

Boxxer

Generated by Doxygen 1.8.11

## Contents

<b>1</b>	<b>Boxxer</b>	<b>2</b>
<b>2</b>	<b>Namespace Index</b>	<b>2</b>
2.1	Namespace List . . . . .	2
<b>3</b>	<b>Hierarchical Index</b>	<b>2</b>
3.1	Class Hierarchy . . . . .	2
<b>4</b>	<b>Class Index</b>	<b>3</b>
4.1	Class List . . . . .	3
<b>5</b>	<b>File Index</b>	<b>3</b>
5.1	File List . . . . .	3
<b>6</b>	<b>Namespace Documentation</b>	<b>4</b>
6.1	boxxer Namespace Reference . . . . .	4
6.1.1	Typedef Documentation . . . . .	5
6.2	boxxer::kernels Namespace Reference . . . . .	5
6.2.1	Detailed Description . . . . .	6
6.2.2	Function Documentation . . . . .	6
<b>7</b>	<b>Class Documentation</b>	<b>14</b>
7.1	boxxer::Boxxer2D< FloatT, IdxT > Class Template Reference . . . . .	14
7.1.1	Detailed Description . . . . .	16
7.1.2	Member Typedef Documentation . . . . .	16
7.1.3	Constructor & Destructor Documentation . . . . .	17
7.1.4	Member Function Documentation . . . . .	17
7.1.5	Member Data Documentation . . . . .	20
7.2	boxxer::Boxxer3D< FloatT, IdxT > Class Template Reference . . . . .	21
7.2.1	Detailed Description . . . . .	22

7.2.2	Member Typedef Documentation . . . . .	22
7.2.3	Constructor & Destructor Documentation . . . . .	23
7.2.4	Member Function Documentation . . . . .	23
7.2.5	Member Data Documentation . . . . .	26
7.3	boxxer::DoGFilter2D< FloatT, IdxT > Class Template Reference . . . . .	27
7.3.1	Detailed Description . . . . .	28
7.3.2	Member Typedef Documentation . . . . .	28
7.3.3	Constructor & Destructor Documentation . . . . .	29
7.3.4	Member Function Documentation . . . . .	29
7.3.5	Friends And Related Function Documentation . . . . .	30
7.3.6	Member Data Documentation . . . . .	30
7.4	boxxer::DoGFilter3D< FloatT, IdxT > Class Template Reference . . . . .	31
7.4.1	Detailed Description . . . . .	32
7.4.2	Member Typedef Documentation . . . . .	33
7.4.3	Constructor & Destructor Documentation . . . . .	33
7.4.4	Member Function Documentation . . . . .	33
7.4.5	Friends And Related Function Documentation . . . . .	34
7.4.6	Member Data Documentation . . . . .	34
7.5	boxxer::GaussFilter2D< FloatT, IdxT > Class Template Reference . . . . .	35
7.5.1	Detailed Description . . . . .	36
7.5.2	Member Typedef Documentation . . . . .	37
7.5.3	Constructor & Destructor Documentation . . . . .	37
7.5.4	Member Function Documentation . . . . .	37
7.5.5	Friends And Related Function Documentation . . . . .	38
7.5.6	Member Data Documentation . . . . .	38
7.6	boxxer::GaussFilter3D< FloatT, IdxT > Class Template Reference . . . . .	39
7.6.1	Detailed Description . . . . .	40
7.6.2	Member Typedef Documentation . . . . .	40

7.6.3	Constructor & Destructor Documentation . . . . .	41
7.6.4	Member Function Documentation . . . . .	41
7.6.5	Friends And Related Function Documentation . . . . .	42
7.6.6	Member Data Documentation . . . . .	42
7.7	boxxer::GaussFIRFilter< FloatT, IdxT > Class Template Reference . . . . .	43
7.7.1	Detailed Description . . . . .	44
7.7.2	Member Typedef Documentation . . . . .	44
7.7.3	Constructor & Destructor Documentation . . . . .	45
7.7.4	Member Function Documentation . . . . .	45
7.7.5	Member Data Documentation . . . . .	45
7.8	boxxer::LoGFilter2D< FloatT, IdxT > Class Template Reference . . . . .	46
7.8.1	Detailed Description . . . . .	47
7.8.2	Member Typedef Documentation . . . . .	48
7.8.3	Constructor & Destructor Documentation . . . . .	48
7.8.4	Member Function Documentation . . . . .	48
7.8.5	Friends And Related Function Documentation . . . . .	49
7.8.6	Member Data Documentation . . . . .	49
7.9	boxxer::LoGFilter3D< FloatT, IdxT > Class Template Reference . . . . .	50
7.9.1	Detailed Description . . . . .	51
7.9.2	Member Typedef Documentation . . . . .	51
7.9.3	Constructor & Destructor Documentation . . . . .	52
7.9.4	Member Function Documentation . . . . .	52
7.9.5	Friends And Related Function Documentation . . . . .	53
7.9.6	Member Data Documentation . . . . .	53
7.10	boxxer::LogicalError Struct Reference . . . . .	54
7.10.1	Detailed Description . . . . .	54
7.10.2	Constructor & Destructor Documentation . . . . .	55
7.11	boxxer::Maxima2D< FloatT, IdxT > Class Template Reference . . . . .	55

