

Qcorr: A Digital Image Correlation Program implemented in QT4

1.0

Generated by Doxygen 1.5.8

Mon Dec 7 13:47:35 2009

Contents

1	Bug List	1
2	Todo List	3
3	Module Index	5
3.1	Modules	5
4	Class Index	7
4.1	Class Hierarchy	7
5	Class Index	9
5.1	Class List	9
6	Module Documentation	11
6.1	Qcorr	11
7	Class Documentation	13
7.1	ControlsWindow Class Reference	13
7.1.1	Detailed Description	14
7.1.2	Member Function Documentation	14
7.1.2.1	getScanInterval	14
7.1.2.2	getTemplateSize	14
7.1.2.3	setEnabledSpinBoxes	15
7.2	CorrMethod Class Reference	16
7.2.1	Detailed Description	17
7.2.2	Member Function Documentation	17
7.2.2.1	getMethod	17
7.2.3	Member Data Documentation	17
7.2.3.1	m_nChosenMethod	17
7.3	ImgLabel Class Reference	19

7.3.1	Detailed Description	21
7.3.2	Member Function Documentation	21
7.3.2.1	checkTemplateRegions	21
7.3.2.2	displayCoordinatesOnStatusLabel	21
7.3.2.3	setSelectable	22
7.3.2.4	setTemplateFlags	22
7.4	Qcorr Class Reference	23
7.4.1	Detailed Description	26
7.4.2	Member Function Documentation	27
7.4.2.1	convertToGrayScaleFloat	27
7.4.2.2	displayImage	27
7.4.2.3	displayImageLabel	28
7.4.2.4	fileDumpQImage	28
7.4.2.5	findCorrelation	28
7.4.2.6	setEnabledActions	30
7.4.3	Friends And Related Function Documentation	30
7.4.3.1	ControlsWindow	30
7.4.3.2	ImgLabel	30
7.5	TargetImgLabel Class Reference	31
7.5.1	Detailed Description	32
7.5.2	Member Function Documentation	32
7.5.2.1	drawEnclosedMatch	32
7.5.2.2	overlayImage	32
7.5.2.3	setImage	33

Chapter 1

Bug List

Class [Qcorr](#) • Perhaps a few around the rubberBand functionality, and others not noticed yet.

Chapter 2

Todo List

- Class `Qcorr`**
- Use pyramids method for faster correlation
 - Better disparity map implementation
 - Spacebar opens full screen mode on image.

Chapter 3

Module Index

3.1 Modules

Here is a list of all modules:

Qcorr	11
-----------------	--------------------

Chapter 4

Class Index

4.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

ControlsWindow	13
CorrMethod	16
ImgLabel	19
TargetImgLabel	31
Ui_QcorrClass	
Ui::QcorrClass	
Qcorr	23

Chapter 5

Class Index

5.1 Class List

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

[ControlsWindow](#) (Allows to modify parameters for the correlation operations, such as template size and scan interval. This class is a friend of [Qcorr](#)) 13

[CorrMethod](#) (Dialog to select a method in which the correlation "template matching" operation will be based upon. The available correlation methods are the following:

- CROSS_CORR (cross correlation):

$$C(u, v) = \frac{\sum \{T(x, y) * I(x - u, y - v)\}}{\sqrt{\sum I(x - u, y - v)^2}}$$

- SUM_SQ_DIFF (sum of squared differences):

$$C(u, v) = \frac{\sum \{T(x, y) - I(x - u, y - v)\}^2}{\sqrt{\sum I(x - u, y - v)^2}}$$

- CORR_COEFF (correlation coefficient):

$$C(u, v) = \frac{\sum \{(T(x, y) - T_{avg}) * (I(x - u, y - v) - I_{avg})\}}{\sqrt{\sum (T(x, y) - T_{avg})^2 * \sum (I(x - u, y - v) - I_{avg})^2}}$$

) 16

[ImgLabel](#) (A sub-classed label widget implemented with the purpose of displaying the reference image on the left panel) 19

[Qcorr](#) (A Digital Image Correlation Program (Template Matching and Pixel Disparity Finder) implemented in QT4) 23

[TargetImgLabel](#) (A sub-classed label widget implemented with the purpose of displaying the target image on the right panel) 31

Chapter 6

Module Documentation

6.1 Qcorr

Classes

- class [Qcorr](#)

A Digital Image Correlation Program (Template Matching and Pixel Disparity Finder) implemented in QT4.

