointTargetUSCalibrationParametersEstimator::DataType Calibration::DataType`

5.1.3 Member Function Documentation

5.1.3.1 `bool Calibration::Calibrate ()`

estimate calibration parameters

5.1.3.2 `void Calibration::ClearImagePoints ()`

clear imagePoints

5.1.3.3 `void Calibration::ClearTransformations ()`

clear transformations

5.1.3.4 `std::vector< double > Calibration::getEstimatedUSCalibrationParameters ()`

5.1.3.5 `void Calibration::InsertImagePoints (double p[2])`

insert the crosswire point of an image to imagePoints

5.1.3.6 `void Calibration::InsertTransformations (vnl_matrix< double > rotationMatrix, vnl_vector< double > translation)`

insert the rotation matrix of an image to transformations

5.1.3.7 `static Calibration* Calibration::New () [inline, static]`

Constructor of the class.

5.1.4 Member Data Documentation

5.1.4.1 `std::vector< DataType > Calibration::data [private]`

contain the data of all images

5.1.4.2 `std::vector<double> Calibration::estimatedUSCalibrationParameters [private]`

5.1.4.3 `std::vector<IsqrRecipes::Point2D> Calibration::imagePoints [private]`

contains the crosswire point in all images

5.1.4.4 `std::vector<lsqrRecipes::Frame> Calibration::transformations` [private]

The documentation for this class was generated from the following files:

- [Calibration.h](#)
- [Calibration.cpp](#)

5.2 MainWindow Class Reference

Main Window for user Interaction.

```
#include <mainwindow.h>
```

Public Member Functions

- [MainWindow](#) (QWidget *parent=0)
- [~MainWindow](#) ()
- void [addLogText](#) (QString str)
Writes at the logger.
- [QVTKImageWidget *](#) [getDisplayWidget](#) ()
return this display widget

Private Slots

- void [addImages](#) ()
Add image folder to application.
- void [displaySelectedImage](#) (int idx)
Display selected image with the image slider.
- void [probeCalibration](#) ()
Implements a ultrasound 3D probe calibration, for navigate with the probe.
- void [print](#) ()
Print the text on logger.

Private Attributes

- Ui::MainWindow * [ui](#)
main window object
- QStringList [imagesFileNames](#)
The filename of each selected image.
- QString [textOnTextArea](#)
Text on the logger.
- [QVTKImageWidget *](#) [displayWidget](#)
- vtkSmartPointer
< vtkEventQtSlotConnect > [Connections](#)
Connects with the probe calibration widget.

5.2.1 Detailed Description

Main Window for user Interaction.

This class is the main window of ht US Probe [Calibration](#). Here the user can open the calibration images and open the calibration widget

5.2.2 Constructor & Destructor Documentation

5.2.2.1 `MainWindow::MainWindow (QWidget * parent = 0)` `[explicit]`

5.2.2.2 `MainWindow::~~MainWindow ()`

5.2.3 Member Function Documentation

5.2.3.1 `void MainWindow::addImages ()` `[private, slot]`

Add image folder to application.

5.2.3.2 `void MainWindow::addLogText (QString str)`

Writes at the logger.

5.2.3.3 `void MainWindow::displaySelectedImage (int idx)` `[private, slot]`

Display selected image with the image slider.

5.2.3.4 `QVTKImageWidget * MainWindow::getDisplayWidget ()`

return this display widget

Parameters

out	<i>this</i>	display widget
-----	-------------	----------------

5.2.3.5 `void MainWindow::print ()` `[private, slot]`

Print the text on logger.

5.2.3.6 `void MainWindow::probeCalibration ()` `[private, slot]`

Implements a ultrasound 3D probe calibration, for navigate with the probe.

5.2.4 Member Data Documentation

5.2.4.1 `vtkSmartPointer<vtkEventQtSlotConnect> MainWindow::Connections` `[private]`

Connects with the probe calibration widget.

5.2.4.2 `QVTKImageWidget* MainWindow::displayWidget` `[private]`

Central widget for display image purposes

5.2.4.3 `QStringList MainWindow::imagesFileNames` `[private]`

The filename of each selected image.

5.2.4.4 QString MainWindow::textOnTextArea [private]

Text on the logger.

