

RV\_P01

Generated by Doxygen 1.8.10

Sat Oct 15 2016 20:30:42



# Contents

<b>1</b>	<b>Namespace Index</b>	<b>1</b>
1.1	Namespace List	1
<b>2</b>	<b>Class Index</b>	<b>3</b>
2.1	Class List	3
<b>3</b>	<b>File Index</b>	<b>5</b>
3.1	File List	5
<b>4</b>	<b>Namespace Documentation</b>	<b>7</b>
4.1	app Namespace Reference	7
4.1.1	Function Documentation	8
4.1.1.1	fixedMedianParam(int param) O3_NOEXCEPT	8
4.1.1.2	getGx(O3_IN Iti::channel8 const &img, int y, int x)	8
4.1.1.3	getGy(O3_IN Iti::channel8 const &img, int y, int x)	8
4.1.1.4	getTargetColor(O3_IN Iti::channel8 const &sPic, int const pointY, int const pointX, int const my, int const mx, int const med)	8
4.1.1.5	isEven(Numeric numeric) O3_NOEXCEPT	8
4.1.1.6	isOdd(Numeric numeric) O3_NOEXCEPT	9
4.1.1.7	keepWindowOpen(O3_IN std::string const &prompt=""Hit enter to exit"") O3_NOEXCEPT	9
4.1.1.8	median(O3_IN Iti::channel8 const &srcPic, O3_OUT Iti::channel8 &dstPic, Mask mask, Strategy strat=(Strategy()))	9
4.1.1.9	medianImpl(O3_IN Iti::channel8 const &srcPic, O3_OUT Iti::channel8 &dstPic, Mask mask, Naive strat)	9
4.1.1.10	medianImpl(O3_IN Iti::channel8 const &srcPic, O3_OUT Iti::channel8 &dstPic, Mask mask, Fast strat)	9
4.1.1.11	parseCIParams(int argc, O3_IN char **argv)	9
4.1.1.12	resizeImage(O3_OUT Iti::channel8 &toResize, int rows, int columns)	9
4.1.1.13	sobelImpl(O3_IN Iti::channel8 const &sPic, O3_OUT Iti::channel8 &gradientPic, O3_OUT Iti::channel8 &directionPic)	9
4.1.1.14	square(double val)	10
4.2	app::Filter Namespace Reference	10
4.2.1	Detailed Description	10

4.2.2	Enumeration Type Documentation	10
4.2.2.1	Filter	10
<b>5</b>	<b>Class Documentation</b>	<b>11</b>
5.1	app::Action Struct Reference	11
5.1.1	Detailed Description	11
5.1.2	Member Data Documentation	11
5.1.2.1	file	11
5.1.2.2	filter	11
5.1.2.3	maskX	12
5.1.2.4	maskY	12
5.2	app::CoutFlusher Struct Reference	12
5.2.1	Detailed Description	12
5.2.2	Member Function Documentation	12
5.2.2.1	operator>() const O3_NOEXCEPT	12
5.3	app::Fast Struct Reference	12
5.3.1	Detailed Description	13
5.4	app::Mask Struct Reference	13
5.4.1	Detailed Description	13
5.4.2	Constructor & Destructor Documentation	13
5.4.2.1	Mask(int x_, int y_) O3_NOEXCEPT	13
5.4.3	Member Data Documentation	13
5.4.3.1	x	13
5.4.3.2	y	14
5.5	app::Naive Struct Reference	14
5.5.1	Detailed Description	14
5.6	app::RvP01 Class Reference	14
5.6.1	Detailed Description	14
5.6.2	Member Function Documentation	14
5.6.2.1	Median(O3_IN Iti::channel8 const &sPic, O3_OUT Iti::channel8 &dPic, int const MaskSizeX, int const MaskSizeY)	14
5.6.2.2	operator()(int argc, O3_IN char **argv)	15
5.6.2.3	Sobel(O3_IN Iti::channel8 const &sPic, O3_OUT Iti::channel8 &GradientPic, O3_OUT Iti::channel8 &DirectionPic)	15
<b>6</b>	<b>File Documentation</b>	<b>17</b>
6.1	C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/CiParams.cpp File Reference	17
6.2	C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/CiParams.h File Reference	17
6.3	C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/main.cpp File Reference	18
6.3.1	Function Documentation	18

