

Ohm - Android Version

1.0

Generated by Doxygen 1.8.12

Contents

1	Module Index	1
1.1	Modules	1
2	Namespace Index	3
2.1	Packages	3
3	Hierarchical Index	5
3.1	Class Hierarchy	5
4	Class Index	7
4.1	Class List	7
5	Module Documentation	9
5.1	CameraInput	9
5.1.1	Detailed Description	9
5.2	UserInterface	10
5.2.1	Detailed Description	10
5.3	ColourMapping	11
5.3.1	Detailed Description	11
5.4	ValueCalculator	12
5.4.1	Detailed Description	12
5.5	BandLocation	13
5.5.1	Detailed Description	13

6	Namespace Documentation	15
6.1	Package imageprocessing	15
6.1.1	Detailed Description	15
6.2	Package valueidentification	15
6.2.1	Detailed Description	15
7	Class Documentation	17
7.1	ca.ryanmarks.ohm.imageprocessing.BandReader Class Reference	17
7.1.1	Detailed Description	17
7.1.2	Member Function Documentation	17
7.1.2.1	read()	17
7.2	ca.ryanmarks.ohm.userinterface.CameraActivity Class Reference	18
7.2.1	Member Function Documentation	18
7.2.1.1	onCameraFrame()	18
7.2.1.2	onCameraViewStarted()	19
7.2.1.3	onCameraViewStopped()	19
7.2.1.4	onCreate()	19
7.2.1.5	onPause()	19
7.2.1.6	onResume()	19
7.3	ca.ryanmarks.ohm.Pair< T, T1 > Class Template Reference	19
7.3.1	Detailed Description	20
7.4	ca.ryanmarks.ohm.valueidentification.ResistorColour Enum Reference	20
7.4.1	Detailed Description	21
7.4.2	Constructor & Destructor Documentation	21
7.4.2.1	ResistorColour()	21
7.4.3	Member Function Documentation	21
7.4.3.1	fit() [1/2]	21
7.4.3.2	fit() [2/2]	22
7.5	ca.ryanmarks.ohm.input.ScanningCameraView Class Reference	22
7.5.1	Member Function Documentation	22
7.5.1.1	initializeCamera()	22
7.6	ca.ryanmarks.ohm.valueidentification.ValueCalculator Class Reference	23
7.6.1	Detailed Description	23
	Index	25

Chapter 1

Module Index

1.1 Modules

Here is a list of all modules:

CameraInput	9
UserInterface	10
ColourMapping	11
ValueCalculator	12
BandLocation	13

Chapter 2

Namespace Index

2.1 Packages

Here are the packages with brief descriptions (if available):

imageprocessing	15
valueidentification	15

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

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

ca.ryanmarks.ohm.imageprocessing.BandReader	17
CvCameraViewListener2	
ca.ryanmarks.ohm.userinterface.CameraActivity	18
ca.ryanmarks.ohm.Pair< T, T1 >	19
ca.ryanmarks.ohm.valueidentification.ResistorColour	20
ca.ryanmarks.ohm.valueidentification.ValueCalculator	23
AppCompatActivity	
ca.ryanmarks.ohm.userinterface.CameraActivity	18
JavaCameraView	
ca.ryanmarks.ohm.input.ScanningCameraView	22

Chapter 4

Class Index

4.1 Class List

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

ca.ryanmarks.ohm.imageprocessing.BandReader	
Module used to analyze the line of pixels selected by the user through the UI. It uses high values of the differential of the RGB colours to detect edges of bands	17
ca.ryanmarks.ohm.userinterface.CameraActivity	18
ca.ryanmarks.ohm.Pair< T, T1 >	
This class represents a generic key, value pair	19
ca.ryanmarks.ohm.valueidentification.ResistorColour	
Enum containing all of the possible colours that a resistor can take on. Also features member functions used to map the colours of bands to values used in the calculation process	20
ca.ryanmarks.ohm.input.ScanningCameraView	22
ca.ryanmarks.ohm.valueidentification.ValueCalculator	
Object used to calculate the resistance of the resistor based on the mapped colours	23