5.2.4.5 Ui::MainWindow* MainWindow::ui [private]

main window object

The documentation for this class was generated from the following files:

- [mainwindow.h](#)
- [mainwindow.cpp](#)

5.3 ProbeCalibrationWidget Class Reference

Obtain data for calibration process.

```
#include <ProbeCalibrationWidget.h>
```

Public Member Functions

- [ProbeCalibrationWidget](#) (QWidget *parent=0)
- virtual [~ProbeCalibrationWidget](#) ()
- void [setImageStack](#) (std::vector< vtkSmartPointer< vtkImageData > > [imageStack](#))
Set this stack of vtkImageData.
- void [setImage](#) (vtkSmartPointer< vtkImageData > [image](#))
Set this vtkImageData.
- void [setMainWindow](#) (MainWindow *mainwindow)

Private Slots

- void [crop](#) ()
crop the images to delete the extra information
- void [getCoordinates](#) ()
get the cross point coordinates
- void [calibrate](#) ()
Calls Calibrate.h to estimate the calibration parameters.
- void [loadRotationsFile](#) ()
Load the rotations file.
- void [loadTranslationsFile](#) ()
Load the translation file.
- void [saveCalibration](#) ()
Save the Estimated Parameters in a .txt file.

Private Member Functions

- vtkSmartPointer< vtkImageData > [cropProbeImage](#) (vtkSmartPointer< vtkImageData > [image](#), int depth-Type)

Private Attributes

- bool [workWithStack](#)
- std::vector< vtkSmartPointer
 < vtkImageData > > [imageStack](#)
 an Array of vtkImageData to work
- vnl_matrix< double > [translations](#)
 a vnl_matrix to store the translations of each image given by the tracker
- vnl_matrix< double > [rotations](#)
 a vnl_matrix to store the rotations of each image given by the tracker
- float [rotations_2](#) [7][4]
- [MainWindow](#) * [mainWindow](#)
- vtkSmartPointer< vtkImageData > [image](#)
 the vtkImageData to work
- vnl_matrix< double > [coords](#)
 a vnl_matrix to store the selected coordinates on each image
- std::vector< double > [calibrationParameters](#)
 the estimate calibration parameters by [Calibration.h](#)

5.3.1 Detailed Description

Obtain data for calibration process.

This class obtain the necessary data to use the Calibration class. It let the user to set the cross wire point on the images, load Rotation and Translation Data. It also allows the user to save the calibration estimated parameters in a .txt file

5.3.2 Constructor & Destructor Documentation

5.3.2.1 `ProbeCalibrationWidget::ProbeCalibrationWidget (QWidget * parent = 0)`

Constructor

5.3.2.2 `ProbeCalibrationWidget::~~ProbeCalibrationWidget ()` [virtual]

5.3.3 Member Function Documentation

5.3.3.1 `void ProbeCalibrationWidget::calibrate ()` [private, slot]

Calls Calibration.h to estimate the calibration parameters.

5.3.3.2 `void ProbeCalibrationWidget::crop ()` [private, slot]

crop the images to delete the extra information

5.3.3.3 `vtkSmartPointer< vtkImageData > ProbeCalibrationWidget::cropProbeImage (vtkSmartPointer< vtkImageData > image, int depthType)` [private]

Crop ultrasound image depending of the depth type

5.3.3.4 void **ProbeCalibrationWidget::getCoordinates** () [private, slot]

get the cross point coordinates

5.3.3.5 void **ProbeCalibrationWidget::loadRotationsFile** () [private, slot]

Load the rotations file.

5.3.3.6 void **ProbeCalibrationWidget::loadTranslationsFile** () [private, slot]

Load the translation file.

5.3.3.7 void **ProbeCalibrationWidget::saveCalibration** () [private, slot]

Save the Estimated Parameters in a .txt file.

5.3.3.8 void **ProbeCalibrationWidget::setImage** (vtkSmartPointer< vtkImageData > *image*)

Set this vtkImageData.

Parameters

in	a	smart Pointer of vtkImageData
----	---	-------------------------------

5.3.3.9 void **ProbeCalibrationWidget::setImageStack** (std::vector< vtkSmartPointer< vtkImageData > > *imageStack*)

Set this stack of vtkImageData.

Parameters

in	a	std Vector of vtkImageData
----	---	----------------------------

5.3.3.10 void **ProbeCalibrationWidget::setMainWindow** (MainWindow * *mainwindow*)

Set the window to display the crop images

5.3.4 Member Data Documentation

5.3.4.1 std::vector<double> **ProbeCalibrationWidget::calibrationParameters** [private]

the estimate calibration parameters by [Calibration.h](#)

5.3.4.2 vnl_matrix<double> **ProbeCalibrationWidget::coords** [private]

a vnl_matrix to store the selected coordinates on each image

5.3.4.3 vtkSmartPointer<vtkImageData> **ProbeCalibrationWidget::image** [private]

the vtkImageData to work

5.3.4.4 `std::vector< vtkSmartPointer<vtkImageData> > ProbeCalibrationWidget::imageStack` [private]

an Array of vtkImageData to work

5.3.4.5 `MainWindow* ProbeCalibrationWidget::mainWindow` [private]

the main window to call it

5.3.4.6 `vnl_matrix<double> ProbeCalibrationWidget::rotations` [private]

a vnl_matrix to store the rotations of each image given by the tracker

5.3.4.7 `float ProbeCalibrationWidget::rotations_2[7][4]` [private]