6.3.1.1	<a href="#">main(int argc, char *argv[])</a>	18
6.4	<a href="#">C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/Median.cpp File Reference</a>	18
6.5	<a href="#">C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/Median.h File Reference</a>	19
6.6	<a href="#">C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/RV_P01.cpp File Reference</a>	19
6.7	<a href="#">C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/RV_P01.h File Reference</a>	20
6.8	<a href="#">C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/Sobel.cpp File Reference</a>	20
6.9	<a href="#">C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/Sobel.h File Reference</a>	21
6.10	<a href="#">C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/utils.cpp File Reference</a>	21
6.10.1	<a href="#">Macro Definition Documentation</a>	22
6.10.1.1	<a href="#">max</a>	22
6.11	<a href="#">C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/utils.h File Reference</a>	22
6.11.1	<a href="#">Macro Definition Documentation</a>	22
6.11.1.1	<a href="#">MAX_GRAYSCALE</a>	22
6.11.1.2	<a href="#">MIN_GRAYSCALE</a>	23
<b>Index</b>		<b>25</b>



# Chapter 1

## Namespace Index

### 1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

<a href="#">app</a> . . . . .	7
<a href="#">app::Filter</a>	
Wrapping namespace to simulate C++11 scoped enums . . . . .	10





## Chapter 2

# Class Index

### 2.1 Class List

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

<a href="#">app::Action</a>	Struct that represents what actions are to be taken . . . . .	11
<a href="#">app::CoutFlusher</a>	Functor that flushes cout in its call operator . . . . .	12
<a href="#">app::Fast</a>	Type to represent the fast histogram based implementation of the median filter . . . . .	12
<a href="#">app::Mask</a>	<a href="#">Mask</a> struct for the median filter . . . . .	13
<a href="#">app::Naive</a>	Type to represent the 'naive', slow implementation of the median filter . . . . .	14
<a href="#">app::RvP01</a>	Class to be instantiated in the main function, calling the call operator will effectively launch the application . . . . .	14



## Chapter 3

# File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/CIParams.cpp . . . . .	17
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/CIParams.h . . . . .	17
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/main.cpp . . . . .	18
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/Median.cpp . . . . .	18
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/Median.h . . . . .	19
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/RV_P01.cpp . . . . .	19
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/RV_P01.h . . . . .	20
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/Sobel.cpp . . . . .	20
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/Sobel.h . . . . .	21
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/utils.cpp . . . . .	21
C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/utils.h . . . . .	22



## Chapter 4

# Namespace Documentation

### 4.1 app Namespace Reference

#### Namespaces

- [Filter](#)

*wrapping namespace to simulate C++11 scoped enums*

#### Classes

- struct [Action](#)

*struct that represents what actions are to be taken*

- struct [CoutFlusher](#)

*functor that flushes cout in its call operator*

- struct [Fast](#)

*type to represent the fast histogram based implementation of the median filter*

- struct [Mask](#)

*[Mask](#) struct for the median filter.*

- struct [Naive](#)

*type to represent the 'naive', slow implementation of the median filter*

- class [RvP01](#)

*class to be instantiated in the main function, calling the call operator will effectively launch the application*

#### Functions

- [Action parseCIParams](#) (int argc, O3\_IN char \*\*argv)

*parses the command line parameters given and produces an appropriate [Action](#) struct*

- void [medianImpl](#) (O3\_IN lti::channel8 const &srcPic, O3\_OUT lti::channel8 &dstPic, [Mask](#) mask, [Naive](#) strat)

*'naive' implementation of the median filter*

- O3\_FORCE\_INLINE int [getTargetColor](#) (O3\_IN lti::channel8 const &sPic, int const pointY, int const pointX, int const my, int const mx, int const med)