7.11.1	Detailed Description	55
7.11.2	Member Typedef Documentation	56
7.11.3	Constructor & Destructor Documentation	56
7.11.4	Member Function Documentation	56
7.11.5	Member Data Documentation	56
7.12	boxxer::Maxima3D< FloatT, IdxT > Class Template Reference	57
7.12.1	Detailed Description	58
7.12.2	Member Typedef Documentation	58
7.12.3	Constructor & Destructor Documentation	59
7.12.4	Member Function Documentation	59
7.12.5	Member Data Documentation	59
7.13	boxxer::NumericalError Struct Reference	60
7.13.1	Detailed Description	60
7.13.2	Constructor & Destructor Documentation	61
7.14	boxxer::ParameterShapeError Struct Reference	61
7.14.1	Detailed Description	62
7.14.2	Constructor & Destructor Documentation	62
7.15	boxxer::ParameterValueError Struct Reference	62
7.15.1	Detailed Description	63
7.15.2	Constructor & Destructor Documentation	63
<b>8</b>	<b>File Documentation</b>	<b>64</b>
8.1	Boxxer2D.h File Reference	64
8.1.1	Detailed Description	64
8.2	Boxxer3D.h File Reference	65
8.2.1	Detailed Description	65
8.3	BoxxerError.h File Reference	66
8.3.1	Detailed Description	66
8.4	FilterKernels.h File Reference	67
8.4.1	Detailed Description	68
8.5	GaussFilter.h File Reference	69
8.5.1	Detailed Description	70
8.6	Maxima.h File Reference	70
8.6.1	Detailed Description	71
8.7	README.md File Reference	71

<a href="#">Index</a>	73
-----------------------	----

## 1 Boxxer

A tool for identifying individual Gaussian emitters for super-resolution microscopy toolchains.

## 2 Namespace Index

### 2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

<a href="#">boxxer</a>	4
<a href="#">boxxer::kernels</a>	5

## 3 Hierarchical Index

### 3.1 Class Hierarchy

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

<a href="#">boxxer::Boxxer2D&lt; FloatT, IdxT &gt;</a>	14
<a href="#">boxxer::Boxxer3D&lt; FloatT, IdxT &gt;</a>	21
<a href="#">BoxxerError</a>	
<a href="#">boxxer::LogicalError</a>	54
<a href="#">boxxer::NumericalError</a>	60
<a href="#">boxxer::ParameterShapeError</a>	61
<a href="#">boxxer::ParameterValueError</a>	62
<a href="#">boxxer::GaussFIRFilter&lt; FloatT, IdxT &gt;</a>	43
<a href="#">boxxer::DoGFilter2D&lt; FloatT, IdxT &gt;</a>	27
<a href="#">boxxer::DoGFilter3D&lt; FloatT, IdxT &gt;</a>	31
<a href="#">boxxer::GaussFilter2D&lt; FloatT, IdxT &gt;</a>	35
<a href="#">boxxer::GaussFilter3D&lt; FloatT, IdxT &gt;</a>	39
<a href="#">boxxer::LoGFilter2D&lt; FloatT, IdxT &gt;</a>	46

<a href="#">boxxer::LoGFilter3D&lt; FloatT, IdxT &gt;</a>	50
<a href="#">boxxer::Maxima2D&lt; FloatT, IdxT &gt;</a>	55
<a href="#">boxxer::Maxima3D&lt; FloatT, IdxT &gt;</a>	57

## 4 Class Index

### 4.1 Class List

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

<a href="#">boxxer::Boxxer2D&lt; FloatT, IdxT &gt;</a>	14
<a href="#">boxxer::Boxxer3D&lt; FloatT, IdxT &gt;</a>	21
<a href="#">boxxer::DoGFilter2D&lt; FloatT, IdxT &gt;</a>	27
<a href="#">boxxer::DoGFilter3D&lt; FloatT, IdxT &gt;</a>	31
<a href="#">boxxer::GaussFilter2D&lt; FloatT, IdxT &gt;</a>	35
<a href="#">boxxer::GaussFilter3D&lt; FloatT, IdxT &gt;</a>	39
<a href="#">boxxer::GaussFIRFilter&lt; FloatT, IdxT &gt;</a>	43
<a href="#">boxxer::LoGFilter2D&lt; FloatT, IdxT &gt;</a>	46
<a href="#">boxxer::LoGFilter3D&lt; FloatT, IdxT &gt;</a>	50
<a href="#">boxxer::LogicalError</a>	
Internal logical error. Bad logic or broken promises	54
<a href="#">boxxer::Maxima2D&lt; FloatT, IdxT &gt;</a>	55
<a href="#">boxxer::Maxima3D&lt; FloatT, IdxT &gt;</a>	57
<a href="#">boxxer::NumericalError</a>	
Internal numerical error	60
<a href="#">boxxer::ParameterShapeError</a>	
Parameters are the incorrect shape, size or number of dimensions	61
<a href="#">boxxer::ParameterValueError</a>	
Parameter value is not valid	62

## 5 File Index

### 5.1 File List

Here is a list of all files with brief descriptions:

