RV_P01

Generated by Doxygen 1.8.10

Sat Oct 15 2016 20:30:42

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

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 te 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 parseClParams (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 lti::channel8 const &img, int y, int x)
- O3_FORCE_INLINE int getGy (O3_IN lti::channel8 const &img, int y, int x)
- void 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
- 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 lti::channel8 &toResize, int rows, int columns)

  resizes the image passed into toRisize 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 lti::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| |-2 0 2| |-1 0 1| |________|

Definition at line 21 of file Sobel.cpp.

4.1.1.3 O3_FORCE_INLINE int app::getGy (O3_IN lti::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 1| |________|

Definition at line 38 of file Sobel.cpp.

4.1.1.4 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

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 lti::channel8 const & *srcPic,* O3_OUT lti::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 lti::channel8 const & *srcPic,* O3_OUT lti::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 lti::channel8 const & *srcPic,* O3_OUT lti::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::parseClParams (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 ClParams.cpp.

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

resizes the image passed into toRisize to have rows rows and columns columns

Definition at line 22 of file utils.cpp.

4.1.1.13 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

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 ClParams.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 ClParams.h.

5.1.2.4 int app::Action::maskY

mask y size for median filter

Definition at line 34 of file ClParams.h.

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

- C:/Users/Jefe/Documents/Visual Studio 2008/Projects/RV_P01/RV_P01/ClParams.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 te represent the fast histogram based implementation of the median filter

`#include <Median.h>`

5.3.1 Detailed Description

type te 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 lti::channel8 const &sPic, O3_OUT lti::channel8 &dPic, int const MaskSizeX, int const MaskSizeY)
- void Sobel (O3_IN lti::channel8 const &sPic, O3_OUT lti::channel8 &GradientPic, O3_OUT lti::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 lti::channel8 const & *sPic,* O3_OUT lti::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 lti::channel8 const & *sPic*, O3_OUT lti::channel8 & *GradientPic*, O3_OUT lti::channel8 & *DirectionPic*)

applies sobel filter to sPic passed, outputs the gradient picture to GradientPic and the direction picture to Direction↩ Pic

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/ClParams.cpp File Reference

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

Namespaces

- app

Functions

- Action app::parseClParams (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/ClParams.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::parseClParams (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 toRisize 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 <cstddef>
#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

  minimun 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 toRisize 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 utils.h.

Index