

LabRoboticsProject

Generated by Doxygen 1.8.14

Contents

| | | |
|----------|--|----------|
| 1 | Class Index | 1 |
| 1.1 | Class List | 1 |
| 2 | File Index | 3 |
| 2.1 | File List | 3 |
| 3 | Class Documentation | 5 |
| 3.1 | Settings Class Reference | 5 |
| 3.1.1 | Member Enumeration Documentation | 7 |
| 3.1.1.1 | InputType | 7 |
| 3.1.1.2 | Pattern | 7 |
| 3.1.2 | Constructor & Destructor Documentation | 7 |
| 3.1.2.1 | Settings() | 7 |
| 3.1.3 | Member Function Documentation | 7 |
| 3.1.3.1 | isListOfImages() | 7 |
| 3.1.3.2 | nextImage() | 8 |
| 3.1.3.3 | read() | 8 |
| 3.1.3.4 | readStringList() | 8 |
| 3.1.3.5 | validate() | 9 |
| 3.1.3.6 | write() | 9 |
| 3.1.4 | Member Data Documentation | 10 |
| 3.1.4.1 | aspectRatio | 10 |
| 3.1.4.2 | atImageList | 10 |
| 3.1.4.3 | boardSize | 10 |

| | | |
|----------|-------------------------------------|----|
| 3.1.4.4 | <code>calibFixPrincipalPoint</code> | 10 |
| 3.1.4.5 | <code>calibrationPattern</code> | 10 |
| 3.1.4.6 | <code>calibZeroTangentDist</code> | 11 |
| 3.1.4.7 | <code>cameraID</code> | 11 |
| 3.1.4.8 | <code>delay</code> | 11 |
| 3.1.4.9 | <code>fixK1</code> | 11 |
| 3.1.4.10 | <code>fixK2</code> | 11 |
| 3.1.4.11 | <code>fixK3</code> | 11 |
| 3.1.4.12 | <code>fixK4</code> | 12 |
| 3.1.4.13 | <code>fixK5</code> | 12 |
| 3.1.4.14 | <code>flag</code> | 12 |
| 3.1.4.15 | <code>flipVertical</code> | 12 |
| 3.1.4.16 | <code>goodInput</code> | 12 |
| 3.1.4.17 | <code>imageList</code> | 12 |
| 3.1.4.18 | <code>input</code> | 12 |
| 3.1.4.19 | <code>inputCapture</code> | 13 |
| 3.1.4.20 | <code>inputType</code> | 13 |
| 3.1.4.21 | <code>nrFrames</code> | 13 |
| 3.1.4.22 | <code>outputFileName</code> | 13 |
| 3.1.4.23 | <code>showUndistorsed</code> | 13 |
| 3.1.4.24 | <code>squareSize</code> | 13 |
| 3.1.4.25 | <code>useFisheye</code> | 14 |
| 3.1.4.26 | <code>writeExtrinsics</code> | 14 |
| 3.1.4.27 | <code>writePoints</code> | 14 |

| | |
|---|-----------|
| 4 File Documentation | 15 |
| 4.1 src/calibration.cc File Reference | 15 |
| 4.1.1 Function Documentation | 16 |
| 4.1.1.1 calcBoardCornerPositions() | 16 |
| 4.1.1.2 calibration() | 16 |
| 4.1.1.3 computeReprojectionErrors() | 16 |
| 4.1.1.4 read() | 17 |
| 4.1.1.5 runCalibration() | 17 |
| 4.1.1.6 runCalibrationAndSave() | 18 |
| 4.1.1.7 saveCameraParams() | 19 |
| 4.2 src/calibration.hh File Reference | 19 |
| 4.2.1 Detailed Description | 20 |
| 4.2.2 Enumeration Type Documentation | 20 |
| 4.2.2.1 anonymous enum | 20 |
| 4.2.3 Function Documentation | 21 |
| 4.2.3.1 calibration() | 21 |
| 4.2.3.2 runCalibrationAndSave() | 21 |
| 4.3 src/calibration_run.cc File Reference | 22 |
| 4.3.1 Function Documentation | 22 |
| 4.3.1.1 main() | 22 |
| 4.4 src/create_xml.cc File Reference | 22 |
| 4.4.1 Function Documentation | 23 |
| 4.4.1.1 main() | 23 |
| 4.5 src/detection.cc File Reference | 23 |
| 4.5.1 Function Documentation | 23 |
| 4.5.1.1 detection() | 24 |
| 4.5.1.2 erode_dilation() | 24 |
| 4.5.1.3 find_contours() | 24 |
| 4.5.1.4 load_number_template() | 24 |
| 4.5.1.5 number_recognition() | 24 |

| | | |
|---------|---|----|
| 4.5.1.6 | save_convex_hull() | 24 |
| 4.5.1.7 | shape_detection() | 25 |
| 4.5.2 | Variable Documentation | 25 |
| 4.5.2.1 | fs_xml | 25 |
| 4.5.2.2 | templates | 25 |
| 4.5.2.3 | xml_settings | 25 |
| 4.6 | src/detection.hh File Reference | 25 |
| 4.6.1 | Function Documentation | 26 |
| 4.6.1.1 | detection() | 26 |
| 4.6.1.2 | erode_dilation() | 26 |
| 4.6.1.3 | find_contours() | 26 |
| 4.6.1.4 | load_number_template() | 27 |
| 4.6.1.5 | number_recognition() | 27 |
| 4.6.1.6 | save_convex_hull() | 27 |
| 4.6.1.7 | shape_detection() | 27 |
| 4.7 | src/detection_run.cc File Reference | 27 |
| 4.7.1 | Function Documentation | 28 |
| 4.7.1.1 | main() | 28 |
| 4.8 | src/main.cc File Reference | 28 |
| 4.8.1 | Function Documentation | 28 |
| 4.8.1.1 | main() | 28 |
| 4.9 | src/unwrapping.cc File Reference | 28 |
| 4.9.1 | Macro Definition Documentation | 29 |
| 4.9.1.1 | area_ratio | 29 |
| 4.9.2 | Function Documentation | 29 |
| 4.9.2.1 | distance() | 29 |
| 4.9.2.2 | loadCoefficients() | 29 |
| 4.9.2.3 | swap() | 30 |
| 4.9.2.4 | unwrapping() | 30 |
| 4.9.3 | Variable Documentation | 30 |