<a href="#">Boxxer2D.h</a>	
The class declaration for Boxxer2D	64
<a href="#">Boxxer3D.h</a>	
The class declaration for Boxxer3D	65
<a href="#">BoxxerError.h</a>	
Error handling	66
<a href="#">FilterKernels.h</a>	
The <a href="#">boxxer::kernels</a> namespace - low-level Gaussian finite-impulse response filters	67
<a href="#">GaussFilter.h</a>	
The class declarations for Gaussian image filter classes	69
<a href="#">Maxima.h</a>	
The class declaration for the local maxima finders Maxima2D and Maxima3D	70

## 6 Namespace Documentation

### 6.1 boxxer Namespace Reference

#### Namespaces

- [kernels](#)

#### Classes

- class [Boxxer2D](#)
- class [Boxxer3D](#)
- class [DoGFilter2D](#)
- class [DoGFilter3D](#)
- class [GaussFilter2D](#)
- class [GaussFilter3D](#)
- class [GaussFIRFilter](#)
- class [LoGFilter2D](#)
- class [LoGFilter3D](#)
- struct [LogicalError](#)
  - Internal logical error. Bad logic or broken promises.*
- class [Maxima2D](#)
- class [Maxima3D](#)
- struct [NumericalError](#)
  - Internal numerical error.*
- struct [ParameterShapeError](#)
  - Parameters are the incorrect shape, size or number of dimensions.*
- struct [ParameterValueError](#)
  - Parameter value is not valid.*



## Typedefs

- using [BoxxerError](#) = `backtrace_exception::BacktraceException`

## 6.1.1 Typedef Documentation

6.1.1.1 using `boxxer::BoxxerError = typedef backtrace_exception::BacktraceException`

Definition at line 14 of file `BoxxerError.h`.

## 6.2 boxxer::kernels Namespace Reference

## Functions

**1D Gauss FIR Filters**

*1D Gaussian finite-impulse response filters.*

- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_1D](#) (IntT size, const FloatT data[], FloatT fdata[], IntT hw, const FloatT kernel[])
- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_1D](#) (const arma::Col< FloatT > &data, arma::Col< FloatT > &fdata, const arma::Col< FloatT > &kernel)
- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_1D\\_small](#) (IntT size, const FloatT data[], FloatT fdata[], IntT hw, const FloatT kernel[])
- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_1D\\_arma](#) (const arma::Col< FloatT > &data, arma::Col< FloatT > &fdata, const arma::Col< FloatT > &kernel)
- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_1D\\_inplace\\_arma](#) (arma::Col< FloatT > &data, const arma::Col< FloatT > &kernel)
- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_1D\\_inplace](#) (IntT size, FloatT data[], IntT hw, const FloatT kernel[])

**2D Gauss FIR Filters**

*2D Gaussian finite-impulse response filters.*

- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_2Dx](#) (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)
- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_2Dx\\_small](#) (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)
- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_2Dx\\_arma](#) (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)
- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_2Dy](#) (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)
- `template<class FloatT = float, class IntT = int32_t>`  
void [gaussFIR\\_2Dy\\_rowmajor](#) (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)

- `template<class FloatT = float, class IntT = int32_t>`  
`void gaussFIR_2Dy_colmajor (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void gaussFIR_2Dy_small (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)`

### 3D Gauss FIR Filters

*3D Gaussian finite-impulse response filters.*

- `template<class FloatT = float, class IntT = int32_t>`  
`void gaussFIR_3Dx (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void gaussFIR_3Dx_small (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void gaussFIR_3Dy (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void gaussFIR_3Dy_small (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void gaussFIR_3Dz (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void gaussFIR_3Dz_small (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`

#### 6.2.1 Detailed Description

Gaussian finite-impulse response kernels: 1D, 2D, and 3D

Template parameters for all namespace member functions. FloatT - float (default) or double. IntT - **signed** integer: int32\_t (default) or int64\_t.

All kernels are explicitly instantiated for:

- FloatT = float, IntT = int32\_t
- FloatT = double, IntT = int32\_t

There is no testing for integer overflow. Probably int32\_t will be sensible for most applications.

#### 6.2.2 Function Documentation

6.2.2.1 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_1D ( IntT size, const FloatT data[], FloatT fdata[], IntT hw, const FloatT kernel[] )`

This is details?