*returns the color to write at pointY, pointX; used by histogram median*

- void [medianImpl](#) (O3\_IN lti::channel8 const &srcPic, O3\_OUT lti::channel8 &dstPic, [Mask](#) mask, [Fast](#) strat)

- template<typename Strategy >

O3\_FORCE\_INLINE void [median](#) (O3\_IN lti::channel8 const &srcPic, O3\_OUT lti::channel8 &dstPic, [Mask](#) mask, Strategy strat=(Strategy()))

- O3\_FORCE\_INLINE double [square](#) (double val)

- calculates the square of the value passed into the parameter and returns the result.*
- O3\_FORCE\_INLINE int [getGx](#) (O3\_IN Iti::channel8 const &img, int y, int x)
- O3\_FORCE\_INLINE int [getGy](#) (O3\_IN Iti::channel8 const &img, int y, int x)
- void [sobelImpl](#) (O3\_IN Iti::channel8 const &sPic, O3\_OUT Iti::channel8 &gradientPic, O3\_OUT Iti::channel8 &directionPic)
- implementation function of the sobel filter, called by [RvP01::Sobel](#)*
- void [keepWindowOpen](#) (O3\_IN std::string const &prompt="Hit enter to exit") O3\_NOEXCEPT
- keeps the console window open, waiting for the user to hit enter, displaying the prompt passed into the parameter*
- void [resizeImage](#) (O3\_OUT Iti::channel8 &toResize, int rows, int columns)
- resizes the image passed into toResize to have rows rows and columns columns*
- template<typename Numeric >  
O3\_FORCE\_INLINE bool [isOdd](#) (Numeric numeric) O3\_NOEXCEPT
- returns true if the value passed into the parameter is odd; false otherwise*
- template<typename Numeric >  
O3\_FORCE\_INLINE bool [isEven](#) (Numeric numeric) O3\_NOEXCEPT
- return true if the value passed into the parameter is even; false otherwise*
- O3\_FORCE\_INLINE int [fixedMedianParam](#) (int param) O3\_NOEXCEPT

## 4.1.1 Function Documentation

### 4.1.1.1 O3\_FORCE\_INLINE int app::fixedMedianParam ( int param )

fixes a median mask parameter, by returning the value one larger than the value passed into the parameter if it is even; otherwise returns the value passed into the parameter.

Definition at line 37 of file utils.h.

### 4.1.1.2 O3\_FORCE\_INLINE int app::getGx ( O3\_IN Iti::channel8 const &img, int y, int x )

returns Gx for the pixel at position y, x in the image img uses the following filter mask: 

-----	-1	0	1
1		0	2
		-1	0

Definition at line 21 of file Sobel.cpp.

### 4.1.1.3 O3\_FORCE\_INLINE int app::getGy ( O3\_IN Iti::channel8 const &img, int y, int x )

return Gy for the pixel at position y, x in the image img uses the following filter mask: 

-----	-1	-2	-1
	0	0	0
1			2

Definition at line 38 of file Sobel.cpp.

### 4.1.1.4 O3\_FORCE\_INLINE int app::getTargetColor ( O3\_IN Iti::channel8 const &sPic, int const pointY, int const pointX, int const my, int const mx, int const med )

returns the color to write at pointY, pointX; used by histogram median

Definition at line 64 of file Median.cpp.

### 4.1.1.5 template<typename Numeric > O3\_FORCE\_INLINE bool app::isEven ( Numeric numeric )

return true if the value passed into the parameter is even; false otherwise

Definition at line 28 of file utils.h.

**4.1.1.6** `template<typename Numeric > O3_FORCE_INLINE bool app::isOdd ( Numeric numeric )`

returns true if the value passed into the parameter is odd; false otherwise

Definition at line 20 of file `utils.h`.

**4.1.1.7** `void app::keepWindowOpen ( O3_IN std::string const & prompt )`

keeps the console window open, waiting for the user to hit enter, displaying the prompt passed into the parameter

Definition at line 10 of file `utils.cpp`.