| | | |
|--------------|--------------------------------------|-----------|
| 4.9.3.1 | xml_settings | 30 |
| 4.10 | src/unwrapping.hh File Reference | 30 |
| 4.10.1 | Function Documentation | 31 |
| 4.10.1.1 | loadCoefficients() | 31 |
| 4.10.1.2 | unwrapping() | 31 |
| 4.11 | src/unwrapping_run.cc File Reference | 31 |
| 4.11.1 | Function Documentation | 32 |
| 4.11.1.1 | main() | 32 |
| 4.12 | src/utlis.cc File Reference | 32 |
| 4.12.1 | Function Documentation | 32 |
| 4.12.1.1 | my_imshow() | 32 |
| 4.13 | src/utlis.hh File Reference | 33 |
| 4.13.1 | Macro Definition Documentation | 33 |
| 4.13.1.1 | INFO | 33 |
| 4.13.2 | Function Documentation | 33 |
| 4.13.2.1 | my_imshow() | 34 |
| Index | | 35 |

Chapter 1

Class Index

1.1 Class List

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

| | |
|------------------------------------|---|
| Settings | 5 |
|------------------------------------|---|

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

| | |
|-----------------------------------|----|
| src/calibration.cc | 15 |
| src/calibration.hh | |
| Library for calibration | 19 |
| src/calibration_run.cc | 22 |
| src/create_xml.cc | 22 |
| src/detection.cc | 23 |
| src/detection.hh | 25 |
| src/detection_run.cc | 27 |
| src/main.cc | 28 |
| src/unwrapping.cc | 28 |
| src/unwrapping.hh | 30 |
| src/unwrapping_run.cc | 31 |
| src/utils.cc | 32 |
| src/utils.hh | 33 |

Chapter 3

Class Documentation

3.1 Settings Class Reference

```
#include <calibration.hh>
```

Public Types

- enum `Pattern` { `NOT_EXISTING`, `CHESSBOARD`, `CIRCLES_GRID`, `ASYMMETRIC_CIRCLES_GRID` }
- enum `InputType` { `INVALID`, `CAMERA`, `VIDEO_FILE`, `IMAGE_LIST` }

Public Member Functions

- `Settings ()`
Constructor that sets `goodInput` to false.
- void `write` (FileStorage &fs) const
Write serialization.
- void `read` (const FileNode &node)
Read serialization.
- void `validate ()`
This function validate the content of the file.
- Mat `nextImage ()`
Get next image from list, or next frame from video.

Static Public Member Functions

- static bool `readStringList` (const string &filename, vector< string > &l)
Read from file a list of images.
- static bool `isListOfImages` (const string &filename)
Check if the file from which is trying to retrieve a list is a valid format (xml or yaml).

Public Attributes

- Size [boardSize](#)
The size of the board -> Number of items by width and height.
- [Pattern calibrationPattern](#)
One of the Chessboard, circles, or asymmetric circle pattern.
- float [squareSize](#)
The size of a square in your defined unit (point, millimeter,etc).
- int [nrFrames](#)
The number of frames to use from the input for calibration.
- float [aspectRatio](#)
The aspect ratio.
- int [delay](#)
In case of a video input.
- bool [writePoints](#)
Write detected feature points.
- bool [writeExtrinsics](#)
Write extrinsic parameters.
- bool [calibZeroTangentDist](#)
Assume zero tangential distortion.
- bool [calibFixPrincipalPoint](#)
Fix the principal point at the center.
- bool [flipVertical](#)
Flip the captured images around the horizontal axis.
- string [outputFileName](#)
The name of the file where to write.
- bool [showUndistorted](#)
Show undistorted images after calibration.
- string [input](#)
The input.
- bool [useFisheye](#)
use fisheye camera model for calibration
- bool [fixK1](#)
fix K1 distortion coefficient
- bool [fixK2](#)
fix K2 distortion coefficient
- bool [fixK3](#)
fix K3 distortion coefficient
- bool [fixK4](#)
fix K4 distortion coefficient
- bool [fixK5](#)
fix K5 distortion coefficient
- int [cameraID](#)
- vector< string > [imageList](#)
- size_t [atImageList](#)
- VideoCapture [inputCapture](#)
- [InputType](#) [inputType](#)
- bool [goodInput](#)
- int [flag](#)

3.1.1 Member Enumeration Documentation

3.1.1.1 InputType

```
enum Settings::InputType
```

Enumerator

| | |
|------------|--|
| INVALID | |
| CAMERA | |
| VIDEO_FILE | |
| IMAGE_LIST | |

3.1.1.2 Pattern

```
enum Settings::Pattern
```

Enumerator

| | |
|-------------------------|--|
| NOT_EXISTING | |
| CHESSBOARD | |
| CIRCLES_GRID | |
| ASYMMETRIC_CIRCLES_GRID | |

3.1.2 Constructor & Destructor Documentation

3.1.2.1 Settings()

```
Settings::Settings ( ) [inline]
```

Constructor that sets `goodInput` to false.

3.1.3 Member Function Documentation

3.1.3.1 isListOfImages()

```
bool Settings::isListOfImages (
    const string & filename ) [static]
```

Check if the file from which is trying to retrieve a list is a valid format (xml or yaml).

Parameters

| | | |
|----|-----------------|---|
| in | <i>filename</i> | The name of the file to check for validity. |
|----|-----------------|---|

Returns

`false` is the file is not xml or yaml
`true` otherwise.

3.1.3.2 nextImage()

```
Mat Settings::nextImage ( )
```

Get next image from list, or next frame from video.

Returns

A matrix containing the next image to consider.

3.1.3.3 read()

```
void Settings::read (
    const FileNode & node )
```

Read serialization.

This function read data from a file and stores each node in their corresponding variables.

Parameters

| | | |
|----|-------------|-----------------------------------|
| in | <i>node</i> | The node of the file to consider. |
|----|-------------|-----------------------------------|

3.1.3.4 readStringList()

```
bool Settings::readStringList (
    const string & filename,
    vector< string > & l ) [static]
```

Read from file a list of images.

Parameters

| | | |
|-----|-----------------|--|
| in | <i>filename</i> | The name of the file from which to read. |
| out | <i>l</i> | A vector which will contain the names of the file from the list. |

Returns

false if the file could not be opened or if the file doesn't contain a list
true otherwise.