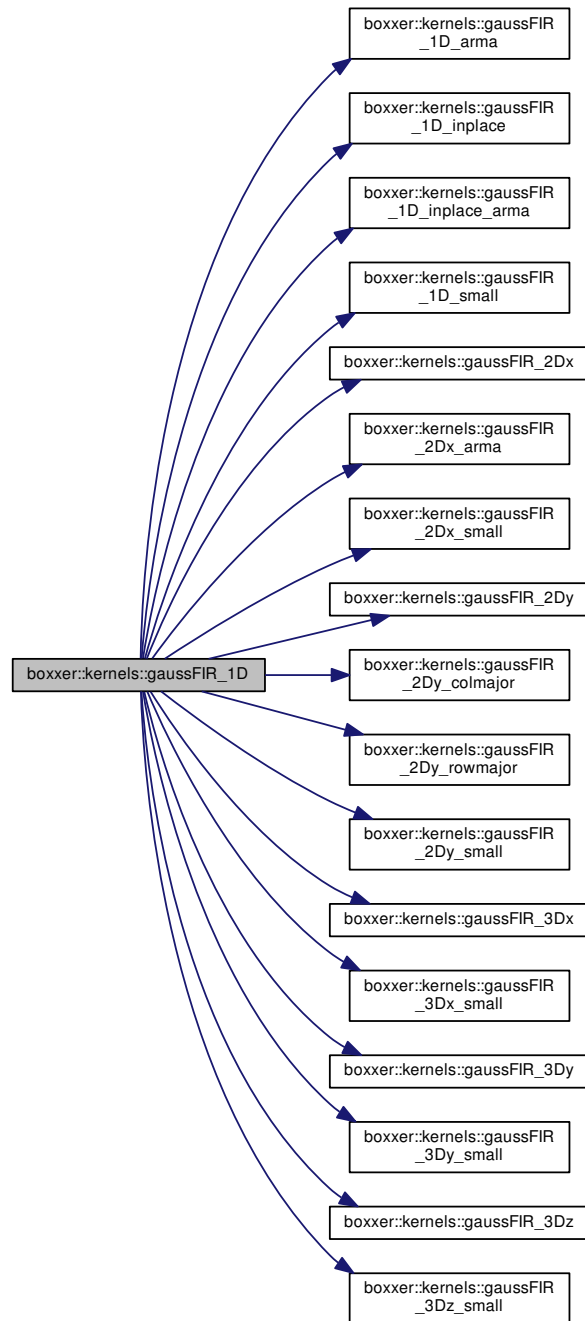
6.2.2.2 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_1D ( const arma::Col< FloatT > & data, arma::Col< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

This is details?

Definition at line 39 of file FilterKernels.h.

References `gaussFIR_1D_arma()`, `gaussFIR_1D_inplace()`, `gaussFIR_1D_inplace_arma()`, `gaussFIR_1D_small()`, `gaussFIR_2Dx()`, `gaussFIR_2Dx_arma()`, `gaussFIR_2Dx_small()`, `gaussFIR_2Dy()`, `gaussFIR_2Dy_colmajor()`, `gaussFIR_2Dy_rowmajor()`, `gaussFIR_2Dy_small()`, `gaussFIR_3Dx()`, `gaussFIR_3Dx_small()`, `gaussFIR_3Dy()`, `gaussFIR_3Dy_small()`, `gaussFIR_3Dz()`, and `gaussFIR_3Dz_small()`.

Here is the call graph for this function:

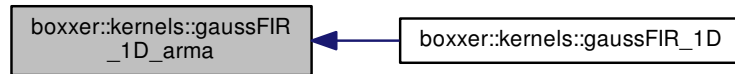


6.2.2.3 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_1D_arma ( const arma::Col< FloatT > & data, arma::Col< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

This is details?

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

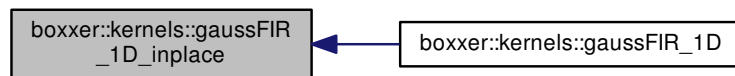


6.2.2.4 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_1D_inplace ( IntT size, FloatT data[], IntT hw, const FloatT kernel[] )`

This is details?

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

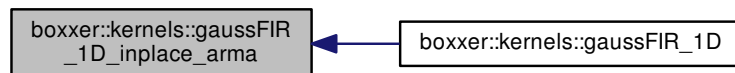


6.2.2.5 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_1D_inplace_arma ( arma::Col< FloatT > & data, const arma::Col< FloatT > & kernel )`

This is details?

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

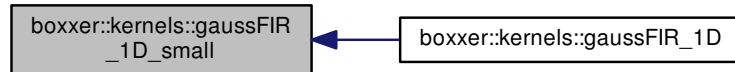


6.2.2.6 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_1D_small ( IntT size, const FloatT data[], FloatT fdata[], IntT hw, const FloatT kernel[] )`

This is details?

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:



6.2.2.7 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_2Dx ( const arma::Mat< FloatT > & data, arma::Mat< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

2D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

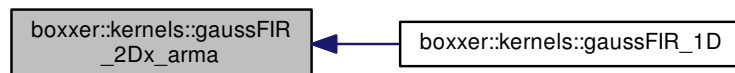


6.2.2.8 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_2Dx_arma ( const arma::Mat< FloatT > & data, arma::Mat< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

2D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

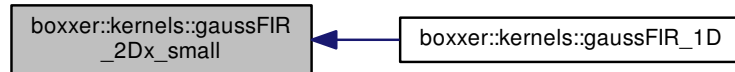


6.2.2.9 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_2Dx_small ( const arma::Mat< FloatT > & data, arma::Mat< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

2D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:



6.2.2.10 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_2Dy ( const arma::Mat< FloatT > & data, arma::Mat< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

2D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

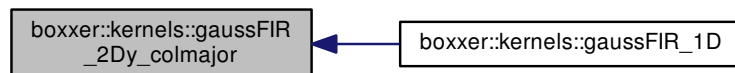


6.2.2.11 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_2Dy_colmajor ( const arma::Mat< FloatT > & data, arma::Mat< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