Chapter 7

Class Documentation

7.1 ControlsWindow Class Reference

allows to modify parameters for the correlation operations, such as template size and scan interval. This class is a friend of [Qcorr](#).

```
#include <src/controlswindow.h>
```

Public Member Functions

- [ControlsWindow](#) ([Qcorr](#) *parentWindow, QWidget *parent=0)
Constructor of [ControlsWindow](#) class must point to a [Qcorr](#) parentWindow.
- [~ControlsWindow](#) ()
Closes and Destroys itself.
- void [setEnabledSpinBoxes](#) (bool bEnable)
Enables/Disables control spinBoxes.
- int [getScanInterval](#) ()
to obtain the scan Interval set in the spin box
- int [getTemplateSize](#) ()
to obtain the template's size value (as width) set in the spin box

Private Slots

- void [setScanInterval](#) (int nValue)
Q_SLOT that sets scan interval parameter in its friend [Qcorr](#) class according to the corresponding spinBox's value.
- void [setTemplateSize](#) (int nValue)
Q_SLOT that sets template size parameter in its friend [Qcorr](#) class according to the corresponding spinBox's value.

Private Member Functions

- void [createActions](#) ()
Create Signal-Slot connections.
- void [updateParentStatusLabel](#) ()
Updates Parent's status bar with the corresponding values from the control spin boxes.

Private Attributes

- Ui::ControlsWindowClass [ui](#)
The Qt GUI form for this class, so its widgets can be accessed and manipulated.
- [Qcorr](#) * [m_parentWindow](#)
Pointer to Main Window Object.
- int [m_nScanInterval](#)
interval of scan traversal by the template
- int [m_nTemplateSize](#)
the template's size (for the width)

7.1.1 Detailed Description

allows to modify parameters for the correlation operations, such as template size and scan interval. This class is a friend of [Qcorr](#).

Definition at line 13 of file `controlswindow.h`.

7.1.2 Member Function Documentation

7.1.2.1 int ControlsWindow::getScanInterval () [inline]

to obtain the scan Interval set in the spin box

Returns:

the scan interval value (in pixels)

Definition at line 31 of file `controlswindow.h`.

References `m_nScanInterval`.

Referenced by `Qcorr::disparity()`, and `Qcorr::updateStatusLabelWithDisparityInfo()`.

7.1.2.2 int ControlsWindow::getTemplateSize () [inline]

to obtain the template's size value (as width) set in the spin box

Returns:

the scan template's size value (in pixels)

Definition at line 41 of file controlswindow.h.

References `m_nTemplateSize`.

Referenced by `Qcorr::disparity()`, and `Qcorr::updateStatusLabelWithDisparityInfo()`.

7.1.2.3 void ControlsWindow::setEnabledSpinBoxes (bool *bEnable*)

Enables/Disables control spinBoxes.

Parameters:

bEnable boolean flag to enable or disable the spinBoxes for controlling correlation parameters of the [Qcorr](#) class Set enable spinBoxes including the maximum values based on the `m_parentWindow`'s size of its right image

Definition at line 43 of file controlswindow.cpp.

References `m_parentWindow`, `Qcorr::m_rightImage`, and `ui`.

Referenced by `Qcorr::changeMouse()`, and `ControlsWindow()`.

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

- `src/controlswindow.h`
- `src/controlswindow.cpp`

7.2 CorrMethod Class Reference

Dialog to select a method in which the correlation "template matching" operation will be based upon. The available correlation methods are the following:

- CROSS_CORR (cross correlation):

$$C(u, v) = \frac{\sum \{T(x, y) * I(x - u, y - v)\}}{\sqrt{\sum I(x - u, y - v)^2}}$$

- SUM_SQ_DIFF (sum of squared differences):

$$C(u, v) = \frac{\sum \{T(x, y) - I(x - u, y - v)\}^2}{\sqrt{\sum I(x - u, y - v)^2}}$$

- CORR_COEFF (correlation coefficient):

$$C(u, v) = \frac{\sum \{(T(x, y) - T_{avg}) * (I(x - u, y - v) - I_{avg})\}}{\sqrt{\sum (T(x, y) - T_{avg})^2 * \sum (I(x - u, y - v) - I_{avg})^2}}$$

```
#include <src/corrmethod.h>
```

Public Member Functions

- [CorrMethod](#) (QWidget *parent=0)

Constructor for the [CorrMethod](#) class.

- int [getMethod](#) ()

Retrieve the selected method to be used in the correlation process.

Private Slots

- void [cancelMethod](#) ()

Q_SLOT to cancel the method selection dialog.

- void [chooseMethod](#) ()

Q_SLOT to store the method selection from the dialog.

Private Attributes

- int [m_nChosenMethod](#)
- Ui::CorrMethodClass [ui](#)

The Qt GUI form for this class, so its widgets can be accessed and manipulated.