3.1.3.5 validate()

```
void Settings::validate ( )
```

This function validate the content of the file.

Even though this function doesn't return anything nor has any parameters for output, it sets a variable of the [Settings](#) class, that is `googInput`, to `false` if some infos were wrong. `true` otherwise. The options it takes in consideration are the following:

- Size must be positive.
- Cells must be greater than 10^{-6} .
- The number of frames considered, that is images, must be greater than 0.
- Check for valid input, that is a valid list of images.
- If input is a number than a camera is being used.
- Else a list of image is being used.
- If inputType is CAMERA than open the camera for input.
- If inputType is VIDEO_FILE than open the video for input.
- If inputType is not IMAGE_LIST but inputCapture is opened, than inputType must be set to invalid.
- Check the field pattern: if it doesn't correspond to a known one than it's invalid.

3.1.3.6 write()

```
void Settings::write (
    FileStorage & fs ) const
```

Write serialization.

This function write data to a file.

Parameters

| | | |
|-----------------|-----------------|------------------------------|
| <code>in</code> | <code>fs</code> | The filename where to write. |
|-----------------|-----------------|------------------------------|

3.1.4 Member Data Documentation

3.1.4.1 aspectRatio

```
float Settings::aspectRatio
```

The aspect ratio.

3.1.4.2 atImageList

```
size_t Settings::atImageList
```

3.1.4.3 boardSize

```
Size Settings::boardSize
```

The size of the board -> Number of items by width and height.

3.1.4.4 calibFixPrincipalPoint

```
bool Settings::calibFixPrincipalPoint
```

Fix the principal point at the center.

3.1.4.5 calibrationPattern

```
Pattern Settings::calibrationPattern
```

One of the Chessboard, circles, or asymmetric circle pattern.

3.1.4.6 calibZeroTangentDist

```
bool Settings::calibZeroTangentDist
```

Assume zero tangential distortion.

3.1.4.7 cameraID

```
int Settings::cameraID
```

3.1.4.8 delay

```
int Settings::delay
```

In case of a video input.

3.1.4.9 fixK1

```
bool Settings::fixK1
```

fix K1 distortion coefficient

3.1.4.10 fixK2

```
bool Settings::fixK2
```

fix K2 distortion coefficient

3.1.4.11 fixK3

```
bool Settings::fixK3
```

fix K3 distortion coefficient

3.1.4.12 fixK4

```
bool Settings::fixK4
```

fix K4 distortion coefficient

3.1.4.13 fixK5

```
bool Settings::fixK5
```

fix K5 distortion coefficient

3.1.4.14 flag

```
int Settings::flag
```

3.1.4.15 flipVertical

```
bool Settings::flipVertical
```

Flip the captured images around the horizontal axis.

3.1.4.16 goodInput

```
bool Settings::goodInput
```

3.1.4.17 imageList

```
vector<string> Settings::imageList
```

3.1.4.18 input

```
string Settings::input
```

The input.

3.1.4.19 inputCapture

```
VideoCapture Settings::inputCapture
```

3.1.4.20 inputType

```
InputType Settings::inputType
```

3.1.4.21 nrFrames

```
int Settings::nrFrames
```

The number of frames to use from the input for calibration.

3.1.4.22 outputFileName

```
string Settings::outputFileName
```

The name of the file where to write.

3.1.4.23 showUndistorted

```
bool Settings::showUndistorted
```

Show undistorted images after calibration.

3.1.4.24 squareSize

```
float Settings::squareSize
```

The size of a square in your defined unit (point, millimeter,etc).

3.1.4.25 useFisheye

```
bool Settings::useFisheye
```

use fisheye camera model for calibration

3.1.4.26 writeExtrinsics

```
bool Settings::writeExtrinsics
```

Write extrinsic parameters.

3.1.4.27 writePoints

```
bool Settings::writePoints
```

Write detected feature points.

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

- [src/calibration.hh](#)
- [src/calibration.cc](#)

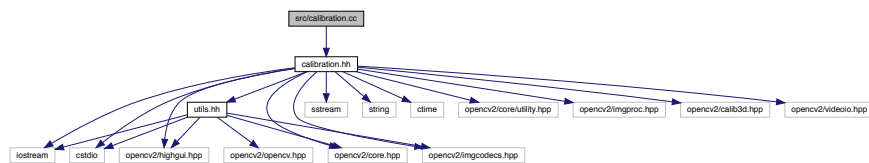
Chapter 4

File Documentation

4.1 src/calibration.cc File Reference

```
#include "calibration.hh"
```

Include dependency graph for calibration.cc:



Functions

- int [calibration](#) (const string inputFile)
Function to run the complete calibration.
- static void [read](#) (const FileNode &node, [Settings](#) &x, const [Settings](#) &default_value)
Reads settings from file. If there is none then initiate a new [Settings](#).
- static double [computeReprojectionErrors](#) (const vector< vector< Point3f > > &objectPoints, const vector< vector< Point2f > > &imagePoints, const vector< Mat > &rvecs, const vector< Mat > &tvecs, const Mat &cameraMatrix, const Mat &distCoeffs, vector< float > &perViewErrors, bool fisheye)
Compute the errors of the projection.
- void [calcBoardCornerPositions](#) (Size boardSize, float squareSize, vector< Point3f > &corners, [Settings::Pattern](#) patternType)
This function compute the position of the upper corners of every cell.
- static bool [runCalibration](#) ([Settings](#) &s, Size &imageSize, Mat &cameraMatrix, Mat &distCoeffs, vector< vector< Point2f > > imagePoints, vector< Mat > &rvecs, vector< Mat > &tvecs, vector< float > &reprojErrs, double &totalAvgErr)
This function run the calibration creating the matrixed for the camera and the distorsion coefficients.
- static void [saveCameraParams](#) (const [Settings](#) &s, const Size &imageSize, const Mat &cameraMatrix, const Mat &distCoeffs, const vector< Mat > &rvecs, const vector< Mat > &tvecs, const vector< float > &reprojErrs, const vector< vector< Point2f > > &imagePoints, const double totalAvgErr)
Function to save the computed parameters to a file.
- bool [runCalibrationAndSave](#) ([Settings](#) &s, Size imageSize, Mat &cameraMatrix, Mat &distCoeffs, vector< vector< Point2f > > imagePoints)
Reads settings from file. If there is none then initiate a new [Settings](#).