Chapter 5

Module Documentation

5.1 CameraInput

Classes

- class [ca.ryanmarks.ohm.input.ScanningCameraView](#)

5.1.1 Detailed Description

Author

Ryan Marks

5.2 UserInterface

Classes

- class [ca.ryanmarks.ohm.userinterface.CameraActivity](#)

5.2.1 Detailed Description

Author

Ryan Marks

5.3 ColourMapping

Classes

- enum [ca.ryanmarks.ohm.valueidentification.ResistorColour](#)

Enum containing all of the possible colours that a resistor can take on. Also features member functions used to map the colours of bands to values used in the calculation process.

5.3.1 Detailed Description

Author

Jonathan Brown

5.4 ValueCalculator

Classes

- class [ca.ryanmarks.ohm.valueidentification.ValueCalculator](#)

Object used to calculate the resistance of the resistor based on the mapped colours.

5.4.1 Detailed Description

Author

Jonathan Brown

5.5 BandLocation

Classes

- class [ca.ryanmarks.ohm.imageprocessing.BandReader](#)

Module used to analyze the line of pixels selected by the user through the UI. It uses high values of the differential of the RGB colours to detect edges of bands.

5.5.1 Detailed Description

Module used to analyze the line of pixels selected by the user through the UI. It uses high values of the differential of the RGB colours to detect edges of bands.

Chapter 6

Namespace Documentation

6.1 Package imageprocessing

6.1.1 Detailed Description

Contains the Band Location and Resistor Body Identification modules.

6.2 Package valueidentification

6.2.1 Detailed Description

Contains the Colour Mapping and Value Identification Modules

Chapter 7

Class Documentation

7.1 ca.ryanmarks.ohm.imageprocessing.BandReader Class Reference

Module used to analyze the line of pixels selected by the user through the UI. It uses high values of the differential of the RGB colours to detect edges of bands.

Static Public Member Functions

- static List< Point > [read](#) (Mat frame, Point p1, Point p2)

7.1.1 Detailed Description

Module used to analyze the line of pixels selected by the user through the UI. It uses high values of the differential of the RGB colours to detect edges of bands.

Author

Ryan Marks & Jonathan Brown

7.1.2 Member Function Documentation

7.1.2.1 read()

```
static List<Point> ca.ryanmarks.ohm.imageprocessing.BandReader.read (
    Mat frame,
    Point p1,
    Point p2 ) [static]
```

Parameters

<i>frame</i>	Image to Sample.
<i>p1</i>	Starting point of the sampling line.
<i>p2</i>	Ending point of the sampling line.

Returns

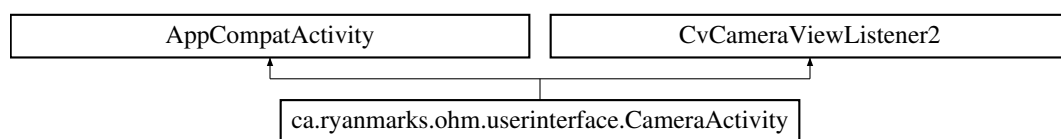
List of points along the line that are likely band edges.

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

- `ca/ryanmarks/ohm/imageprocessing/BandReader.java`

7.2 `ca.ryanmarks.ohm.userinterface.CameraActivity` Class Reference

Inheritance diagram for `ca.ryanmarks.ohm.userinterface.CameraActivity`:

**Public Member Functions**

- void `onCreate` (Bundle savedInstanceState)
- void `onPause` ()
- void `onResume` ()
- void `onDestroy` ()
- void `onCameraViewStarted` (int width, int height)
- void `onCameraViewStopped` ()
- Mat `onCameraFrame` (CameraBridgeViewBase.CvCameraViewFrame inputFrame)

7.2.1 Member Function Documentation

7.2.1.1 `onCameraFrame()`

```
Mat ca.ryanmarks.ohm.userinterface.CameraActivity.onCameraFrame (
    CameraBridgeViewBase.CvCameraViewFrame inputFrame )
```