7.2.1 Detailed Description

Dialog to select a method in which the correlation "template matching" operation will be based upon. The available correlation methods are the following:

- CROSS_CORR (cross correlation):

$$C(u, v) = \frac{\sum \{T(x, y) * I(x - u, y - v)\}}{\sqrt{\sum I(x - u, y - v)^2}}$$

- SUM_SQ_DIFF (sum of squared differences):

$$C(u, v) = \frac{\sum \{T(x, y) - I(x - u, y - v)\}^2}{\sqrt{\sum I(x - u, y - v)^2}}$$

- CORR_COEFF (correlation coefficient):

$$C(u, v) = \frac{\sum \{(T(x, y) - T_{avg}) * (I(x - u, y - v) - I_{avg})\}}{\sqrt{\sum (T(x, y) - T_{avg})^2 * \sum (I(x - u, y - v) - I_{avg})^2}}$$

Definition at line 30 of file corrmeth.h.

7.2.2 Member Function Documentation

7.2.2.1 int CorrMethod::getMethod ()

Retrieve the selected method to be used in the correlation process.

Returns the selected method in which the correlation "template matching" operation will be based upon. From the global enumeration MethodOfCorrelation, the available correlation methods are: N0_CORR_METHOD = 0, CROSS_CORR = 1, SUM_SQ_DIFF = 2, or CORR_COEFF = 3

Returns:

selected correlation method (from global enumeration MethodOfCorrelation defined in [globals.h](#)). The available correlation methods are: N0_CORR_METHOD = 0, CROSS_CORR = 1, SUM_SQ_DIFF = 2, or CORR_COEFF = 3

Definition at line 22 of file corrmeth.cpp.

References `m_nChosenMethod`.

Referenced by `Qcorr::correlate()`.

7.2.3 Member Data Documentation

7.2.3.1 int CorrMethod::m_nChosenMethod [private]

The selected method in which the correlation "template matching" operation will be based upon. From the global enumeration MethodOfCorrelation, the available correlation methods are: N0_CORR_METHOD = 0, CROSS_CORR = 1, SUM_SQ_DIFF = 2, or CORR_COEFF = 3

Definition at line 61 of file corrmeth.h.

Referenced by `cancelMethod()`, `chooseMethod()`, `CorrMethod()`, and `getMethod()`.

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

- `src/corrmethhod.h`
- `src/corrmethhod.cpp`

7.3 ImgLabel Class Reference

A sub-classed label widget implemented with the purpose of displaying the reference image on the left panel.

```
#include <src/imgLabel.h>
```

Public Member Functions

- [ImgLabel](#) ([Qcorr](#) *parentWindow, [QWidget](#) *parent=0)
Constructor of [ImgLabel](#) class must point to a [Qcorr](#) parentWindow.
- void [setSelectable](#) (bool bHasRubberBand)
Allows to enable/disable the [ImgLabel](#)'s rubberBand selection functionality.

Protected Member Functions

- void [mousePressEvent](#) ([QMouseEvent](#) *event)
press mouse event handler
- void [mouseMoveEvent](#) ([QMouseEvent](#) *event)
move mouse event handler
- void [mouseReleaseEvent](#) ([QMouseEvent](#) *event)
release mouse event handler

Private Member Functions

- void [setTemplateFlags](#) (bool bStatus)
Sets a group of flags used for the mouse pointer in relation to the template's region and edges.
- void [checkTemplateRegions](#) (int mouseX, int mouseY)
Used to check the mouse pointer in relation to the template's region and edges.
- void [displayCoordinatesOnStatusLabel](#) ([QPoint](#) &point0, [QPoint](#) &point1)
Displays the XY-coordinates of the mouse pointer on the status bar Displays the XY-coordinates of the mouse pointer (relative to the [ImgLabel](#)) in the status bar of the friend-parent's main window. It only display the mouse pointer coordinate information if rubberBand actions are enabled.

Private Attributes

- [Qcorr](#) * [m_parentWindow](#)
Pointer to Main Window Object (There should be only one on these).
- [QRubberBand](#) * [m_rubberBand](#)
the rubberBand object.