4.1.1 Function Documentation

4.1.1.1 calcBoardCornerPositions()

```
void calcBoardCornerPositions (
    Size boardSize,
    float squareSize,
    vector< Point3f > & corners,
    Settings::Pattern patternType )
```

This function compute the position of the upper corners of every cell.

Parameters

| | | |
|-----|--------------------|---|
| in | <i>boardSiz</i> | the dimension of the chess board. |
| in | <i>squareSize</i> | the dimension of the edge of a cell. |
| in | <i>patternType</i> | the type of pattern following <code>Setting::Pattern</code> . |
| out | <i>corners</i> | a vector of <code>Point3fs</code> which equals to the corners of the cells. |

4.1.1.2 calibration()

```
int calibration (
    const string inputFile )
```

Function to run the complete calibration.

Parameters

| | | |
|----|------------------|--|
| in | <i>inputFile</i> | Name of the setting.xml file. It's set to default to default.xml |
|----|------------------|--|

Returns

- 2 if the settings file could be load but the input was not well-formed
- 1 if the settings file could not be opened.
- 0 if everything went fine.

4.1.1.3 computeReprojectionErrors()

```
static double computeReprojectionErrors (
    const vector< vector< Point3f > > & objectPoints,
    const vector< vector< Point2f > > & imagePoints,
    const vector< Mat > & rvecs,
```



```

const vector< Mat > & tvecs,
const Mat & cameraMatrix,
const Mat & distCoeffs,
vector< float > & perViewErrors,
bool fisheye ) [static]

```

Compute the errors of the projection.

Parameters

| | | |
|-----|----------------------|---|
| in | <i>objectPoints</i> | The real image points which will be projected |
| in | <i>rvecs</i> | Input vector of rotation vectors estimated for each pattern view. |
| in | <i>tvecs</i> | Input vector of translation vectors estimated for each pattern view. |
| in | <i>cameraMatrix</i> | The matrix containing the parameters for the camera |
| in | <i>distCoeffs</i> | The matrix containing the distortion coefficients. |
| in | <i>fisheye</i> | A variable which says if a fish eye correction should be applied or no. |
| out | <i>perViewErrors</i> | A vector containing the error for each image. |
| out | <i>imagePoints</i> | The projected points for each image. |

Returns

The total error.

4.1.1.4 read()

```

static void read (
    const FileNode & node,
    Settings & x,
    const Settings & default_value ) [inline], [static]

```

Reads settings from file. If there is none then initiate a new [Settings](#).

Parameters

| | | |
|----|----------------------|--|
| in | <i>node</i> | node to consider for getting settings; |
| in | <i>x</i> | Settings to configure; |
| in | <i>default_value</i> | Settings default value. Setted to Settings() . |

4.1.1.5 runCalibration()

```

static bool runCalibration (
    Settings & s,
    Size & imageSize,
    Mat & cameraMatrix,
    Mat & distCoeffs,

```

```

vector< vector< Point2f > > imagePoints,
vector< Mat > & rvecs,
vector< Mat > & tvecs,
vector< float > & reprojErrs,
double & totalAvgErr ) [static]

```

This function run the calibration creating the matrixed for the camera and the distorsion coefficients.

Parameters

| | | |
|-----|---------------------|--|
| in | <i>s</i> | The Settings read from the file and memorized. |
| in | <i>imageSize</i> | The size of the image used in <code>calibrateCamera()</code> to initialize the camera matrix. |
| in | <i>imagePoints</i> | The projected points for each image. |
| in | <i>reprojErrs</i> | The re-projection error, that is a geometric error corresponding to the image distance between a projected point and a measured one. |
| out | <i>cameraMatrix</i> | The matrix of the camera parameters |
| out | <i>distCoeffs</i> | The matrix of the distorsion coefficients. |
| out | <i>rvecs</i> | Output vector of rotation vectors estimated for each pattern view. |
| out | <i>tvecs</i> | Output vector of translation vectors estimated for each pattern view. |
| out | <i>totalAvgErr</i> | The total avarage error given from distorsion. |

Returns

`false` if one or more elements in the `cameraMatrix` and `distCoeffs` are invalid.
`true` if all the elements are valid.

4.1.1.6 runCalibrationAndSave()

```

bool runCalibrationAndSave (
    Settings & s,
    Size imageSize,
    Mat & cameraMatrix,
    Mat & distCoeffs,
    vector< vector< Point2f > > imagePoints )

```

Reads settings from file. If there is none then initiate a new [Settings](#).

Parameters

| | | |
|-----|---------------------|---|
| in | <i>s</i> | The Settings being used during the execution. |
| in | <i>imageSize</i> | The dimensions of the images. |
| in | <i>imagePoints</i> | The projected points for each image. |
| out | <i>cameraMatrix</i> | The matrix which is used to store the values for the camera parameters. |
| out | <i>distCoeffs</i> | The matrix which is used to store the distortion coefficients. |

Returns

`true` if the calibration succeeded.
`false` otherwise.

4.1.1.7 saveCameraParams()

```
static void saveCameraParams (
    const Settings & s,
    const Size & imageSize,
    const Mat & cameraMatrix,
    const Mat & distCoeffs,
    const vector< Mat > & rvecs,
    const vector< Mat > & tvecs,
    const vector< float > & reprojErrs,
    const vector< vector< Point2f > > & imagePoints,
    const double totalAvgErr ) [static]
```

Function to save the computed parameters to a file.