5.3.4.8 `vnl_matrix<double> ProbeCalibrationWidget::translations` [private]

a vnl_matrix to store the translations of each image given by the tracker

5.3.4.9 `bool ProbeCalibrationWidget::workWithStack` [private]

The documentation for this class was generated from the following files:

- [ProbeCalibrationWidget.h](#)
- [ProbeCalibrationWidget.cpp](#)

5.4 QVTKImageWidget Class Reference

Display VTK images.

```
#include <QVTKImageWidget.h>
```

Public Member Functions

- [QVTKImageWidget](#) (QWidget *parent=0)
- virtual [~QVTKImageWidget](#) ()
- void [setAndDisplayImage](#) (QString imageFilename)
- void [setAndDisplayImage](#) (vtkSmartPointer< vtkImageData > image)
Sets and display the given vtkImageData.
- void [setAndDisplayMultipleImages](#) (QStringList filenames)
Set and display multiple images from a given images filenames. Display the image corresponding to the first element on the filenames list.
- void [setAndDisplayMultipleImages](#) (std::vector< vtkSmartPointer< vtkImageData > > imageStack)
Set and display multiple images from a given vtkImageData Array.
- void [displaySelectedImage](#) (int idx)
display an image stored in this imageStack.
- QString [getPixelType](#) ()
Returns the pixel type in loaded images.
- QString [getImageType](#) ()
Returns the type of image displayed.
- QString [getNumOfDimesions](#) ()
Return the numer of dimensions of the image.

- `std::vector< vtkSmartPointer
< vtkImageData > > getImageStack ()`
return this image stack
- `int * getImageSize ()`
- `int getImageWidth ()`
- `int getImageHeight ()`
- `int getXPicked ()`
Return the mouse x coordinate position when mouse left button is pressed.
- `int getYPicked ()`
Return the mouse y coordinate position when mouse left button is pressed.
- `void setXPicked (int xPosition)`
Set the mouse x coordinate position when mouse left button is pressed.
- `void setYPicked (int yPosition)`
Set the mouse y coordinate position when mouse left button is pressed.
- `vtkSmartPointer< vtkImageViewer2 > getImageViewer ()`
Return this widget image viewer.
- `QVTKWidget * getQVTKWidget ()`
Return this qvtkWidget.
- `int getImageDisplayedIndex ()`
If an image stack is loaded, then return the index in the image stack of displayed image.

Public Attributes

- `bool isImageStackLoaded`
Flag to know if it's displayed an image stack.

Private Member Functions

- `void setImageProperties (bool verbose)`
- `void displayImage (vtkImageData *image)`

Private Attributes

- `QVTKWidget * qvtkWidget`
- `ImageType::Pointer itkImage`
- `RGBImageType::Pointer rgbItkImage`
- `vtkSmartPointer< vtkImageData > vtkImage`
- `vtkSmartPointer< vtkRenderWindow > renwin`
- `vtkSmartPointer< vtkRenderer > renderer`
- `std::vector< vtkSmartPointer
< vtkImageData > > imageStack`
A vtkImageData Vector for keep the image references when load an image stack.
- `std::string pixelType`
- `int imageType`
- `size_t numDimensions`
- `int imageWidth`
- `int imageHeight`
- `int xPosition`
- `int yPosition`
- `int xPicked`
- `int yPicked`
- `int imageDisplayedIndex`

- `vtkSmartPointer< vtkImageViewer2 > imageView`
- `vtkSmartPointer`
`< vtkCornerAnnotation > cornerAnnotation`
Object for display information in the corners of the vtkImageViewer2.

5.4.1 Detailed Description

Display VTK images.

This class allows the user to display vtkUsers on QT

5.4.2 Constructor & Destructor Documentation

5.4.2.1 `QVTKImageWidget::QVTKImageWidget (QWidget * parent = 0)`

Constructor for this ImageWidget

5.4.2.2 `QVTKImageWidget::~~QVTKImageWidget ()` [virtual]

Destructor

5.4.3 Member Function Documentation

5.4.3.1 `void QVTKImageWidget::displayImage (vtkImageData * image)` [private]

Display the given vtkImage

5.4.3.2 `void QVTKImageWidget::displaySelectedImage (int idx)`

display an image stored in this imageStack.

Parameters

<i>in</i>	<i>the</i>	index in the stack position of the image
-----------	------------	--

5.4.3.3 `int QVTKImageWidget::getImageDisplayedIndex ()`

If an image stack is loaded, then return the index in the image stack of displayed image.