- [QPoint m_originPoint](#)
rubberBand's top-left corner point
- [QPoint m_finalPoint](#)
rubberBand's bottom-right corner point
- [QPoint m_currentPressedPoint](#)
the coordinates of the current mouse pointer when it's pressed (held-down)
- [QPoint m_labelUpperLeftCornerPoint](#)
Top-left corner of the label.
- [QPoint m_labelLowerRightCornerPoint](#)
Bottom-right corner of the label.
- [QPoint m_mousePosPoint1](#)
First mouse position point of the template selection.
- [QPoint m_mousePosPoint2](#)
Final mouse position point of the template selection.
- [bool m_bMouseIsPressed](#)
indicates whether the mouse is currently being pressed (held-down)
- [bool m_bStartedTemplateSelection](#)
indicates whether the template selection has begun
- [bool m_bMouseInTemplateRegion](#)
indicates whether the mouse pointer is inside the rubberBand selection area
- [bool m_bMouseAtTemplateTopEdge](#)
indicates whether the mouse pointer is at the top edge of the rubberBand selection area
- [bool m_bMouseAtTemplateBottomEdge](#)
indicates whether the mouse pointer is at the bottom edge of the rubberBand selection area
- [bool m_bMouseAtTemplateLeftEdge](#)
indicates whether the mouse pointer is at the left edge of the rubberBand selection area
- [bool m_bMouseAtTemplateRightEdge](#)
indicates whether the mouse pointer is at the right edge of the rubberBand selection area
- [bool m_bHasRubberBand](#)
indicates whether the rubberBand is enabled/disabled.
- [int m_nXNewPos](#)
X-value used in the rubberBand movement.
- [int m_nYNewPos](#)
Y-value used in the rubberBand movement.

Friends

- class [Qcorr](#)

[Qcorr](#) is a friend class of this [ImgLabel](#) class, so all members from [ImgLabel](#) are accessible by a [Qcorr](#) object.

7.3.1 Detailed Description

A sub-classed label widget implemented with the purpose of displaying the reference image on the left panel.

A sub-classed label widget implemented with the purpose of displaying the reference image on the left panel with added functionality such as the rubberBand selection and coordinate tracking of the mouse pointer.

Definition at line 25 of file `imgLabel.h`.

7.3.2 Member Function Documentation

7.3.2.1 void ImgLabel::checkTemplateRegions (int *mouseX*, int *mouseY*) [private]

Used to check the mouse pointer in relation to the template's region and edges.

Parameters:

mouseX X-coordinates of the mouse pointer

mouseY Y-coordinates of the mouse pointer

Definition at line 267 of file `imgLabel.cpp`.

References `m_bMouseAtTemplateBottomEdge`, `m_bMouseAtTemplateLeftEdge`, `m_bMouseAtTemplateRightEdge`, `m_bMouseAtTemplateTopEdge`, `m_bMouseInTemplateRegion`, `m_finalPoint`, `m_originPoint`, and `setTemplateFlags()`.

Referenced by `mouseMoveEvent()`, and `mouseReleaseEvent()`.

7.3.2.2 void ImgLabel::displayCoordinatesOnStatusLabel (QPoint & *point0*, QPoint & *point1*) [private]

Displays the XY-coordinates of the mouse pointer on the status bar Displays the XY-coordinates of the mouse pointer (relative to the [ImgLabel](#)) in the status bar of the friend-parent's main window. It only display the mouse pointer coordinate information if rubberBand actions are enabled.

Parameters:

point0 the coordinates of the origin point of the rubberBand selection

point1 the coordinates of the final point of the rubberBand selection

Definition at line 302 of file `imgLabel.cpp`.

References `m_bHasRubberBand`, `m_parentWindow`, and `Qcorr::m_status_label`.

Referenced by `mouseMoveEvent()`.

7.3.2.3 void ImgLabel::setSelectable (bool *bHasRubberBand*)

Allows to enable/disable the ImgLabel's rubberBand selection functionality.

Parameters:

bHasRubberBand boolean flag to enable or disable the rubberBand selection functionality.

Definition at line 43 of file imgLabel.cpp.

References m_bHasRubberBand, and m_rubberBand.

Referenced by Qcorr::changeMouse(), and ImgLabel().

7.3.2.4 void ImgLabel::setTemplateFlags (bool *bStatus*) [private]

Sets a group of flags used for the mouse pointer in relation to the template's region and edges.

Parameters:

bStatus template region's status

Definition at line 257 of file imgLabel.cpp.

References m_bMouseAtTemplateBottomEdge, m_bMouseAtTemplateLeftEdge, m_bMouseAtTemplateRightEdge, m_bMouseAtTemplateTopEdge, and m_bMouseInTemplateRegion.

Referenced by checkTemplateRegions(), ImgLabel(), and mouseMoveEvent().

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

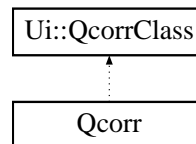
- src/imgLabel.h
- src/imgLabel.cpp

7.4 Qcorr Class Reference

A Digital Image Correlation Program (Template Matching and Pixel Disparity Finder) implemented in QT4.

```
#include <src/qcorr.h>
```

Inheritance diagram for Qcorr::



Public Member Functions

- [Qcorr](#) (QWidget *parent=0)
Constructor of the [Qcorr](#) class.
- [~Qcorr](#) ()
Destructor: House keeping that should be done when the [Qcorr](#) main window is closed – specifically, it closes the controlsWindow;.