This method performs the transformation from the acquired camera frame to the image that is shown to the user. It identifies resistances and overlays the resistance value

Parameters

<i>inputFrame</i>	The camera frame captured
-------------------	---------------------------

Returns

The image to be shown to the user

7.2.1.2 onCameraViewStarted()

```
void ca.ryanmarks.ohm.userinterface.CameraActivity.onCameraViewStarted (
    int width,
    int height )
```

Unneeded interface method from openCV

Parameters

<i>width</i>	- the width of the frames that will be delivered
<i>height</i>	- the height of the frames that will be delivered

7.2.1.3 onCameraViewStopped()

```
void ca.ryanmarks.ohm.userinterface.CameraActivity.onCameraViewStopped ( )
```

Unneeded interface method from openCV

7.2.1.4 onCreate()

```
void ca.ryanmarks.ohm.userinterface.CameraActivity.onCreate (
    Bundle savedInstanceState )
```

Called when the activity is first created by Android.

7.2.1.5 onPause()

```
void ca.ryanmarks.ohm.userinterface.CameraActivity.onPause ( )
```

Disable camera acquisition while the application is paused

7.2.1.6 onResume()

```
void ca.ryanmarks.ohm.userinterface.CameraActivity.onResume ( )
```

Whenever the application resumes ensures opencv is loaded.

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

- ca/ryanmarks/ohm/userinterface/CameraActivity.java

7.3 ca.ryanmarks.ohm.Pair< T, T1 > Class Template Reference

This class represents a generic key, value pair.

Public Member Functions

- **Pair** (T key, T1 val)
- T **getKey** ()
- T1 **getValue** ()

7.3.1 Detailed Description

This class represents a generic key, value pair.

Author

Ryan Marks

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

- ca/ryanmarks/ohm/Pair.java

7.4 ca.ryanmarks.ohm.valueidentification.ResistorColour Enum Reference

Enum containing all of the possible colours that a resistor can take on. Also features member functions used to map the colours of bands to values used in the calculation process.

Public Member Functions

- [ResistorColour](#) (int v)

Static Public Member Functions

- static void **trainNN** (Reader trainingData)
- static int [fit](#) (float r, float g, float b)
- static int [fit](#) (int r, int g, int b, int colorSpace)

Public Attributes

- **BLACK** =(0)
- **BROWN** =(1)
- **RED** =(2)
- **ORANGE** =(3)
- **YELLOW** =(4)
- **GREEN** =(5)
- **BLUE** =(6)
- **VIOLET** =(7)
- **GREY** =(8)
- **WHITE** =(9)
- **GOLD** =(11)
- **BASE** =(10)
- int **value**

Static Public Attributes

- static KNearest **KNN**
- static DTrees **dt**

7.4.1 Detailed Description

Enum containing all of the possible colours that a resistor can take on. Also features member functions used to map the colours of bands to values used in the calculation process.

Author

Jonathan Brown

7.4.2 Constructor & Destructor Documentation

7.4.2.1 ResistorColour()

```
ca.ryanmarks.ohm.valueidentification.ResistorColour.ResistorColour (
    int v )
```

Parameters

v	The number represented by the colour in the calculation of the resistor's ohmage.
----------	---

7.4.3 Member Function Documentation

7.4.3.1 fit() [1/2]

```
static int ca.ryanmarks.ohm.valueidentification.ResistorColour.fit (
    float r,
    float g,
    float b ) [static]
```

Function takes in a sampled colour from the images and attempts to fit it to the closest known colour a resistor can possess.

Parameters

r	The red colour value of the colour to be fit.
g	The green colour value of the colour to be fit.
b	The blue colour value of the colour to be fit.

Returns

The known colour that best represents the sampled colour.

7.4.3.2 `fit()` [2/2]

```
static int ca.ryanmarks.ohm.valueidentification.ResistorColour.fit (
    int r,
    int g,
    int b,
    int colorSpace ) [static]
```

Function takes in a sampled colour from the images and attempts to fit it to the closest known colour a resistor can possess. Function takes in a sampled colour from the images and attempts to fitOld it to the closest known colour a resistor can possess.