Parameters

| | | |
|----|---------------------|--|
| in | <i>s</i> | Use the Settings got at the beginning for information as the output file name, image and board size. |
| in | <i>imageSize</i> | The size of the image. |
| in | <i>cameraMatrix</i> | The camera matrix. |
| in | <i>distCoeffs</i> | The distortion coefficient matrix. |
| | <i>[int]</i> | rvecs Vector of rotation vectors estimated for each pattern view. |
| in | <i>tvecs</i> | Vector of translation vectors estimated for each pattern view. |
| in | <i>reprojErrs</i> | The re-projection error, that is a geometric error corresponding to the image distance between a projected point and a measured one. |
| in | <i>imagePoints</i> | The projected points for each image. |
| in | <i>totalAvgErr</i> | The total average error given from distortion. |

Open file for writing

Stores time of calibration

Store infos about the images

4.2 src/calibration.hh File Reference

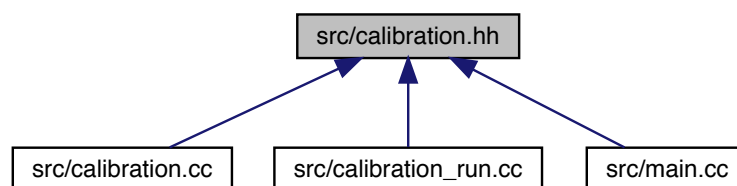
Library for calibration.

```
#include "utils.hh"
#include <iostream>
#include <sstream>
#include <string>
#include <ctime>
#include <cstdio>
#include <opencv2/core.hpp>
#include <opencv2/core/utility.hpp>
#include <opencv2/imgproc.hpp>
#include <opencv2/calib3d.hpp>
```

```
#include <opencv2/imgcodecs.hpp>
#include <opencv2/videoio.hpp>
#include <opencv2/highgui.hpp>
Include dependency graph for calibration.hh:
```



This graph shows which files directly or indirectly include this file:



Classes

- class [Settings](#)

Enumerations

- enum { [DETECTION](#) = 0, [CAPTURING](#) = 1, [CALIBRATED](#) = 2 }

Functions

- int [calibration](#) (const string inputFile="data/calib_config.xml")
Function to run the complete calibration.
- bool [runCalibrationAndSave](#) ([Settings](#) &s, Size imageSize, Mat &cameraMatrix, Mat &distCoeffs, vector< vector< Point2f > > imagePoints)
Reads settings from file. If there is none then initiate a new [Settings](#).

4.2.1 Detailed Description

Library for calibration.

4.2.2 Enumeration Type Documentation

4.2.2.1 anonymous enum

anonymous enum

Enumerator

| | |
|------------|--|
| DETECTION | |
| CAPTURING | |
| CALIBRATED | |

4.2.3 Function Documentation

4.2.3.1 calibration()

```
int calibration (
    const string inputFile )
```

Function to run the complete calibration.

Parameters

| | | |
|----|------------------|--|
| in | <i>inputFile</i> | Name of the setting.xml file. It's set to default to default.xml |
|----|------------------|--|

Returns

-2 if the settings file could be load but the input was not well-formed
 -1 if the settings file could not be opened.
 0 if everything went fine.

4.2.3.2 runCalibrationAndSave()

```
bool runCalibrationAndSave (
    Settings & s,
    Size imageSize,
    Mat & cameraMatrix,
    Mat & distCoeffs,
    vector< vector< Point2f > > imagePoints )
```

Reads settings from file. If there is none then initiate a new [Settings](#).

Parameters

| | | |
|-----|---------------------|---|
| in | <i>s</i> | The Settings being used during the execution. |
| in | <i>imageSize</i> | The dimensions of the images. |
| in | <i>imagePoints</i> | The projected points for each image. |
| out | <i>cameraMatrix</i> | The matrix which is used to store the values for the camera parameters. |
| out | <i>distCoeffs</i> | The matrix which is used to store the distortion coefficients. |

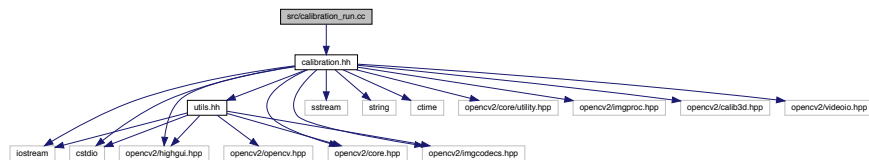
Returns

`true` if the calibration succeeded.
`false` otherwise.

4.3 src/calibration_run.cc File Reference

```
#include "calibration.hh"
```

Include dependency graph for `calibration_run.cc`:



Functions

- `int` [main](#) ()

4.3.1 Function Documentation

4.3.1.1 main()

```
int main ( )
```

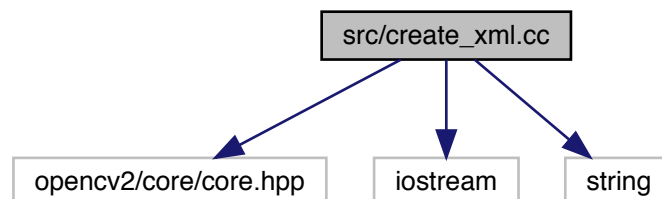
4.4 src/create_xml.cc File Reference

```
#include <opencv2/core/core.hpp>
```

```
#include <iostream>
```

```
#include <string>
```

Include dependency graph for `create_xml.cc`:



Functions

- int [main](#) ()

4.4.1 Function Documentation

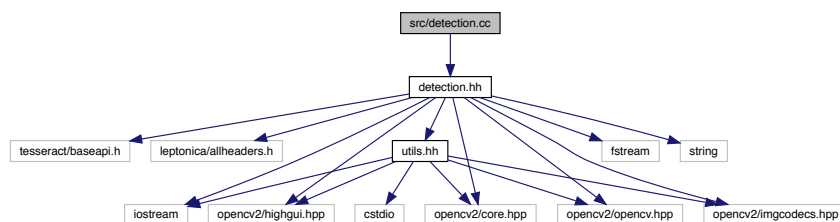
4.4.1.1 main()

```
int main ( )
```

4.5 src/detection.cc File Reference

```
#include "detection.hh"
```

Include dependency graph for detection.cc:



Functions

- int [detection](#) ()
- void [load_number_template](#) ()
- void [shape_detection](#) (Mat img, const int color)
- void [erode_dilation](#) (Mat &img, const int color)
- void [find_contours](#) (const Mat &img, Mat original, const int color)
- void [save_convex_hull](#) (vector< vector< Point >> &contours, const int color, vector< int > victims)
- int [number_recognition](#) (Rect blob, const Mat &base)

Variables

- const string [xml_settings](#) = "data/settings.xml"
- FileStorage [fs_xml](#)
- vector< Mat > [templates](#)

4.5.1 Function Documentation

4.5.1.1 detection()

```
int detection ( )
```

4.5.1.2 erode_dilation()

```
void erode_dilation (
    Mat & img,
    const int color )
```