Private Slots

- void [closeWindows](#) ()
Q_SLOT that closes the main [Qcorr](#) window and the controls Window;.
- void [showControlsWindow](#) ()
Q_SLOT that shows the controls window and raises it to the foreground if already open.
- void [browseLeftImage](#) ()
Q_SLOT that allows to browse and load an image on the left panel.
- void [browseRightImage](#) ()
Q_SLOT that allows to browse and load an image on the right panel.
- void [changeMouse](#) ()
Q_SLOT used to change the mouse pointer according to the current operation mode.
- void [viewMap](#) ()
Q_SLOT that selectively provides the appropriate existent map of results from correlation (template matching) or pixel disparity.
- void [operate](#) ()
Q_SLOT that starts the chosen operation according to the mode chosen, either template matching or disparity finder.

- void [abortOperation](#) ()
Q_SLOT triggers the flag to abort the current operation.

Private Member Functions

- void [displayImage](#) (QImage *image, QLabel *label)
Displays any QImage on a QLabel.
- void [displayImageLabel](#) (QImage *image, [ImgLabel](#) *label)
Displays the reference image on the left panel's label. which is a sub-classed widget.
- void [createActions](#) ()
Sets up actions and Q_SIGNAL-Q_SLOT connections.
- void [setEnabledActions](#) (bool bEnable)
Enables/Disables operation modes actions related widgets.
- void [setImageLabels](#) ()
Instantiates label widgets and initializes other member variables pertinent to the overall GUI functionality of the main window itself.
- void [showEStop](#) ()
emergency stop to abort the disparity operation process
- void [correlate](#) ()
Performs template matching through correlation of the selected template against the target image. It calls [findCorrelation\(\)](#) with the appropriate parameters to do a single template correlation across the entire target image.
- void [disparity](#) ()
Pixel-disparity finding operation. It should be mainly be used with stereo images that can get correlated row-by-row by using [findCorrelation\(\)](#) with the pertinent parameters and the appropriate Q_SLOT function implementation.
- float [findCorrelation](#) (const unsigned char *imgTarget, const int nWI, const int nHI, const int nDepthI, const unsigned char *imgTemplate, const int nWT, const int nHT, const int nDepthT, int &rnDx, int &rnDy, int nMethod, bool bMultires=false, int nInitialXPosition=0, int nInitialYPosition=0, int nNumberOfRows=0)
Cross-correlation of target image with template image.
- float * [convertToGrayScaleFloat](#) (const unsigned char *pchImgOriginalBits, int nSize, int nDepth)
Cast images to an 8-bit gray-scale channel of type float.
- void [updateStatusLabelWithDisparityInfo](#) ()
Updates Status Bar with Information about parameters for the disparity operation.
- bool [fileDumpQImage](#) (const QString &fileName)
Used for testing and investing the bits stored in QImages or other Qt Widgets.

Private Attributes

- `int m_nXCorrelationCoordinate`
the resulting x-coordinate match obtained from correlation of a template against a target
- `int m_nYCorrelationCoordinate`
the resulting y-coordinate match obtained from correlation of a template against a target
- `CorrMethod * m_corrMethodDialog`
Dialog Box to choose a correlation method.
- `ControlsWindow * m_controlsWindow`
Dialog Box to modify correlation parameters such as template size and scan interval.
- `QString initialName`
String that saves the path for the first file image that has been loaded.
- `QImage * m_leftImage`
Left Panel's Image (a.k.a. "reference image" when performing template matching through correlation).
- `QImage * m_rightImage`
Right Panel's Image (a.k.a "target image").
- `QImage * m_templateImage`
Template Image that is selected from the reference image (in the left panel) that will be matched against.
- `QImage * m_corrMapImage`
Correlation Map resulting from the Template Matching action.
- `QImage * m_disparityMapImage`
Disparity Map that results from the pixel disparities found between the left and right images.
- `ImgLabel * m_leftImage_label`
label used to display the left image. This is sub-classed widget implemented with the functionality of allowing template selection by a rectangular rubber-band
- `TargetImgLabel * m_targetImage_label`
a sub-classed label widget implemented with the purpose of displaying the target image in the right panel with added functionality such as the visualization of the found match
- `QLabel * m_status_label`
status label that updates the mouse-pointer's position coordinates as it moves and selects templates. It also provides other types of information when required.
- `QVector< QRgb > * m_grayColorTab`
8-bit gray-scale color table
- `QVector< QRgb > * m_greenColorTab`
8-bit green-scale color table
- `QPoint m_matchingPoint`

upper-left corner point where the correlation match was found

- [QSize m_templateSize](#)
the current template's size as a QSize object
- [QActionGroup * modes_actionGroup](#)
group of menu actions for the mode of operation.
- [QDialogButtonBox * m_eStopDialog](#)
A dialogue to stop the current correlation process.
- [bool m_bHasLeftImage](#)
indicates that an image is loaded in the left panel
- [bool m_bHasRightImage](#)
indicates that an image is loaded in the right panel
- [bool m_bHasCorrMap](#)
indicates that a correlation map exists
- [bool m_bHasDisparityMap](#)
indicates that a disparity map exists
- [bool m_bEstop](#)
emergency stop for the
- [int m_nScanInterval](#)
interval of scan traversal by the template