2D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

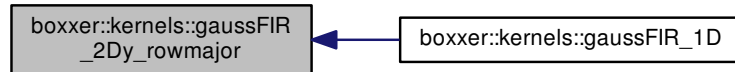


6.2.2.12 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_2Dy_rowmajor ( const arma::Mat< FloatT > & data, arma::Mat< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

2D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

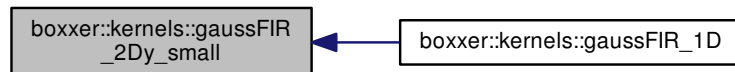


6.2.2.13 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_2Dy_small ( const arma::Mat< FloatT > & data, arma::Mat< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

2D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:



6.2.2.14 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_3Dx ( const arma::Cube< FloatT > & data, arma::Cube< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

3D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:



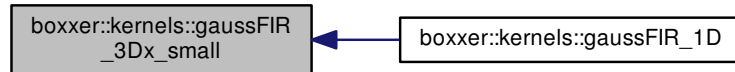


6.2.2.15 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_3Dx_small ( const arma::Cube< FloatT > & data, arma::Cube< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

3D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:



6.2.2.16 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_3Dy ( const arma::Cube< FloatT > & data, arma::Cube< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

3D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

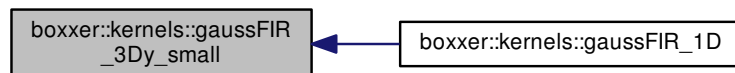


6.2.2.17 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_3Dy_small ( const arma::Cube< FloatT > & data, arma::Cube< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

3D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:



6.2.2.18 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_3Dz ( const arma::Cube< FloatT > & data, arma::Cube< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

3D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:

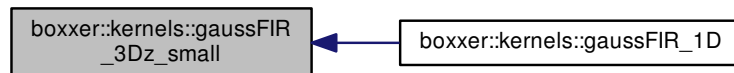


6.2.2.19 `template<class FloatT = float, class IntT = int32_t> void boxxer::kernels::gaussFIR_3Dz_small ( const arma::Cube< FloatT > & data, arma::Cube< FloatT > & fdata, const arma::Col< FloatT > & kernel )`

3D Gauss FIR Filters

Referenced by gaussFIR\_1D().

Here is the caller graph for this function:



## 7 Class Documentation

### 7.1 `boxxer::Boxxer2D< FloatT, IdxT >` Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/Boxxer2D.h>
```

## Public Types

- using [IVecT](#) = arma::Col< IdxT >
- using [IMatT](#) = arma::Mat< IdxT >
- using [VecT](#) = arma::Col< FloatT >
- using [MatT](#) = arma::Mat< FloatT >
- using [ImageT](#) = arma::Mat< FloatT >
- using [ImageStackT](#) = arma::Cube< FloatT >
- using [ScaledImageT](#) = arma::Cube< FloatT >
- using [ScaledImageStackT](#) = hypercube::Hypercube< FloatT >

## Public Member Functions

- [Boxxer2D](#) (const [IVecT](#) &imsize, const [MatT](#) &sigma)
- void [setDoGSigmaRatio](#) (FloatT [sigma\\_ratio](#))
- void [filterScaledLoG](#) (const [ImageStackT](#) &im, [ScaledImageStackT](#) &fim) const
- void [filterScaledDoG](#) (const [ImageStackT](#) &im, [ScaledImageStackT](#) &fim) const
- IdxT [scaleSpaceLoGMaxima](#) (const [ImageStackT](#) &im, [IMatT](#) &maxima, [VecT](#) &max\_vals, IdxT neighborhood←\_size, IdxT scale\_neighborhood\_size) const
- IdxT [scaleSpaceDoGMaxima](#) (const [ImageStackT](#) &im, [IMatT](#) &maxima, [VecT](#) &max\_vals, IdxT neighborhood←\_size, IdxT scale\_neighborhood\_size) const
- [ImageT](#) [make\\_image](#) () const
- [ImageStackT](#) [make\\_image\\_stack](#) (IdxT nT) const
- [ScaledImageT](#) [make\\_scaled\\_image](#) () const
- [ScaledImageStackT](#) [make\\_scaled\\_image\\_stack](#) (IdxT nT) const

## Static Public Member Functions

- static void [filterLoG](#) (const [ImageStackT](#) &im, [ImageStackT](#) &fim, const [VecT](#) &sigma)
- static void [filterDoG](#) (const [ImageStackT](#) &im, [ImageStackT](#) &fim, const [VecT](#) &sigma, FloatT [sigma\\_ratio](#))
- static void [filterGauss](#) (const [ImageStackT](#) &im, [ImageStackT](#) &fim, const [VecT](#) &sigma)
- static void [checkMaxima](#) (const [ImageStackT](#) &im, [IMatT](#) &maxima, [VecT](#) &max\_vals)
- static IdxT [enumerateImageMaxima](#) (const [ImageStackT](#) &im, [IMatT](#) &maxima, [VecT](#) &max\_vals, IdxT neighborhood\_size)

## Public Attributes

- IdxT [nScales](#)
- [IVecT](#) [imsize](#)
- [MatT](#) [sigma](#)
- FloatT [sigma\\_ratio](#)

## Static Public Attributes

- static const FloatT [DefaultSigmaRatio](#)
- static const IdxT [dim](#)

### 7.1.1 Detailed Description

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::Boxxer2D< FloatT, IdxT >
```