4.5.1.3 find_contours()

```
void find_contours (
    const Mat & img,
    Mat original,
    const int color )
```

4.5.1.4 load_number_template()

```
void load_number_template ( )
```

4.5.1.5 number_recognition()

```
int number_recognition (
    Rect blob,
    const Mat & base )
```

4.5.1.6 save_convex_hull()

```
void save_convex_hull (
    vector< vector< Point >> & contours,
    const int color,
    vector< int > victims )
```


4.5.1.7 shape_detection()

```
void shape_detection (
    Mat img,
    const int color )
```

4.5.2 Variable Documentation

4.5.2.1 fs_xml

```
FileStorage fs_xml
```

4.5.2.2 templates

```
vector<Mat> templates
```

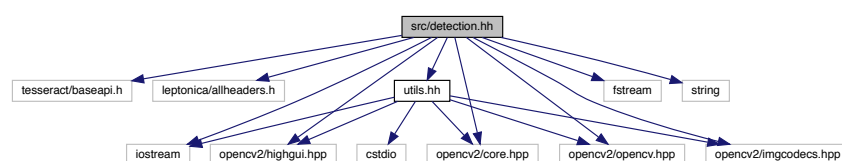
4.5.2.3 xml_settings

```
const string xml_settings = "data/settings.xml"
```

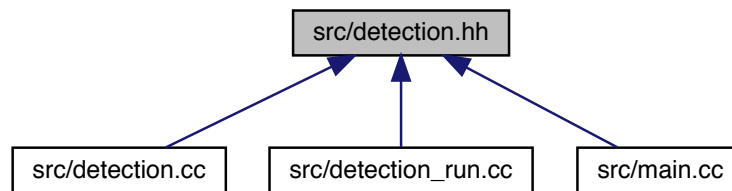
4.6 src/detection.hh File Reference

```
#include <tesseract/baseapi.h>
#include <leptonica/allheaders.h>
#include "utils.hh"
#include <iostream>
#include <fstream>
#include <string>
#include <opencv2/highgui.hpp>
#include <opencv2/core.hpp>
#include <opencv2/opencv.hpp>
#include <opencv2/imgcodecs.hpp>
```

Include dependency graph for detection.hh:



This graph shows which files directly or indirectly include this file:



Functions

- int [detection](#) ()
- void [shape_detection](#) (Mat img, const int color)
- void [erode_dilation](#) (Mat &img, const int color)
- void [find_contours](#) (const Mat &img, Mat original, const int color)
- void [save_convex_hull](#) (vector< vector< Point >> &contours, const int color, vector< int > victims={})
- int [number_recognition](#) (Rect blob, const Mat &base)
- void [load_number_template](#) ()

4.6.1 Function Documentation

4.6.1.1 [detection\(\)](#)

```
int detection ( )
```

4.6.1.2 [erode_dilation\(\)](#)

```
void erode_dilation (
    Mat & img,
    const int color )
```

4.6.1.3 [find_contours\(\)](#)

```
void find_contours (
    const Mat & img,
    Mat original,
    const int color )
```

4.6.1.4 load_number_template()

```
void load_number_template ( )
```

4.6.1.5 number_recognition()

```
int number_recognition (
    Rect blob,
    const Mat & base )
```

4.6.1.6 save_convex_hull()

```
void save_convex_hull (
    vector< vector< Point >> & contours,
    const int color,
    vector< int > victims = {} )
```

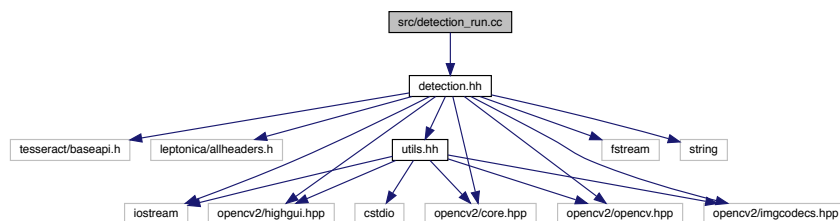
4.6.1.7 shape_detection()

```
void shape_detection (
    Mat img,
    const int color )
```

4.7 src/detection_run.cc File Reference

```
#include "detection.hh"
```

Include dependency graph for detection_run.cc:



Functions

- int [main](#) ()

4.7.1 Function Documentation

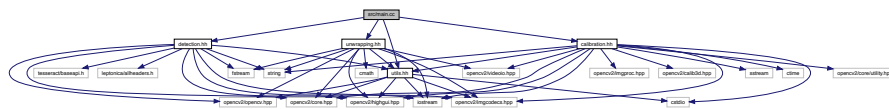
4.7.1.1 main()

```
int main ( )
```

4.8 src/main.cc File Reference

```
#include "utils.hh"
#include "detection.hh"
#include "unwrapping.hh"
#include "calibration.hh"
```

Include dependency graph for main.cc:



Functions

- int [main](#) ()

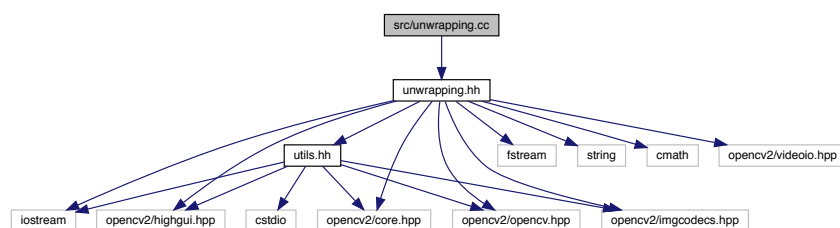
4.8.1 Function Documentation

4.8.1.1 main()

```
int main ( )
```

4.9 src/unwrapping.cc File Reference

```
#include "unwrapping.hh"
Include dependency graph for unwrapping.cc:
```



Macros

- `#define` [area_ratio](#) 0.7

Functions

- static float [distance](#) (Point c1, Point c2)
- static void [swap](#) (int &a, int &b)
- int [unwrapping](#) ()
- void [loadCoefficients](#) (const string &filename, Mat &camera_matrix, Mat &dist_coeffs)

Variables

- const string [xml_settings](#) = "data/settings.xml"

4.9.1 Macro Definition Documentation

4.9.1.1 area_ratio

```
#define area_ratio 0.7
```

4.9.2 Function Documentation

4.9.2.1 distance()

```
static float distance (  
    Point c1,  
    Point c2 ) [static]
```