**4.1.1.8** `template<typename Strategy > O3_FORCE_INLINE void app::median ( O3_IN Iti::channel8 const & srcPic, O3_OUT Iti::channel8 & dstPic, Mask mask, Strategy strat = (Strategy()) )`

wrapper function that delegates to the appropriate `medianImpl` function based on the strategy to be used, called by [RvP01::Median](#)

Definition at line 32 of file `Median.h`.

**4.1.1.9** `void app::medianImpl ( O3_IN Iti::channel8 const & srcPic, O3_OUT Iti::channel8 & dstPic, Mask mask, Naive strat )`

'naive' implementation of the median filter

Definition at line 9 of file `Median.cpp`.

**4.1.1.10** `void app::medianImpl ( O3_IN Iti::channel8 const & srcPic, O3_OUT Iti::channel8 & dstPic, Mask mask, Fast strat )`

fast histogram based implementation of the median filter, approximately two times faster than the 'naive' implementation

Definition at line 90 of file `Median.cpp`.

**4.1.1.11** `Action app::parseCIParams ( int argc, O3_IN char ** argv )`

parses the command line parameters given and produces an appropriate [Action](#) struct stringstream that starts of with the string in `argv[maskXOffset]`

Definition at line 7 of file `CIParams.cpp`.

**4.1.1.12** `void app::resizeImage ( O3_OUT Iti::channel8 & toResize, int rows, int columns )`

resizes the image passed into `toResize` to have rows rows and columns columns

Definition at line 22 of file `utils.cpp`.

**4.1.1.13** `void app::sobelImpl ( O3_IN Iti::channel8 const & sPic, O3_OUT Iti::channel8 & gradientPic, O3_OUT Iti::channel8 & directionPic )`

implementation function of the sobel filter, called by [RvP01::Sobel](#)

Definition at line 47 of file `Sobel.cpp`.

#### 4.1.1.14 O3\_FORCE\_INLINE double app::square ( double *val* )

calculates the square of the value passed into the parameter and returns the result.

Definition at line 9 of file Sobel.cpp.

## 4.2 app::Filter Namespace Reference

wrapping namespace to simulate C++11 scoped enums

### Enumerations

- enum [Filter](#) { [Median](#), [Sobel](#), [Error](#) }  
*enum for the different type of filters*

#### 4.2.1 Detailed Description

wrapping namespace to simulate C++11 scoped enums

#### 4.2.2 Enumeration Type Documentation

##### 4.2.2.1 enum app::Filter::Filter

enum for the different type of filters

#### Enumerator

***Median*** represents the median filter

***Sobel*** represents the sobel filter

***Error*** used to indicate that the filter to use could not be determined from the command line parameters

Definition at line 10 of file CIParams.h.



## Chapter 5

# Class Documentation

### 5.1 app::Action Struct Reference

struct that represents what actions are to be taken

```
#include <ClParams.h>
```

#### Public Attributes

- [Filter::Filter filter](#)  
*the filter to use on the image*
- `std::string` [file](#)  
*path to the image*
- `int` [maskX](#)  
*mask x size for median filter*
- `int` [maskY](#)  
*mask y size for median filter*

#### 5.1.1 Detailed Description

struct that represents what actions are to be taken

Definition at line 23 of file ClParams.h.

#### 5.1.2 Member Data Documentation

##### 5.1.2.1 `std::string` app::Action::file

path to the image

Definition at line 28 of file ClParams.h.

##### 5.1.2.2 `Filter::Filter` app::Action::filter

the filter to use on the image

Definition at line 25 of file ClParams.h.

### 5.1.2.3 int app::Action::maskX

mask x size for median filter

Definition at line 31 of file CIParams.h.

### 5.1.2.4 int app::Action::maskY

mask y size for median filter

Definition at line 34 of file CIParams.h.