Friends

- [class ImgLabel](#)
the [ImgLabel](#) class is a friend of [Qcorr](#) in order to make all members of [Qcorr](#) accessible by an [ImgLabel](#) object
- [class ControlsWindow](#)
allows to modify parameters for the correlation operations, such as template size and scan interval. This class is a friend of [Qcorr](#).

7.4.1 Detailed Description

A Digital Image Correlation Program (Template Matching and Pixel Disparity Finder) implemented in QT4.

This class combines all other sub-classed QWidgets implemented throughout the program. The main correlation used in the "Template matching" and "Disparity Finder" modes rely strongly in the functionality implemented in this class through the procedure [convertToGrayScaleFloat\(\)](#)

Note:

Correlation is performed on gray-scale (1 channel image) that are computed in this procedure through the [convertToGrayScaleFloat\(\)](#) function.

Compile-time dependencies

- The QT4 Framework
- g++

Bug

- Perhaps a few around the rubberBand functionality, and others not noticed yet.

Todo

- Use pyramids method for faster correlation
- Better disparity map implementation
- Spacebar opens full screen mode on image.

Authors:

Carlos Jaramillo
Joel Gonzalez

Definition at line 52 of file qcorr.h.

7.4.2 Member Function Documentation

7.4.2.1 `float * Qcorr::convertToGrayScaleFloat (const unsigned char * pchImgOriginalBits, int nSize, int nDepth) [private]`

Cast images to an 8-bit gray-scale channel of type float.

Parameters:

pchImgOriginalBits Buffer of unsigned characters as the source image
nSize Number of pixels in the source image
nDepth Pixel depth of the source image

Returns:

the correlation number computed by method (directly). Additionally, the (dx,dy) offset of the template for which there exists a best match.

Definition at line 1296 of file qcorr.cpp.

Referenced by `findCorrelation()`.

7.4.2.2 `void Qcorr::displayImage (QImage * image, QLabel * label) [private]`

Displays any QImage on a QLabel.

Parameters:

image Pointer to a QImage instance

label Pointer to a QLabel instance

Definition at line 1279 of file qcorr.cpp.

7.4.2.3 void Qcorr::displayImageLabel (QImage * *image*, ImgLabel * *label*) [private]

Displays the reference image on the left panel's label. which is a sub-classed widget.

Parameters:

image Pointer to a QImage instance

label Pointer to a [ImgLabel](#) class instance

Definition at line 1286 of file qcorr.cpp.

References [ImgLabel::m_labelLowerRightCornerPoint](#).

Referenced by [browseLeftImage\(\)](#).

7.4.2.4 bool Qcorr::fileDumpQImage (const QString & *fileName*) [private]

Used for testing and investing the bits stored in QImages or other Qt Widgets.

Parameters:

fileName a QString reference to the file-name as which the raw bits will be stored
 Buffer of unsigned characters as the source image

Returns:

true if the bits were properly dumped to the specified fileName

Definition at line 1353 of file qcorr.cpp.

References [m_rightImage](#).

7.4.2.5 float Qcorr::findCorrelation (const unsigned char * *imgTarget*, const int *nWI*, const int *nHI*, const int *nDepthI*, const unsigned char * *imgTemplate*, const int *nWT*, const int *nHT*, const int *nDepthT*, int & *rnDx*, int & *rnDy*, int *nMethod*, bool *bMultires* = false, int *nInitialXPosition* = 0, int *nInitialYPosition* = 0, int *nNumberOfRows* = 0) [private]

Cross-correlation of target image with template image.

Note:

Correlation is performed on gray-scale (1 channel image) that are computed in this procedure through the [convertToGrayScaleFloat\(\)](#) function.