4.9.2.2 loadCoefficients()

```
void loadCoefficients (  
    const string & filename,  
    Mat & camera_matrix,  
    Mat & dist_coeffs )
```

4.9.2.3 swap()

```
static void swap (
    int & a,
    int & b ) [static]
```

4.9.2.4 unwrapping()

```
int unwrapping ( )
```

4.9.3 Variable Documentation

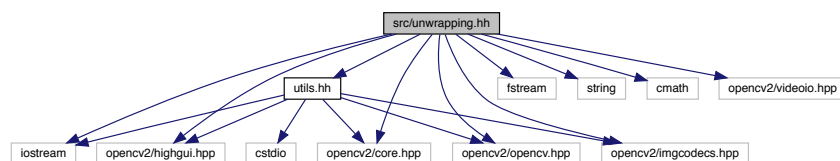
4.9.3.1 xml_settings

```
const string xml_settings = "data/settings.xml"
```

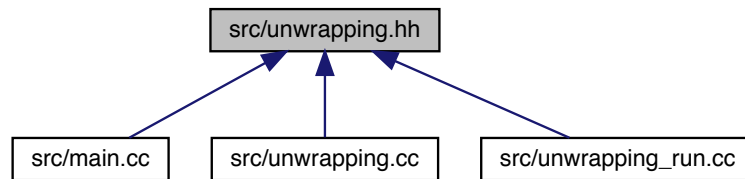
4.10 src/unwrapping.hh File Reference

```
#include "utils.hh"
#include <iostream>
#include <fstream>
#include <string>
#include <cmath>
#include <opencv2/videoio.hpp>
#include <opencv2/highgui.hpp>
#include <opencv2/core.hpp>
#include <opencv2/opencv.hpp>
#include <opencv2/imgcodecs.hpp>
```

Include dependency graph for unwrapping.hh:



This graph shows which files directly or indirectly include this file:



Functions

- int [unwrapping](#) ()
- void [loadCoefficients](#) (const string &filename, Mat &camera_matrix, Mat &dist_coeffs)

4.10.1 Function Documentation

4.10.1.1 loadCoefficients()

```

void loadCoefficients (
    const string & filename,
    Mat & camera_matrix,
    Mat & dist_coeffs )
  
```

4.10.1.2 unwrapping()

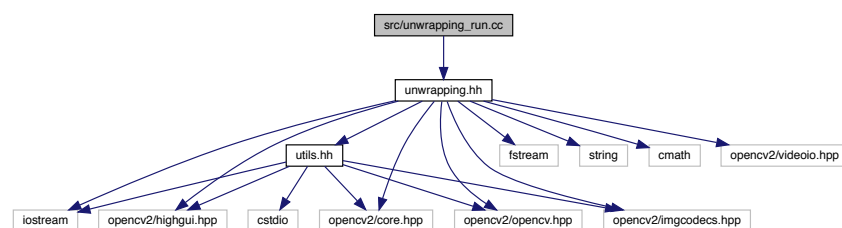
```

int unwrapping ( )
  
```

4.11 src/unwrapping_run.cc File Reference

```
#include "unwrapping.hh"
```

Include dependency graph for unwrapping_run.cc:



Functions

- int [main](#) ()

4.11.1 Function Documentation

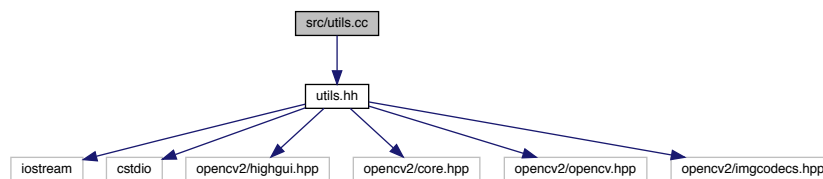
4.11.1.1 main()

```
int main ( )
```

4.12 src/Utils.cc File Reference

```
#include "utils.hh"
```

Include dependency graph for utils.cc:



Functions

- void [my_imshow](#) (const char *win_name, Mat img, bool reset)

4.12.1 Function Documentation

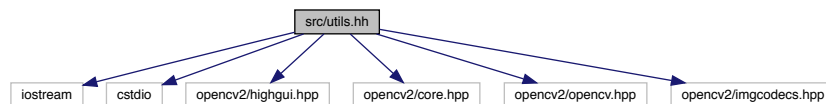
4.12.1.1 my_imshow()

```
void my_imshow (
    const char * win_name,
    Mat img,
    bool reset )
```

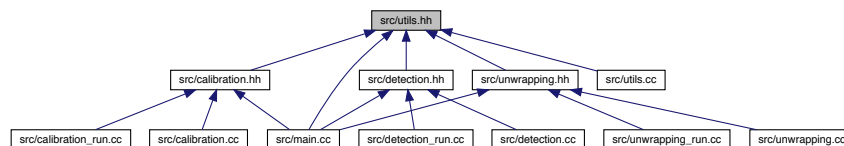

4.13 src/utils.hh File Reference

```
#include <iostream>
#include <cstdio>
#include <opencv2/highgui.hpp>
#include <opencv2/core.hpp>
#include <opencv2/opencv.hpp>
#include <opencv2/imgcodecs.hpp>
```

Include dependency graph for utils.hh:



This graph shows which files directly or indirectly include this file:



Macros

- `#define` [INFO\(msg\)](#)

Functions

- `void` [my_imshow](#) (const char *win_name, Mat img, bool reset=false)

4.13.1 Macro Definition Documentation

4.13.1.1 INFO

```
#define INFO(  
    msg )
```

4.13.2 Function Documentation

4.13.2.1 my_imshow()

```
void my_imshow (
    const char * win_name,
    Mat img,
    bool reset = false )
```