In this class we make the assumption that images are stored in column-major format and that x=rows, y=cols, t=slices. This relationship is important in the choice of imsize and sigma parameters.

```
imsize = [nrows, ncols, nframes]; sigma = [ sigma_rows (X scale=1), sigma_rows (X scale=2); sigma_cols(Y scale=1),
sigma_cols (Y scale=2)]
```

This is contrary to normal image coordinates in matlab, but for this low level it is easier to think about x as the first index into an image and understand that the meaning for "X" and "Y" will be reversed from the matlab interpretation, but only internally within the Boxxer\_IFace MexIFace class.

Definition at line 33 of file Boxxer2D.h.

### 7.1.2 Member Typedef Documentation

```
7.1.2.1 template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer2D< FloatT, IdxT >::ImageStackT =
arma::Cube<FloatT>
```

Definition at line 41 of file Boxxer2D.h.

```
7.1.2.2 template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer2D< FloatT, IdxT >::ImageT =
arma::Mat<FloatT>
```

Definition at line 40 of file Boxxer2D.h.

```
7.1.2.3 template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer2D< FloatT, IdxT >::IMatT =
arma::Mat<IdxT>
```

Definition at line 37 of file Boxxer2D.h.

```
7.1.2.4 template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer2D< FloatT, IdxT >::IVecT =
arma::Col<IdxT>
```

Definition at line 36 of file Boxxer2D.h.

```
7.1.2.5 template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer2D< FloatT, IdxT >::MatT =
arma::Mat<FloatT>
```

Definition at line 39 of file Boxxer2D.h.

```
7.1.2.6 template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer2D< FloatT, IdxT >::ScaledImageStackT
= hypercube::Hypercube<FloatT>
```

Definition at line 43 of file Boxxer2D.h.

7.1.2.7 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer2D< FloatT, IdxT >::ScaledImageT = arma::Cube<FloatT>`

Definition at line 42 of file Boxxer2D.h.

7.1.2.8 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer2D< FloatT, IdxT >::VecT = arma::Col<FloatT>`

Definition at line 38 of file Boxxer2D.h.

### 7.1.3 Constructor & Destructor Documentation

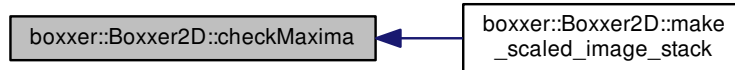
7.1.3.1 `template<class FloatT = float, class IdxT = uint32_t> boxxer::Boxxer2D< FloatT, IdxT >::Boxxer2D ( const IVecT & imsize, const MatT & sigma )`

### 7.1.4 Member Function Documentation

7.1.4.1 `template<class FloatT = float, class IdxT = uint32_t> static void boxxer::Boxxer2D< FloatT, IdxT >::checkMaxima ( const ImageStackT & im, IMatT & maxima, VecT & max_vals ) [static]`

Referenced by `boxxer::Boxxer2D< FloatT, IdxT >::make_scaled_image_stack()`.

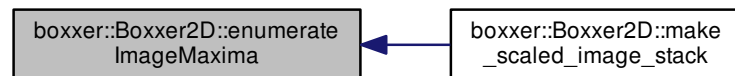
Here is the caller graph for this function:



7.1.4.2 `template<class FloatT = float, class IdxT = uint32_t> static IdxT boxxer::Boxxer2D< FloatT, IdxT >::enumerateImageMaxima ( const ImageStackT & im, IMatT & maxima, VecT & max_vals, IdxT neighborhood_size ) [static]`

Referenced by `boxxer::Boxxer2D< FloatT, IdxT >::make_scaled_image_stack()`.

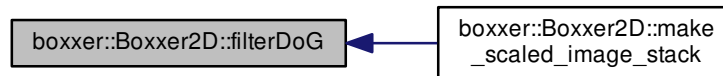
Here is the caller graph for this function:



7.1.4.3 `template<class FloatT = float, class IdxT = uint32_t> static void boxxer::Boxxer2D< FloatT, IdxT >::filterDoG ( const ImageStackT & im, ImageStackT & fim, const Vect & sigma, FloatT sigma_ratio ) [static]`

Referenced by `boxxer::Boxxer2D< FloatT, IdxT >::make_scaled_image_stack()`.

Here is the caller graph for this function:



7.1.4.4 `template<class FloatT = float, class IdxT = uint32_t> static void boxxer::Boxxer2D< FloatT, IdxT >::filterGauss ( const ImageStackT & im, ImageStackT & fim, const Vect & sigma ) [static]`

Referenced by `boxxer::Boxxer2D< FloatT, IdxT >::make_scaled_image_stack()`.