Parameters:

imgTarget Buffer of unsigned characters containing the target image where correlation match is to be found

nWI Width of target image

nHI Height of target image

nDepthI Pixel depth of target image

imgTemplate Buffer of unsigned characters containing the template image

nWT Width of target image

nHT Height of target image

nDepthT Pixel depth of target image

rnDx x-coordinate where the highest level of correlation match is found (top-left corner of the match)

rnDy y-coordinate where the highest level of correlation match is found (top-left corner of the match)

nMethod Determines the selected method to be used in the correlation process. The method is a globally defined enumeration. The available methods are:

- CROSS_CORR (cross correlation):

$$C(u, v) = \frac{\sum \{T(x, y) * I(x - u, y - v)\}}{\sqrt{\sum I(x - u, y - v)^2}}$$

- SUM_SQ_DIFF (sum of squared differences):

$$C(u, v) = \frac{\sum \{T(x, y) - I(x - u, y - v)\}^2}{\sqrt{\sum I(x - u, y - v)^2}}$$

- CORR_COEFF (correlation coefficient):

$$C(u, v) = \frac{\sum \{(T(x, y) - T_{avg}) * (I(x - u, y - v) - I_{avg})\}}{\sqrt{\sum (T(x, y) - T_{avg})^2 * \sum (I(x - u, y - v) - I_{avg})^2}}$$

Remark: The square root of the sum of the squares (RSS) is being used to calculate the aggregate accuracy of a measurement when the accuracies of the all the measuring devices are known. The average accuracy is not merely the arithmetic average of the accuracies (or uncertainties), nor is it the sum of them. Note how the RSS result in this case is greater than the largest of the values under the radical.

bMultires Determines if multiresolution correlation should be applied, by making use of image pyramids. With multiresolution, the correlation can be determined faster than direct correlation. Default is false.

nInitialXPosition Indicates the x-coordinate of the pixel on the target image where correlation should start. Default is 0

nInitialYPosition Indicates the y-coordinate of the pixel on the target image where correlation should start. Default is 0

nNumberOfRows Indicates the number of rows on the target image to be scanned. Default is 0, which means all of them.

Returns:

a float array resulting from the conversion of the source image into an 8-bit gray-scale image.

Definition at line 643 of file qcorr.cpp.

References `convertToGrayScaleFloat()`, `m_bEstop`, `m_bHasCorrMap`, `m_corrMapImage`, and `m_grayColorTab`.

Referenced by `correlate()`, and `disparity()`.

7.4.2.6 void Qcorr::setEnabledActions (bool *bEnable*) [private]

Enables/Disables operation modes actions related widgets.

Parameters:

bEnable boolean flag to enable or disable the operation mode's actions widgets as well as the related widgets such as, start_pushButton and controls_pushButton

Definition at line 167 of file qcorr.cpp.

References changeMouse().

Referenced by browseLeftImage(), and browseRightImage().

7.4.3 Friends And Related Function Documentation

7.4.3.1 friend class ControlsWindow [friend]

allows to modify parameters for the correlation operations, such as template size and scan interval. This class is a friend of [Qcorr](#).

[ControlsWindow](#)

Definition at line 87 of file qcorr.h.

Referenced by setImageLabels().

7.4.3.2 friend class ImgLabel [friend]

the [ImgLabel](#) class is a friend of [Qcorr](#) in order to make all members of [Qcorr](#) accessible by an [ImgLabel](#) object

[ImgLabel](#)

Definition at line 82 of file qcorr.h.

Referenced by setImageLabels().

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

- src/qcorr.h
- src/qcorr.cpp

7.5 TargetImgLabel Class Reference

A sub-classed label widget implemented with the purpose of displaying the target image on the right panel.

```
#include <src/targetImgLabel.h>
```

Public Member Functions

- [TargetImgLabel](#) (QWidget *parent=0)
Constructor for the [TargetImgLabel](#) class.
- void [setImage](#) (const QImage &labelImage)
Sets background image on this [TargetImgLabel](#) class.
- void [overlayImage](#) (const QImage &otherImage, int nXoffset=0, int nYoffset=0)
Overlays an Image ontop of the current one.
- void [drawEnclosedMatch](#) (const QPoint originPoint, const QSize rectSize)
Draws a dashed, green rectangle to indicate the position of a matching template on the target image.
- void [eraseEnclosedMatch](#) ()
Erases the green rectangle that indicated a matching template on the target image.

Protected Member Functions

- void [paintEvent](#) (QPaintEvent *)
Handles the paint events by refreshing/updating the contents of this [TargetImgLabel](#) class.

Private Attributes

- QImage * [m_image](#)
the right image (a.k.a target image)
- QImage * [m_overlayImage](#)
the overlaid correlation map image
- QPoint [m_originPoint](#)
the point that indicates the top-left coordinates of the matching rectangle
- QSize [m_rectSize](#)
the size of the enclosing match rectangle
- int [m_nXoffset](#)
X offset for overlay image.
- int [m_nYoffset](#)
Y offset for overlay image.