Index

- area_ratio
 - [unwrapping.cc, 29](#)
- aspectRatio
 - [Settings, 10](#)
- atImageList
 - [Settings, 10](#)
- boardSize
 - [Settings, 10](#)
- calcBoardCornerPositions
 - [calibration.cc, 16](#)
- calibFixPrincipalPoint
 - [Settings, 10](#)
- calibZeroTangentDist
 - [Settings, 10](#)
- calibration
 - [calibration.cc, 16](#)
 - [calibration.hh, 21](#)
- calibration.cc
 - [calcBoardCornerPositions, 16](#)
 - [calibration, 16](#)
 - [computeReprojectionErrors, 16](#)
 - [read, 17](#)
 - [runCalibration, 17](#)
 - [runCalibrationAndSave, 18](#)
 - [saveCameraParams, 19](#)
- calibration.hh
 - [calibration, 21](#)
 - [runCalibrationAndSave, 21](#)
- calibration_run.cc
 - [main, 22](#)
- calibrationPattern
 - [Settings, 10](#)
- cameraID
 - [Settings, 11](#)
- computeReprojectionErrors
 - [calibration.cc, 16](#)
- create_xml.cc
 - [main, 23](#)
- delay
 - [Settings, 11](#)
- detection
 - [detection.cc, 23](#)
 - [detection.hh, 26](#)
- detection.cc
 - [detection, 23](#)
 - [erode_dilation, 24](#)
 - [find_contours, 24](#)
 - [fs_xml, 25](#)
 - [load_number_template, 24](#)
 - [number_recognition, 24](#)
 - [save_convex_hull, 24](#)
 - [shape_detection, 24](#)
 - [templates, 25](#)
 - [xml_settings, 25](#)
- detection.hh
 - [detection, 26](#)
 - [erode_dilation, 26](#)
 - [find_contours, 26](#)
 - [load_number_template, 26](#)
 - [number_recognition, 27](#)
 - [save_convex_hull, 27](#)
 - [shape_detection, 27](#)
- detection_run.cc
 - [main, 28](#)
- distance
 - [unwrapping.cc, 29](#)
- erode_dilation
 - [detection.cc, 24](#)
 - [detection.hh, 26](#)
- find_contours
 - [detection.cc, 24](#)
 - [detection.hh, 26](#)
- fixK1
 - [Settings, 11](#)
- fixK2
 - [Settings, 11](#)
- fixK3
 - [Settings, 11](#)
- fixK4
 - [Settings, 11](#)
- fixK5
 - [Settings, 12](#)
- flag
 - [Settings, 12](#)
- flipVertical
 - [Settings, 12](#)
- fs_xml
 - [detection.cc, 25](#)
- goodInput
 - [Settings, 12](#)
- INFO
 - [utils.hh, 33](#)
- imageList

- Settings, 12
- input
 - Settings, 12
- inputCapture
 - Settings, 12
- InputType
 - Settings, 7
- inputType
 - Settings, 13
- isListOfImages
 - Settings, 7
- load_number_template
 - detection.cc, 24
 - detection.hh, 26
- loadCoefficients
 - unwrapping.cc, 29
 - unwrapping.hh, 31
- main
 - calibration_run.cc, 22
 - create_xml.cc, 23
 - detection_run.cc, 28
 - main.cc, 28
 - unwrapping_run.cc, 32
- main.cc
 - main, 28
- my_imshow
 - utils.cc, 32
 - utils.hh, 33
- nextImage
 - Settings, 8
- nrFrames
 - Settings, 13
- number_recognition
 - detection.cc, 24
 - detection.hh, 27
- outputFileName
 - Settings, 13
- Pattern
 - Settings, 7
- read
 - calibration.cc, 17
 - Settings, 8
- readStringList
 - Settings, 8
- runCalibration
 - calibration.cc, 17
- runCalibrationAndSave
 - calibration.cc, 18
 - calibration.hh, 21
- save_convex_hull
 - detection.cc, 24
 - detection.hh, 27
- saveCameraParams
 - calibration.cc, 19
- Settings, 5
 - aspectRatio, 10
 - atImageList, 10
 - boardSize, 10
 - calibFixPrincipalPoint, 10
 - calibZeroTangentDist, 10
 - calibrationPattern, 10
 - cameraID, 11
 - delay, 11
 - fixK1, 11
 - fixK2, 11
 - fixK3, 11
 - fixK4, 11
 - fixK5, 12
 - flag, 12
 - flipVertical, 12
 - goodInput, 12
 - imageList, 12
 - input, 12
 - inputCapture, 12
 - InputType, 7
 - inputType, 13
 - isListOfImages, 7
 - nextImage, 8
 - nrFrames, 13
 - outputFileName, 13
 - Pattern, 7
 - read, 8
 - readStringList, 8
 - Settings, 7
 - showUndistorsed, 13
 - squareSize, 13
 - useFisheye, 13
 - validate, 9
 - write, 9
 - writeExtrinsics, 14
 - writePoints, 14
- shape_detection
 - detection.cc, 24
 - detection.hh, 27
- showUndistorsed
 - Settings, 13
- squareSize
 - Settings, 13
- src/calibration.cc, 15
- src/calibration.hh, 19
- src/calibration_run.cc, 22
- src/create_xml.cc, 22
- src/detection.cc, 23
- src/detection.hh, 25
- src/detection_run.cc, 27
- src/main.cc, 28
- src/unwrapping.cc, 28
- src/unwrapping.hh, 30
- src/unwrapping_run.cc, 31
- src/utils.cc, 32
- src/utils.hh, 33

- swap
 - [unwrapping.cc](#), [29](#)
- templates
 - [detection.cc](#), [25](#)
- unwrapping
 - [unwrapping.cc](#), [30](#)
 - [unwrapping.hh](#), [31](#)
- [unwrapping.cc](#)
 - [area_ratio](#), [29](#)
 - [distance](#), [29](#)
 - [loadCoefficients](#), [29](#)
 - [swap](#), [29](#)
 - [unwrapping](#), [30](#)
 - [xml_settings](#), [30](#)
- [unwrapping.hh](#)
 - [loadCoefficients](#), [31](#)
 - [unwrapping](#), [31](#)
- [unwrapping_run.cc](#)
 - [main](#), [32](#)
- useFisheye
 - [Settings](#), [13](#)
- [utils.cc](#)
 - [my_imshow](#), [32](#)
- [utils.hh](#)
 - [INFO](#), [33](#)
 - [my_imshow](#), [33](#)
- validate
 - [Settings](#), [9](#)
- write
 - [Settings](#), [9](#)
- writeExtrinsics
 - [Settings](#), [14](#)
- writePoints
 - [Settings](#), [14](#)
- [xml_settings](#)
 - [detection.cc](#), [25](#)
 - [unwrapping.cc](#), [30](#)