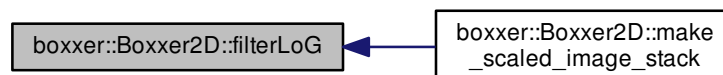
Here is the caller graph for this function:



7.1.4.5 `template<class FloatT = float, class IdxT = uint32_t> static void boxxer::Boxxer2D< FloatT, IdxT >::filterLoG ( const ImageStackT & im, ImageStackT & fim, const Vect & sigma ) [static]`

Referenced by `boxxer::Boxxer2D< FloatT, IdxT >::make_scaled_image_stack()`.

Here is the caller graph for this function:



7.1.4.6 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Boxxer2D< FloatT, IdxT >::filterScaledDoG ( const ImageStackT & im, ScaledImageStackT & fim ) const`

7.1.4.7 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Boxxer2D< FloatT, IdxT >::filterScaledLoG ( const ImageStackT & im, ScaledImageStackT & fim ) const`

7.1.4.8 `template<class FloatT = float, class IdxT = uint32_t> ImageT boxxer::Boxxer2D< FloatT, IdxT >::make_image ( ) const [inline]`

Definition at line 61 of file Boxxer2D.h.

References `boxxer::Boxxer2D< FloatT, IdxT >::imshow`.

7.1.4.9 `template<class FloatT = float, class IdxT = uint32_t> ImageStackT boxxer::Boxxer2D< FloatT, IdxT >::make_image_stack ( IdxT nT ) const [inline]`

Definition at line 62 of file Boxxer2D.h.

References `boxxer::Boxxer2D< FloatT, IdxT >::imshow`.

7.1.4.10 `template<class FloatT = float, class IdxT = uint32_t> ScaledImageT boxxer::Boxxer2D< FloatT, IdxT >::make_scaled_image ( ) const [inline]`

Definition at line 63 of file Boxxer2D.h.

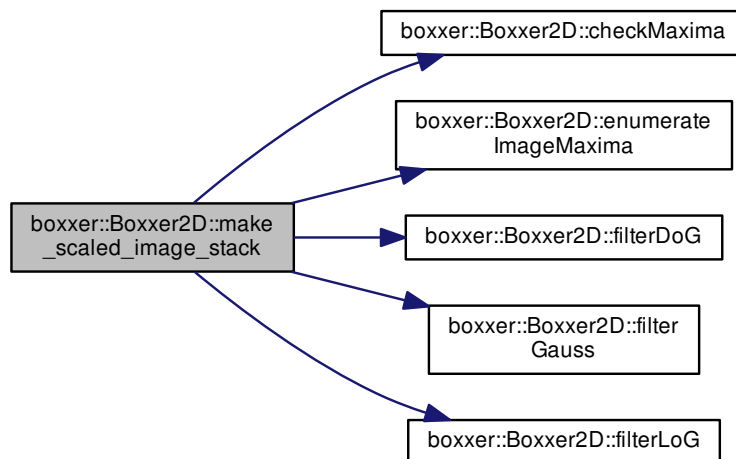
References `boxxer::Boxxer2D< FloatT, IdxT >::imshow`.

7.1.4.11 `template<class FloatT = float, class IdxT = uint32_t> ScaledImageStackT boxxer::Boxxer2D< FloatT, IdxT >::make_scaled_image_stack ( IdxT nT ) const [inline]`

Definition at line 64 of file Boxxer2D.h.

References `boxxer::Boxxer2D< FloatT, IdxT >::checkMaxima()`, `boxxer::Boxxer2D< FloatT, IdxT >::enumerateImageMaxima()`, `boxxer::Boxxer2D< FloatT, IdxT >::filterDoG()`, `boxxer::Boxxer2D< FloatT, IdxT >::filterGauss()`, `boxxer::Boxxer2D< FloatT, IdxT >::filterLoG()`, and `boxxer::Boxxer2D< FloatT, IdxT >::imshow`.

Here is the call graph for this function:



7.1.4.12 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Boxxer2D< FloatT, IdxT >::scaleSpaceDoGMaxima ( const ImageStackT & im, IMatT & maxima, VecT & max_vals, IdxT neighborhood_size, IdxT scale_neighborhood_size ) const`

7.1.4.13 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Boxxer2D< FloatT, IdxT >::scaleSpaceLoGMaxima ( const ImageStackT & im, IMatT & maxima, VecT & max_vals, IdxT neighborhood_size, IdxT scale_neighborhood_size ) const`

7.1.4.14 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Boxxer2D< FloatT, IdxT >::setDoGSigmaRatio ( FloatT sigma_ratio )`

### 7.1.5 Member Data Documentation

7.1.5.1 `template<class FloatT = float, class IdxT = uint32_t> const FloatT boxxer::Boxxer2D< FloatT, IdxT >::DefaultSigmaRatio [static]`

Definition at line 45 of file `Boxxer2D.h`.

7.1.5.2 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::Boxxer2D< FloatT, IdxT >::dim [static]`

Definition at line 46 of file `Boxxer2D.h`.

7.1.5.3 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::Boxxer2D< FloatT, IdxT >::imsizes`

Definition at line 49 of file `Boxxer2D.h`.

Referenced by `boxxer::Boxxer2D< FloatT, IdxT >::make_image()`, `boxxer::Boxxer2D< FloatT, IdxT >::make_image_stack()`, `boxxer::Boxxer2D< FloatT, IdxT >::make_scaled_image()`, and `boxxer::Boxxer2D< FloatT, IdxT >::make_scaled_image_stack()`.