5.4.3.4 `int QVTKImageWidget::getImageHeigth ()`

returns this image heigth

5.4.3.5 `int* QVTKImageWidget::getImageSize ()`

returns an array with the width and height of the image

5.4.3.6 `std::vector< vtkSmartPointer< vtkImageData > > QVTKImageWidget::getImageStack ()`

return this image stack

5.4.3.7 QString QVTKImageWidget::getImageType ()

Returns the type of image displayed.

5.4.3.8 vtkSmartPointer< vtkImageViewer2 > QVTKImageWidget::getImageViewer ()

Return this widget image viewer.

Parameters

out	<i>imageViewer</i>	vtkImageViewer2 target 2D image.
-----	--------------------	----------------------------------

5.4.3.9 int QVTKImageWidget::getImageWidth ()

returns this image width

5.4.3.10 QString QVTKImageWidget::getNumOfDimesions ()

Return the numer of dimensions of the image.

5.4.3.11 QString QVTKImageWidget::getPixelType ()

Returns the pixel type in loaded images.

5.4.3.12 QVTKWidget * QVTKImageWidget::getQVTKWidget ()

Return this qvtkWidget.

Parameters

out	<i>the</i>	QVTKWidget
-----	------------	------------

5.4.3.13 int QVTKImageWidget::getXPicked ()

Return the mouse x coordinate position when mouse left button is pressed.

Parameters

out	<i>int</i>	x position
-----	------------	------------

5.4.3.14 int QVTKImageWidget::getYPicked ()

Return the mouse y coordinate position when mouse left button is pressed.

Parameters

out	<i>int</i>	y position
-----	------------	------------

5.4.3.15 void QVTKImageWidget::setAndDisplayImage (QString *imageFilename*)

Sets and display an image from a given image path

5.4.3.16 void QVTKImageWidget::setAndDisplayImage (vtkSmartPointer< vtkImageData > *image*)

Sets and display the given vtkImageData.

Parameters

in	a	vtkImageData to set and display
----	---	---------------------------------

5.4.3.17 void QVTKImageWidget::setAndDisplayMultipleImages (QStringList *filenames*)

Set and display multiple images from a given images filenames. Display the image corresponding to the first element on the filenames list.

Parameters

in	a	QStringList that contain the filename of each image
----	---	---

5.4.3.18 void QVTKImageWidget::setAndDisplayMultipleImages (std::vector< vtkSmartPointer< vtkImageData > > *imageStack*)

Set and display multiple images from a given vtkImageData Array.

Parameters

in	a	std::vector of vtkImageData
----	---	-----------------------------

5.4.3.19 void QVTKImageWidget::setImageProperties (bool *verbose*) [private]

Set the needed image properties (pixelType, imageType, num of dimensions)

5.4.3.20 void QVTKImageWidget::setXPicked (int *xPosition*)

Set the mouse x coordinate position when mouse left button is pressed.

Parameters

out	int	x position
-----	-----	------------

5.4.3.21 void QVTKImageWidget::setYPicked (int *yPosition*)

Set the mouse y coordinate position when mouse left button is pressed.

Parameters

out	int	y position
-----	-----	------------

5.4.4 Member Data Documentation

5.4.4.1 vtkSmartPointer<vtkCornerAnnotation> QVTKImageWidget::cornerAnnotation [private]

Object for display information in the corners of the vtkImageViewer2.

5.4.4.2 int QVTKImageWidget::imageDisplayedIndex [private]

If image stack is displayed this sets a reference to current image displayed

5.4.4.3 int QVTKImageWidget::imageHeight [private]

Height of the image

5.4.4.4 std::vector< vtkSmartPointer<vtkImageData> > QVTKImageWidget::imageStack [private]

A vtkImageData Vector for keep the image references when load an image stack.

5.4.4.5 int QVTKImageWidget::imageType [private]

the number of scalar components in the image 1 => grayscale, 3 => rgb

5.4.4.6 vtkSmartPointer<vtkImageViewer2> QVTKImageWidget::imageView [private]

the image viewer for display images

5.4.4.7 int QVTKImageWidget::imageWidth [private]

Width of the image

5.4.4.8 bool QVTKImageWidget::isImageStackLoaded

Flag to know if it's displayed an image stack.