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

- C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/CIParams.h

## 5.2 app::CoutFlusher Struct Reference

functor that flushes cout in its call operator

```
#include <utils.h>
```

### Public Member Functions

- void [operator\(\)](#) () const O3\_NOEXCEPT  
*the [CoutFlusher](#) structs call operator; will flush cout when called.*

### 5.2.1 Detailed Description

functor that flushes cout in its call operator

Definition at line 43 of file utils.h.

### 5.2.2 Member Function Documentation

#### 5.2.2.1 void app::CoutFlusher::operator() ( ) const

the [CoutFlusher](#) structs call operator; will flush cout when called.

Definition at line 18 of file utils.cpp.

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

- C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/[utils.h](#)
- C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/[utils.cpp](#)

## 5.3 app::Fast Struct Reference

type to represent the fast histogram based implementation of the median filter

```
#include <Median.h>
```

### 5.3.1 Detailed Description

type to represent the fast histogram based implementation of the median filter

Definition at line 25 of file Median.h.

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

- C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/[Median.h](#)

## 5.4 app::Mask Struct Reference

[Mask](#) struct for the median filter.

```
#include <Median.h>
```

### Public Member Functions

- `O3_FORCE_INLINE Mask (int x_, int y_) O3_NOEXCEPT`  
*constructs a [Mask](#) from two integers*

### Public Attributes

- `int x`  
*mask x size*
- `int y`  
*mask y size*

### 5.4.1 Detailed Description

[Mask](#) struct for the median filter.

Definition at line 8 of file Median.h.

### 5.4.2 Constructor & Destructor Documentation

#### 5.4.2.1 `O3_FORCE_INLINE app::Mask::Mask ( int x_, int y_ ) [inline]`

constructs a [Mask](#) from two integers

Definition at line 10 of file Median.h.

### 5.4.3 Member Data Documentation

#### 5.4.3.1 `int app::Mask::x`

mask x size

Definition at line 15 of file Median.h.

### 5.4.3.2 int app::Mask::y

mask y size

Definition at line 18 of file Median.h.

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

- C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/[Median.h](#)

## 5.5 app::Naive Struct Reference

type to represent the 'naive', slow implementation of the median filter

```
#include <Median.h>
```

### 5.5.1 Detailed Description

type to represent the 'naive', slow implementation of the median filter

Definition at line 22 of file Median.h.

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

- C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/[Median.h](#)

## 5.6 app::RvP01 Class Reference

class to be instantiated in the main function, calling the call operator will effectively launch the application

```
#include <RV_P01.h>
```

### Public Member Functions

- void [operator\(\)](#) (int argc, O3\_IN char \*\*argv)  
*the call operator, to be called in the main function, delegate the argc and argv parameters to this member function*
- void [Median](#) (O3\_IN Iti::channel8 const &sPic, O3\_OUT Iti::channel8 &dPic, int const MaskSizeX, int const MaskSizeY)
- void [Sobel](#) (O3\_IN Iti::channel8 const &sPic, O3\_OUT Iti::channel8 &GradientPic, O3\_OUT Iti::channel8 &DirectionPic)

### 5.6.1 Detailed Description

class to be instantiated in the main function, calling the call operator will effectively launch the application

Definition at line 8 of file RV\_P01.h.

### 5.6.2 Member Function Documentation

**5.6.2.1** void app::RvP01::Median ( O3\_IN Iti::channel8 const & *sPic*, O3\_OUT Iti::channel8 & *dPic*, int const *MaskSizeX*, int const *MaskSizeY* )

applies median filter to the sPic passed using the median mask size of MaskSizeX and MaskSizeY, writes the result to dPic

Definition at line 82 of file RV\_P01.cpp.

### 5.6.2.2 void app::RvP01::operator() ( int *argc*, O3\_IN char \*\* *argv* )

the call operator, to be called in the main function, delegate the argc and argv parameters to this member function  
Definition at line 14 of file RV\_P01.cpp.