Parameters

<i>r</i>	The red colour value of the colour to be fitOld.
<i>g</i>	The green colour value of the colour to be fitOld.
<i>b</i>	The blue colour value of the colour to be fitOld.

Returns

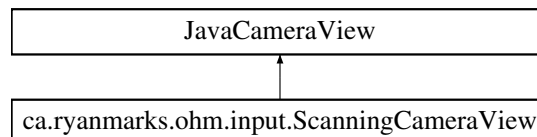
The known colour that best represents the sampled colour.

The documentation for this enum was generated from the following file:

- `ca/ryanmarks/ohm/valueidentification/ResistorColour.java`

7.5 `ca.ryanmarks.ohm.input.ScanningCameraView` Class Reference

Inheritance diagram for `ca.ryanmarks.ohm.input.ScanningCameraView`:



Public Member Functions

- **ScanningCameraView** (Context context, int cameraId)
- **ScanningCameraView** (Context context, AttributeSet attrs)

Protected Member Functions

- boolean [initializeCamera](#) (int width, int height)

7.5.1 Member Function Documentation

7.5.1.1 `initializeCamera()`

```
boolean ca.ryanmarks.ohm.input.ScanningCameraView.initializeCamera (
    int width,
    int height ) [protected]
```

Initialize a camera with the preferred parameters for resistor scanning

Parameters

<i>width</i>	the width of the pictures, in pixels
<i>height</i>	the height of the pictures, in pixels

Returns

The success of the initialization

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

- ca/ryanmarks/ohm/input/ScanningCameraView.java

7.6 ca.ryanmarks.ohm.valueidentification.ValueCalculator Class Reference

Object used to calculate the resistance of the resistor based on the mapped colours.

Public Member Functions

- **ValueCalculator** (Integer a, Integer b, Integer c, Integer d)
- String **getValue** ()

7.6.1 Detailed Description

Object used to calculate the resistance of the resistor based on the mapped colours.

Author

Jonathan Brown

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

- ca/ryanmarks/ohm/valueidentification/ValueCalculator.java

Index

BandLocation, 13

ca.ryanmarks.ohm.imageprocessing.BandReader, 17

ca.ryanmarks.ohm.input.ScanningCameraView, 22

ca.ryanmarks.ohm.Pair< T, T1 >, 19

ca.ryanmarks.ohm.userinterface.CameraActivity, 18

ca.ryanmarks.ohm.valueidentification.ResistorColour, 20

ca.ryanmarks.ohm.valueidentification.ValueCalculator, 23

ca::ryanmarks::ohm::imageprocessing::BandReader
read, 17

ca::ryanmarks::ohm::input::ScanningCameraView
initializeCamera, 22

ca::ryanmarks::ohm::userinterface::CameraActivity

onCameraFrame, 18

onCameraViewStarted, 18

onCameraViewStopped, 19

onCreate, 19

onPause, 19

onResume, 19

ca::ryanmarks::ohm::valueidentification::ResistorColour
fit, 21

ResistorColour, 21

CameraInput, 9

ColourMapping, 11

fit

ca::ryanmarks::ohm::valueidentification::ResistorColour, 21

imageprocessing, 15

initializeCamera

ca::ryanmarks::ohm::input::ScanningCameraView, 22

onCameraFrame

ca::ryanmarks::ohm::userinterface::CameraActivity, 18

onCameraViewStarted

ca::ryanmarks::ohm::userinterface::CameraActivity, 18

onCameraViewStopped

ca::ryanmarks::ohm::userinterface::CameraActivity, 19

onCreate

ca::ryanmarks::ohm::userinterface::CameraActivity, 19

onPause

ca::ryanmarks::ohm::userinterface::CameraActivity, 19

onResume

ca::ryanmarks::ohm::userinterface::CameraActivity, 19

read

ca::ryanmarks::ohm::imageprocessing::BandReader, 17

ResistorColour

ca::ryanmarks::ohm::valueidentification::ResistorColour, 21

UserInterface, 10

ValueCalculator, 12

valueidentification, 15