5.4.4.9 ImageType::Pointer QVTKImageWidget::itkImage [private]

The grayscale image displayed in this widget

5.4.4.10 size_t QVTKImageWidget::numDimensions [private]

The number of the image dimensions

5.4.4.11 std::string QVTKImageWidget::pixelType [private]

The type of the image pixels

5.4.4.12 QVTKWidget* QVTKImageWidget::qvtkWidget [private]

The QVTKWidget for display and interact with the images

5.4.4.13 vtkSmartPointer<vtkRenderer> QVTKImageWidget::renderer [private]

The VTK renderer

5.4.4.14 `vtkSmartPointer<vtkRenderWindow> QVTKImageWidget::renwin` [private]

The VTK render window

5.4.4.15 `RGBImageType::Pointer QVTKImageWidget::rgbIImage` [private]

The RGB image displayed for this widget

5.4.4.16 `vtkSmartPointer<vtkImageData> QVTKImageWidget::vtkImage` [private]

The VTK image to display i this window

5.4.4.17 `int QVTKImageWidget::xPicked` [private]

The x coordinate of the picked position over the image

5.4.4.18 `int QVTKImageWidget::xPosition` [private]

current x coordinate of mouse position over the image

5.4.4.19 `int QVTKImageWidget::yPicked` [private]

current y coordinate of picked position over the image

5.4.4.20 `int QVTKImageWidget::yPosition` [private]

current y coordinate of mouse position over the image

The documentation for this class was generated from the following files:

- [QVTKImageWidget.h](#)
- [QVTKImageWidget.cpp](#)

5.5 QVTKImageWidgetCommand Class Reference

Interaction with mouse.

```
#include <QVTKImageWidgetCommand.h>
```

Public Member Functions

- [QVTKImageWidgetCommand](#) ()
- [~QVTKImageWidgetCommand](#) ()
- void [SetPicker](#) (vtkSmartPointer< vtkPropPicker > picker)
- void [SetAnnotation](#) (vtkSmartPointer< vtkCornerAnnotation > annotation)
- void [SetImageWidget](#) (QVTKImageWidget *imageWidget)
Set the 2d image widget related to this 2d event manager.
- virtual void [Execute](#) (vtkObject *, unsigned long vtkNotUsed(event), void *)

Static Public Member Functions

- static [QVTKImageWidgetCommand](#) * [New](#) ()

Private Attributes

- [vtkSmartPointer](#)< [vtkPropPicker](#) > [Picker](#)
- [vtkSmartPointer](#)
< [vtkCornerAnnotation](#) > [Annotation](#)
- [QVTKImageWidget](#) * [ImageWidget](#)

5.5.1 Detailed Description

Interaction with mouse.

The mouse motion callback, to pick the image and recover pixel values

5.5.2 Constructor & Destructor Documentation

5.5.2.1 [QVTKImageWidgetCommand::QVTKImageWidgetCommand](#) ()

5.5.2.2 [QVTKImageWidgetCommand::~~QVTKImageWidgetCommand](#) ()

5.5.3 Member Function Documentation

5.5.3.1 [void QVTKImageWidgetCommand::Execute](#) ([vtkObject](#) * , unsigned long *vtkNotUsedevent*, void *)
[virtual]

5.5.3.2 [QVTKImageWidgetCommand](#) * [QVTKImageWidgetCommand::New](#) () [static]

5.5.3.3 [void QVTKImageWidgetCommand::SetAnnotation](#) ([vtkSmartPointer](#)< [vtkCornerAnnotation](#) > *annotation*)

5.5.3.4 [void QVTKImageWidgetCommand::SetImageWidget](#) ([QVTKImageWidget](#) * *imageWidget*)

Set the 2d image widget related to this 2d event manager.

Parameters

in	<i>viewer</i>	QVTKImageWidget target 2D image
----	---------------	---

5.5.3.5 [void QVTKImageWidgetCommand::SetPicker](#) ([vtkSmartPointer](#)< [vtkPropPicker](#) > *picker*)

5.5.4 Member Data Documentation

5.5.4.1 [vtkSmartPointer](#)<[vtkCornerAnnotation](#)> [QVTKImageWidgetCommand::Annotation](#) [private]

Pointer to the annotation

5.5.4.2 [QVTKImageWidget](#)* [QVTKImageWidgetCommand::ImageWidget](#) [private]

The widget related to the mouse events

5.5.4.3 `vtkSmartPointer<vtkPropPicker> QVTKImageWidgetCommand::Picker` `[private]`

Pointer to the picker

The documentation for this class was generated from the following files:

- [QVTKImageWidgetCommand.h](#)
- [QVTKImageWidgetCommand.cpp](#)

Chapter 6

File Documentation

6.1 Calibration.cpp File Reference

```
#include "Calibration.h"
```

6.2 Calibration.h File Reference

```
#include "SinglePointTargetUSCalibrationParametersEstimator.h"  
#include "RANSAC.h"
```

Classes

- class [Calibration](#)
Implements LSQRRecepies methods.

6.3 main.cpp File Reference

```
#include <QtGui/QApplication>  
#include "mainwindow.h"
```

Functions

- int [main](#) (int argc, char *argv[])
Main function. Creates a new [MainWindow](#) object.