### 5.6.2.3 void app::RvP01::Sobel ( O3\_IN Iti::channel8 const & *sPic*, O3\_OUT Iti::channel8 & *GradientPic*, O3\_OUT Iti::channel8 & *DirectionPic* )

applies sobel filter to sPic passed, outputs the gradient picture to GradientPic and the direction picture to DirectionPic

Definition at line 96 of file RV\_P01.cpp.

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

- C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/[RV\\_P01.h](#)
- C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/[RV\\_P01.cpp](#)



## Chapter 6

# File Documentation

### 6.1 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/CIParams.cpp File Reference

```
#include <sstream>
#include "O3/macros.h"
#include "utils.h"
#include "ClParams.h"
```

#### Namespaces

- [app](#)

#### Functions

- Action [app::parseCIParams](#) (int argc, O3\_IN char \*\*argv)  
*parses the command line parameters given and produces an appropriate [Action](#) struct*

### 6.2 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/CIParams.h File Reference

```
#include <string>
#include "O3/macros.h"
```

#### Classes

- struct [app::Action](#)  
*struct that represents what actions are to be taken*

#### Namespaces

- [app](#)
- [app::Filter](#)  
*wrapping namespace to simulate C++11 scoped enums*

## Enumerations

- enum [app::Filter::Filter](#) { [app::Filter::Median](#), [app::Filter::Sobel](#), [app::Filter::Error](#) }  
*enum for the different type of filters*

## Functions

- Action [app::parseCIParams](#) (int argc, O3\_IN char \*\*argv)  
*parses the command line parameters given and produces an appropriate [Action](#) struct*

## 6.3 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/main.cpp File Reference

```
#include <iostream>
#include <cstdlib>
#include <ltiException.h>
#include "O3/except.h"
#include "O3/final_act.h"
#include "RV_P01.h"
#include "utils.h"
```

## Functions

- int [main](#) (int argc, char \*argv[ ])  
*main function; the entry point of the application*

### 6.3.1 Function Documentation

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

main function; the entry point of the application

Definition at line 10 of file main.cpp.

## 6.4 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/Median.cpp File Reference

```
#include <algorithm>
#include "O3/memory.h"
#include "O3/containers.h"
#include "O3/macros.h"
#include "utils.h"
#include "Median.h"
```

## Namespaces

- [app](#)



## Functions

- void [app::medianImpl](#) (O3\_IN lti::channel8 const &srcPic, O3\_OUT lti::channel8 &dstPic, Mask mask, Naive strat)  
*'naive' implementation of the median filter*
- O3\_FORCE\_INLINE int [app::getTargetColor](#) (O3\_IN lti::channel8 const &sPic, int const pointY, int const pointX, int const my, int const mx, int const med)  
*returns the color to write at pointY, pointX; used by histogram median*
- void [app::medianImpl](#) (O3\_IN lti::channel8 const &srcPic, O3\_OUT lti::channel8 &dstPic, Mask mask, Fast strat)

## 6.5 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/Median.h File Reference

```
#include <ltiImage.h>
#include "O3/macros.h"
```

## Classes

- struct [app::Mask](#)  
*Mask struct for the median filter.*
- struct [app::Naive](#)  
*type to represent the 'naive', slow implementation of the median filter*
- struct [app::Fast](#)  
*type te represent the fast histogram based implementation of the median filter*

## Namespaces

- [app](#)

## Functions

- template<typename Strategy >  
O3\_FORCE\_INLINE void [app::median](#) (O3\_IN lti::channel8 const &srcPic, O3\_OUT lti::channel8 &dstPic, Mask mask, Strategy strat=(Strategy()))
- void [app::medianImpl](#) (O3\_IN lti::channel8 const &srcPic, O3\_OUT lti::channel8 &dstPic, Mask mask, Naive strat)  
*'naive' implementation of the median filter*
- void [app::medianImpl](#) (O3\_IN lti::channel8 const &srcPic, O3\_OUT lti::channel8 &dstPic, Mask mask, Fast strat)