- bool [m_bHasCorrResults](#)
indicates whether correlation results exist so an overlaid image can be composed on top of the target image
- bool [m_bHasImage](#)
indicates whether this [TargetImgLabel](#) class has a target (main) image
- bool [m_bHasOverlayImage](#)
indicates whether this [TargetImgLabel](#) class has an overlaid image (Usually, a correlation results map)

7.5.1 Detailed Description

A sub-classed label widget implemented with the purpose of displaying the target image on the right panel.

A sub-classed label widget implemented with the purpose of displaying the target image in the right panel with added functionality such as the enclosing match visualization and overlaid correlation map images

Definition at line 23 of file `targetImgLabel.h`.

7.5.2 Member Function Documentation

7.5.2.1 void TargetImgLabel::drawEnclosedMatch (const QPoint *originPoint*, const QSize *rectSize*)

Draws a dashed, green rectangle to indicate the position of a matching template on the target image.

Parameters:

originPoint A QPoint that indicates the top-left coordinates of the matching rectangle

rectSize A QSize for the matching rectangle to be drawn

Definition at line 99 of file `targetImgLabel.cpp`.

References `m_bHasCorrResults`, `m_originPoint`, and `m_rectSize`.

Referenced by `Qcorr::correlate()`, and `Qcorr::disparity()`.

7.5.2.2 void TargetImgLabel::overlayImage (const QImage & *otherImage*, int *nXoffset* = 0, int *nYoffset* = 0)

Overlays an Image ontop of the current one.

Parameters:

otherImage Pointer to a QImage instance that will overlay the current one using Qt::CompositionMode_Screen mode

nXoffset X offset for overlay image. Default is 0

nYoffset Y offset for overlay image. Default is 0

Definition at line 37 of file `targetImgLabel.cpp`.

References `m_bHasOverlayImage`, `m_nXoffset`, `m_nYoffset`, and `m_overlayImage`.

Referenced by `Qcorr::viewMap()`.

7.5.2.3 void TargetImgLabel::setImage (const QImage & *labelImage*)

Sets background image on this [TargetImgLabel](#) class.

Parameters:

labelImage Pointer to a QImage instance that will be used as the label's background

Definition at line 26 of file targetImgLabel.cpp.

References `m_bHasImage`, `m_bHasOverlayImage`, and `m_image`.

Referenced by `Qcorr::browseRightImage()`, `Qcorr::changeMouse()`, `Qcorr::correlate()`, and `Qcorr::viewMap()`.

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

- `src/targetImgLabel.h`
- `src/targetImgLabel.cpp`

Index

- checkTemplateRegions
 - ImgLabel, [21](#)
- ControlsWindow, [13](#)
 - getScanInterval, [14](#)
 - getTemplateSize, [14](#)
 - Qcorr, [30](#)
 - setEnabledSpinBoxes, [15](#)
- convertToGrayScaleFloat
 - Qcorr, [27](#)
- CorrMethod, [16](#)
 - getMethod, [17](#)
 - m_nChosenMethod, [17](#)
- displayCoordinatesOnStatusLabel
 - ImgLabel, [21](#)
- displayImage
 - Qcorr, [27](#)
- displayImageLabel
 - Qcorr, [28](#)
- drawEnclosedMatch
 - TargetImgLabel, [32](#)
- fileDumpQImage
 - Qcorr, [28](#)
- findCorrelation
 - Qcorr, [28](#)
- getMethod
 - CorrMethod, [17](#)
- getScanInterval
 - ControlsWindow, [14](#)
- getTemplateSize
 - ControlsWindow, [14](#)
- ImgLabel, [19](#)
 - checkTemplateRegions, [21](#)
 - displayCoordinatesOnStatusLabel, [21](#)
 - Qcorr, [30](#)
 - setSelectable, [21](#)
 - setTemplateFlags, [22](#)
- m_nChosenMethod
 - CorrMethod, [17](#)
- overlayImage
 - TargetImgLabel, [32](#)
- Qcorr, [11](#), [23](#)
 - ControlsWindow, [30](#)
 - convertToGrayScaleFloat, [27](#)
 - displayImage, [27](#)
 - displayImageLabel, [28](#)
 - fileDumpQImage, [28](#)
 - findCorrelation, [28](#)
 - ImgLabel, [30](#)
 - setEnabledActions, [29](#)
- setEnabledActions
 - Qcorr, [29](#)
- setEnabledSpinBoxes
 - ControlsWindow, [15](#)
- setImage
 - TargetImgLabel, [32](#)
- setSelectable
 - ImgLabel, [21](#)
- setTemplateFlags
 - ImgLabel, [22](#)
- TargetImgLabel, [31](#)
 - drawEnclosedMatch, [32](#)
 - overlayImage, [32](#)
 - setImage, [32](#)