6.3.1 Function Documentation

6.3.1.1 int main (int argc, char * argv[])

Main function. Creates a new [MainWindow](#) object.

6.4 mainWindow.cpp File Reference

```
#include "ui_mainwindow.h"  
#include "mainwindow.h"  
#include "ProbeCalibrationWidget.h"  
#include <QVBoxLayout>  
#include <vtkEventQtSlotConnect.h>
```

6.5 mainWindow.h File Reference

```
#include <QMainWindow>  
#include <QtGui>  
#include "QVTKImageWidget.h"  
#include <vtkEventQtSlotConnect.h>
```

Classes

- class [MainWindow](#)
Main Window for user Interaction.

Namespaces

- namespace [Ui](#)

6.6 ProbeCalibrationWidget.cpp File Reference

```
#include "ProbeCalibrationWidget.h"  
#include "Calibration.h"  
#include <QErrorMessage>  
#include <QString>  
#include <QFile>  
#include <QTextStream>  
#include <vtkExtractVOI.h>  
#include <vn1/vn1_quaternion.h>  
#include <vn1/vn1_vector_fixed.h>  
#include <vn1/algo/vn1_levenberg_marquardt.h>  
#include <vn1/vn1_double_2.h>
```

Variables

- bool [setCoordsSize](#) = false

6.6.1 Variable Documentation

6.6.1.1 bool setCoordsSize = false

6.7 ProbeCalibrationWidget.h File Reference

```
#include "ui_ProbeCalibrationWidget.h"
#include "mainwindow.h"
#include <QWidget>
#include <vtkSmartPointer.h>
#include <vtkImageData.h>
#include <vn1/vn1_matrix.h>
#include <string>
#include <fstream>
#include <stdio.h>
```

Classes

- class [ProbeCalibrationWidget](#)

Obtain data for calibration process.

6.8 QVTKImageWidget.cpp File Reference

```
#include "QVTKImageWidget.h"
#include "QVTKImageWidgetCommand.h"
#include <QSize.h>
#include <QBoxLayout>
#include <QString>
#include <itkImage.h>
#include <itkImageFileReader.h>
#include <vtkImageReader2.h>
#include <vtkImageReader2Factory.h>
#include <vtkCornerAnnotation.h>
#include <vtkPropPicker.h>
#include <vtkTextProperty.h>
#include <vtkImageActor.h>
#include <vtkInteractorStyleImage.h>
```

6.9 QVTKImageWidget.h File Reference

```
#include <QtGui>
#include <QWidget>
#include <QVTKWidget.h>
#include <itkImage.h>
#include <itkRGBPixel.h>
#include <vtkSmartPointer.h>
#include <vtkImageData.h>
#include <vtkRenderWindow.h>
#include <vtkRenderer.h>
#include <vtkCamera.h>
#include <vtkImageActor.h>
#include <vtkCommand.h>
#include <vtkImageViewer2.h>
#include <vtkCornerAnnotation.h>
#include <vtkTransform.h>
#include <vnl/vnl_quaternion.h>
#include <vnl/vnl_matrix.h>
#include <vnl/vnl_vector.h>
```

Classes

- class [QVTKImageWidget](#)

Display VTK images.

Typedefs

- typedef itk::RGBPixel
< unsigned char > [RGBPixelType](#)
- typedef itk::Image< unsigned char > [ImageType](#)
- typedef itk::Image
< [RGBPixelType](#), 2 > [RGBImageType](#)

6.9.1 Typedef Documentation

6.9.1.1 typedef itk::Image< unsigned char > [ImageType](#)

6.9.1.2 typedef itk::Image< [RGBPixelType](#), 2> [RGBImageType](#)

6.9.1.3 typedef itk::RGBPixel< unsigned char > [RGBPixelType](#)

6.10 QVTKImageWidgetCommand.cpp File Reference

```
#include "QVTKImageWidgetCommand.h"
#include <vtkImageActor.h>
#include <vtkImageData.h>
#include <vtkInteractorStyleImage.h>
#include <vtkRenderWindow.h>
#include <vtkRenderWindowInteractor.h>
#include <vtkVariant.h>
#include <vtkMath.h>
#include <vtkCommand.h>
#include <vtkImageViewer2.h>
```

6.11 QVTKImageWidgetCommand.h File Reference

```
#include "QVTKImageWidget.h"
#include <vtkCommand.h>
#include <vtkPropPicker.h>
#include <vtkCornerAnnotation.h>
#include <vtkSmartPointer.h>
```

Classes

- class [QVTKImageWidgetCommand](#)
Interaction with mouse.