## 6.6 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/RV\_P01.cpp File Reference

```
#include <cstdlib>
```

```
#include <ltiViewer.h>
#include <ltiBMPFunctor.h>
#include <ltiSplitImageToHSI.h>
#include <gtk.h>
#include <ltiGtkServer.h>
#include "utils.h"
#include "Median.h"
#include "Sobel.h"
#include "ClParams.h"
#include "RV_P01.h"
```

## Namespaces

- [app](#)

## 6.7 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/RV\_P01.h File Reference

```
#include <ltiImage.h>
#include "O3/macros.h"
```

## Classes

- class [app::RvP01](#)

*class to be instantiated in the main function, calling the call operator will effectively launch the application*

## Namespaces

- [app](#)

## 6.8 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/Sobel.cpp File Reference

```
#include <cmath>
#include "O3/macros.h"
#include "O3/algorithm.h"
#include "utils.h"
#include "Sobel.h"
```

## Namespaces

- [app](#)

## Functions

- O3\_FORCE\_INLINE double [app::square](#) (double val)

*calculates the square of the value passed into the parameter and returns the result.*

- O3\_FORCE\_INLINE int [app::getGx](#) (O3\_IN lti::channel8 const &img, int y, int x)
- O3\_FORCE\_INLINE int [app::getGy](#) (O3\_IN lti::channel8 const &img, int y, int x)
- void [app::sobelImpl](#) (O3\_IN lti::channel8 const &sPic, O3\_OUT lti::channel8 &gradientPic, O3\_OUT lti::channel8 &directionPic)

*implementation function of the sobel filter, called by [RvP01::Sobel](#)*

## 6.9 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/Sobel.h File Reference

```
#include <ltiImage.h>
```

### Namespaces

- [app](#)

### Functions

- void [app::sobelImpl](#) (O3\_IN lti::channel8 const &sPic, O3\_OUT lti::channel8 &gradientPic, O3\_OUT lti::channel8 &directionPic)

*implementation function of the sobel filter, called by [RvP01::Sobel](#)*

## 6.10 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/utils.cpp File Reference

```
#include <iostream>
#include <limits>
#include "utils.h"
```

### Namespaces

- [app](#)

### Macros

- `#define max(a, b) (((a) > (b)) ? (a) : (b))`

### Functions

- void [app::keepWindowOpen](#) (O3\_IN std::string const &prompt="Hit enter to exit") O3\_NOEXCEPT  
*keeps the console window open, waiting for the user to hit enter, displaying the prompt passed into the parameter*
- void [app::resizeImage](#) (O3\_OUT lti::channel8 &toResize, int rows, int columns)  
*resizes the image passed into toResize to have rows rows and columns columns*

## 6.10.1 Macro Definition Documentation

### 6.10.1.1 `#define max( a, b ) (((a) > (b)) ? (a) : (b))`

Definition at line 35 of file `utils.cpp`.

## 6.11 C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV\_P01/RV\_P01/utils.h File Reference

```
#include <string>
#include <cstdint>
#include <ltiImage.h>
#include "O3/macros.h"
```

### Classes

- struct [app::CoutFlusher](#)  
*functor that flushes cout in its call operator*

### Namespaces

- [app](#)

### Macros

- `#define MIN_GRAYSCALE 0`  
*minimum possible grayscale value: 0*
- `#define MAX_GRAYSCALE 255`  
*maximum possible grayscale value: 255*

### Functions

- void [app::keepWindowOpen](#) (O3\_IN std::string const &prompt="Hit enter to exit") O3\_NOEXCEPT  
*keeps the console window open, waiting for the user to hit enter, displaying the prompt passed into the parameter*
- template<typename Numeric >  
O3\_FORCE\_INLINE bool [app::isOdd](#) (Numeric numeric) O3\_NOEXCEPT  
*returns true if the value passed into the parameter is odd; false otherwise*
- template<typename Numeric >  
O3\_FORCE\_INLINE bool [app::isEven](#) (Numeric numeric) O3\_NOEXCEPT  
*return true if the value passed into the parameter is even; false otherwise*
- O3\_FORCE\_INLINE int [app::fixedMedianParam](#) (int param) O3\_NOEXCEPT
- void [app::resizeImage](#) (O3\_OUT lti::channel8 &toResize, int rows, int columns)  
*resizes the image passed into toResize to have rows rows and columns columns*

## 6.11.1 Macro Definition Documentation

### 6.11.1.1 `#define MAX_GRAYSCALE 255`

maximum possible grayscale value: 255

Definition at line 12 of file `utils.h`.