Index

- ~MainWindow
 - MainWindow, [12](#)
- ~ProbeCalibrationWidget
 - ProbeCalibrationWidget, [14](#)
- ~QVTKImageWidget
 - QVTKImageWidget, [18](#)
- ~QVTKImageWidgetCommand
 - QVTKImageWidgetCommand, [23](#)
- addImages
 - MainWindow, [12](#)
- addLogText
 - MainWindow, [12](#)
- Annotation
 - QVTKImageWidgetCommand, [23](#)
- Calibrate
 - Calibration, [10](#)
- calibrate
 - ProbeCalibrationWidget, [14](#)
- Calibration, [9](#)
 - Calibrate, [10](#)
 - ClearImagePoints, [10](#)
 - ClearTransformations, [10](#)
 - data, [10](#)
 - DataType, [10](#)
 - estimatedUSCalibrationParameters, [10](#)
 - getEstimatedUSCalibrationParameters, [10](#)
 - imagePoints, [10](#)
 - InsertImagePoints, [10](#)
 - InsertTransformations, [10](#)
 - New, [10](#)
 - transformations, [10](#)
- Calibration.cpp, [25](#)
- Calibration.h, [25](#)
- calibrationParameters
 - ProbeCalibrationWidget, [15](#)
- ClearImagePoints
 - Calibration, [10](#)
- ClearTransformations
 - Calibration, [10](#)
- Connections
 - MainWindow, [12](#)
- coords
 - ProbeCalibrationWidget, [15](#)
- cornerAnnotation
 - QVTKImageWidget, [20](#)
- crop
 - ProbeCalibrationWidget, [14](#)
- cropProbeImage
 - ProbeCalibrationWidget, [14](#)
- data
 - Calibration, [10](#)
- DataType
 - Calibration, [10](#)
- displayImage
 - QVTKImageWidget, [18](#)
- displaySelectedImage
 - MainWindow, [12](#)
 - QVTKImageWidget, [18](#)
- displayWidget
 - MainWindow, [12](#)
- estimatedUSCalibrationParameters
 - Calibration, [10](#)
- Execute
 - QVTKImageWidgetCommand, [23](#)
- getCoordinates
 - ProbeCalibrationWidget, [14](#)
- getDisplayWidget
 - MainWindow, [12](#)
- getEstimatedUSCalibrationParameters
 - Calibration, [10](#)
- getImageDisplayedIndex
 - QVTKImageWidget, [18](#)
- getImageHeight
 - QVTKImageWidget, [18](#)
- getImageSize
 - QVTKImageWidget, [18](#)
- getImageStack
 - QVTKImageWidget, [18](#)
- getImageType
 - QVTKImageWidget, [18](#)
- getImageViewer
 - QVTKImageWidget, [19](#)
- getImageWidth
 - QVTKImageWidget, [19](#)
- getNumOfDimesions
 - QVTKImageWidget, [19](#)
- getPixelType
 - QVTKImageWidget, [19](#)
- getQVTKWidget
 - QVTKImageWidget, [19](#)
- getXPicked
 - QVTKImageWidget, [19](#)
- getYPicked
 - QVTKImageWidget, [19](#)
- image

- ProbeCalibrationWidget, 15
- imageDisplayedIndex
 - QVTKImageWidget, 20
- imageHeight
 - QVTKImageWidget, 21
- imagePoints
 - Calibration, 10
- imageStack
 - ProbeCalibrationWidget, 15
 - QVTKImageWidget, 21
- ImageType
 - QVTKImageWidget.h, 28
- imageType
 - QVTKImageWidget, 21
- imageView
 - QVTKImageWidget, 21
- ImageWidget
 - QVTKImageWidgetCommand, 23
- imageWidth
 - QVTKImageWidget, 21
- imagesFileNames
 - MainWindow, 12
- InsertImagePoints
 - Calibration, 10
- InsertTransformations
 - Calibration, 10
- isImageStackLoaded
 - QVTKImageWidget, 21
- itkImage
 - QVTKImageWidget, 21
- loadRotationsFile
 - ProbeCalibrationWidget, 15
- loadTranslationsFile
 - ProbeCalibrationWidget, 15
- main
 - main.cpp, 25
- main.cpp, 25
 - main, 25
- MainWindow, 11
 - ~MainWindow, 12
 - addImages, 12
 - addLogText, 12
 - Connections, 12
 - displaySelectedImage, 12
 - displayWidget, 12
 - getDisplayWidget, 12
 - imagesFileNames, 12
 - MainWindow, 12
 - MainWindow, 12
 - print, 12
 - probeCalibration, 12
 - textOnTextArea, 12
 - ui, 13
- mainWindow
 - ProbeCalibrationWidget, 16
- mainwindow.cpp, 26
- mainwindow.h, 26