#### 6.11.1.2 #define MIN\_GRAYSCALE 0

minimun possible grayscale value: 0

Definition at line 9 of file utls.h.



# Index

app, [7](#)  
    fixedMedianParam, [8](#)  
    getGx, [8](#)  
    getGy, [8](#)  
    getTargetColor, [8](#)  
    isEven, [8](#)  
    isOdd, [8](#)  
    keepWindowOpen, [9](#)  
    median, [9](#)  
    medianImpl, [9](#)  
    parseCIParams, [9](#)  
    resizeImage, [9](#)  
    sobelImpl, [9](#)  
    square, [9](#)  
app::Action, [11](#)  
    file, [11](#)  
    filter, [11](#)  
    maskX, [11](#)  
    maskY, [12](#)  
app::CoutFlusher, [12](#)  
    operator(), [12](#)  
app::Fast, [12](#)  
app::Filter, [10](#)  
    Error, [10](#)  
    Filter, [10](#)  
    Median, [10](#)  
    Sobel, [10](#)  
app::Mask, [13](#)  
    Mask, [13](#)  
    x, [13](#)  
    y, [13](#)  
app::Naive, [14](#)  
app::RvP01, [14](#)  
    Median, [14](#)  
    operator(), [14](#)  
    Sobel, [15](#)  
  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/CIParams.cpp, [17](#)  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/CIParams.h, [17](#)  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/Median.cpp, [18](#)  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/Median.h, [19](#)  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/RV\_P01.cpp, [19](#)  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/RV\_P01.h, [20](#)  
  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/Sobel.cpp, [20](#)  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/Sobel.h, [21](#)  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/main.cpp, [18](#)  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/utils.cpp, [21](#)  
C:/Users/Jefe/Documents/Visual Studio 2008/↵  
    Projects/RV\_P01/RV\_P01/utils.h, [22](#)  
  
Error  
    app::Filter, [10](#)  
  
file  
    app::Action, [11](#)  
Filter  
    app::Filter, [10](#)  
filter  
    app::Action, [11](#)  
fixedMedianParam  
    app, [8](#)  
  
getGx  
    app, [8](#)  
getGy  
    app, [8](#)  
getTargetColor  
    app, [8](#)  
  
isEven  
    app, [8](#)  
isOdd  
    app, [8](#)  
  
keepWindowOpen  
    app, [9](#)  
  
MAX\_GRAYSCALE  
    utils.h, [22](#)  
MIN\_GRAYSCALE  
    utils.h, [22](#)  
main  
    main.cpp, [18](#)  
main.cpp  
    main, [18](#)  
Mask  
    app::Mask, [13](#)  
maskX  
    app::Action, [11](#)  
maskY

- app::Action, [12](#)
- max
  - utils.cpp, [22](#)
- Median
  - app::Filter, [10](#)
  - app::RvP01, [14](#)
- median
  - app, [9](#)
- medianImpl
  - app, [9](#)
- operator()
  - app::CoutFlusher, [12](#)
  - app::RvP01, [14](#)
- parseCIParams
  - app, [9](#)
- resizeImage
  - app, [9](#)
- Sobel
  - app::Filter, [10](#)
  - app::RvP01, [15](#)
- sobelImpl
  - app, [9](#)
- square
  - app, [9](#)
- utils.cpp
  - max, [22](#)
- utils.h
  - MAX\_GRAYSCALE, [22](#)
  - MIN\_GRAYSCALE, [22](#)
- x
  - app::Mask, [13](#)
- y
  - app::Mask, [13](#)