- New
 - Calibration, 10
 - QVTKImageWidgetCommand, 23
- numDimensions
 - QVTKImageWidget, 21
- Picker
 - QVTKImageWidgetCommand, 23
- pixelType
 - QVTKImageWidget, 21
- print
 - MainWindow, 12
- probeCalibration
 - MainWindow, 12
- ProbeCalibrationWidget, 13
 - ~ProbeCalibrationWidget, 14
 - calibrate, 14
 - calibrationParameters, 15
 - coords, 15
 - crop, 14
 - cropProbelImage, 14
 - getCoordinates, 14
 - image, 15
 - imageStack, 15
 - loadRotationsFile, 15
 - loadTranslationsFile, 15
 - mainWindow, 16
 - ProbeCalibrationWidget, 14
 - ProbeCalibrationWidget, 14
 - rotations, 16
 - rotations_2, 16
 - saveCalibration, 15
 - setImage, 15
 - setImageStack, 15
 - setMainWindow, 15
 - translations, 16
 - workWithStack, 16
- ProbeCalibrationWidget.cpp, 26
 - setCoordsSize, 26
- ProbeCalibrationWidget.h, 27
- QVTKImageWidget, 16
 - ~QVTKImageWidget, 18
 - cornerAnnotation, 20
 - displayImage, 18
 - displaySelectedImage, 18
 - getImageDisplayedIndex, 18
 - getImageHeight, 18
 - getImageSize, 18
 - getImageStack, 18
 - getImageType, 18
 - imageView, 19
 - getImageWidth, 19
 - getNumOfDimesions, 19
 - getPixelType, 19
 - getQVTKWidget, 19
 - getXPicked, 19
 - getYPicked, 19
 - imageDisplayedIndex, 20

- imageHeight, 21
- imageStack, 21
- imageType, 21
- imageView, 21
- imageWidth, 21
- isImageStackLoaded, 21
- itkImage, 21
- numDimensions, 21
- pixelType, 21
- QVTKImageWidget, 18
- qvtkWidget, 21
- QVTKImageWidget, 18
- renderer, 21
- renwin, 21
- rgbltkImage, 22
- setAndDisplayImage, 19
- setAndDisplayMultipleImages, 20
- setImageProperties, 20
- setXPicked, 20
- setYPicked, 20
- vtkImage, 22
- xPicked, 22
- xPosition, 22
- yPicked, 22
- yPosition, 22
- QVTKImageWidget.cpp, 27
- QVTKImageWidget.h, 28
 - ImageType, 28
 - RGBImageType, 28
 - RGBPixelType, 28
- QVTKImageWidgetCommand, 22
 - ~QVTKImageWidgetCommand, 23
 - Annotation, 23
 - Execute, 23
 - ImageWidget, 23
 - New, 23
 - Picker, 23
 - QVTKImageWidgetCommand, 23
 - QVTKImageWidgetCommand, 23
 - SetAnnotation, 23
 - SetImageWidget, 23
 - SetPicker, 23
- QVTKImageWidgetCommand.cpp, 29
- QVTKImageWidgetCommand.h, 29
- qvtkWidget
 - QVTKImageWidget, 21
- RGBImageType
 - QVTKImageWidget.h, 28
- RGBPixelType
 - QVTKImageWidget.h, 28
- renderer
 - QVTKImageWidget, 21
- renwin
 - QVTKImageWidget, 21
- rgbltkImage
 - QVTKImageWidget, 22
- rotations
 - ProbeCalibrationWidget, 16
- rotations_2
 - ProbeCalibrationWidget, 16
- saveCalibration
 - ProbeCalibrationWidget, 15
- setAndDisplayImage
 - QVTKImageWidget, 19
- setAndDisplayMultipleImages
 - QVTKImageWidget, 20
- SetAnnotation
 - QVTKImageWidgetCommand, 23
- setCoordsSize
 - ProbeCalibrationWidget.cpp, 26
- setImage
 - ProbeCalibrationWidget, 15
- setImageProperties
 - QVTKImageWidget, 20
- setImageStack
 - ProbeCalibrationWidget, 15
- SetImageWidget
 - QVTKImageWidgetCommand, 23
- setMainWindow
 - ProbeCalibrationWidget, 15
- SetPicker
 - QVTKImageWidgetCommand, 23
- setXPicked
 - QVTKImageWidget, 20
- setYPicked
 - QVTKImageWidget, 20
- textOnTextArea
 - MainWindow, 12
- transformations
 - Calibration, 10
- translations
 - ProbeCalibrationWidget, 16
- Ui, 7
- ui
 - MainWindow, 13
- vtkImage
 - QVTKImageWidget, 22
- workWithStack
 - ProbeCalibrationWidget, 16
- xPicked
 - QVTKImageWidget, 22
- xPosition
 - QVTKImageWidget, 22
- yPicked
 - QVTKImageWidget, 22
- yPosition
 - QVTKImageWidget, 22