7.1.5.4 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Boxxer2D< FloatT, IdxT >::nScales`

Definition at line 48 of file `Boxxer2D.h`.

7.1.5.5 `template<class FloatT = float, class IdxT = uint32_t> MatT boxxer::Boxxer2D< FloatT, IdxT >::sigma`

Definition at line 50 of file `Boxxer2D.h`.

7.1.5.6 `template<class FloatT = float, class IdxT = uint32_t> FloatT boxxer::Boxxer2D< FloatT, IdxT >::sigma_ratio`

Definition at line 51 of file `Boxxer2D.h`.

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

- [Boxxer2D.h](#)



## 7.2 boxxer::Boxxer3D&lt; FloatT, IdxT &gt; Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/Boxxer3D.h>
```

## Public Types

- using [IVecT](#) = arma::Col< IdxT >
- using [IMatT](#) = arma::Mat< IdxT >
- using [VecT](#) = arma::Col< FloatT >
- using [MatT](#) = arma::Mat< FloatT >
- using [ImageT](#) = arma::Cube< FloatT >
- using [ImageStackT](#) = hypercube::Hypercube< FloatT >
- using [ScaledImageT](#) = hypercube::Hypercube< FloatT >

## Public Member Functions

- [Boxxer3D](#) (const [IVecT](#) &size, const [MatT](#) &sigma)
- void [setDoGSigmaRatio](#) (FloatT [sigma\\_ratio](#))
- void [filterScaledLoG](#) (const [ImageT](#) &im, [ScaledImageT](#) &fim)
- void [filterScaledDoG](#) (const [ImageT](#) &im, [ScaledImageT](#) &fim)
- IdxT [scaleSpaceLoGMaxima](#) (const [ImageStackT](#) &im, [IMatT](#) &maxima, [VecT](#) &max\_vals, IdxT neighborhood←\_size, IdxT scale\_neighborhood\_size)
- IdxT [scaleSpaceDoGMaxima](#) (const [ImageStackT](#) &im, [IMatT](#) &maxima, [VecT](#) &max\_vals, IdxT neighborhood←\_size, IdxT scale\_neighborhood\_size)
- [ImageT](#) [make\\_image](#) () const
- [ImageStackT](#) [make\\_image\\_stack](#) (IdxT nT) const
- [ScaledImageT](#) [make\\_scaled\\_image](#) () const

## Static Public Member Functions

- static void [filterLoG](#) (const [ImageStackT](#) &im, [ImageStackT](#) &fim, const [VecT](#) &sigma)
- static void [filterDoG](#) (const [ImageStackT](#) &im, [ImageStackT](#) &fim, const [VecT](#) &sigma, FloatT [sigma\\_ratio](#))
- static void [filterGauss](#) (const [ImageStackT](#) &im, [ImageStackT](#) &fim, const [VecT](#) &sigma)
- static void [checkMaxima](#) (const [ImageStackT](#) &im, [IMatT](#) &maxima, [VecT](#) &max\_vals)
- static IdxT [enumerateImageMaxima](#) (const [ImageStackT](#) &im, [IMatT](#) &maxima, [VecT](#) &max\_vals, IdxT neighborhood\_size)

## Public Attributes

- IdxT [nScales](#)
- [IVecT](#) [imsize](#)
- [MatT](#) [sigma](#)
- FloatT [sigma\\_ratio](#)

## Static Public Attributes

- static const FloatT [DefaultSigmaRatio](#)
- static const IdxT [dim](#)

### 7.2.1 Detailed Description

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::Boxxer3D< FloatT, IdxT >
```

A box finding algorithm for 3D hyper-spectral microscopy data.

Estimates the center coordinates of Gaussian blobs with anisotropic sigmas.

All image data manipulated is stored as column-major FloatT arrays with dimension ordering [L Y X T].

The [Boxxer3D](#) class makes uses of lower level class which are agnostic about the data source being hyperspectral, they don't care what the coordinate dimensions represent scientifically, but this class is associated with the Matlab [Boxxer3D](#) class and so maintains the knowledge that the actual coordinates are [L Y X T].

Definition at line 27 of file `Boxxer3D.h`.

### 7.2.2 Member Typedef Documentation

**7.2.2.1** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer3D< FloatT, IdxT >::ImageStackT = hypercube::Hypercube<FloatT>`

Definition at line 35 of file `Boxxer3D.h`.

**7.2.2.2** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer3D< FloatT, IdxT >::ImageT = arma::Cube<FloatT>`

Definition at line 34 of file `Boxxer3D.h`.

**7.2.2.3** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer3D< FloatT, IdxT >::IMatT = arma::Mat<IdxT>`

Definition at line 31 of file `Boxxer3D.h`.

**7.2.2.4** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer3D< FloatT, IdxT >::IVecT = arma::Col<IdxT>`

Definition at line 30 of file `Boxxer3D.h`.

**7.2.2.5** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer3D< FloatT, IdxT >::MatT = arma::Mat<FloatT>`

Definition at line 33 of file `Boxxer3D.h`.

**7.2.2.6** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer3D< FloatT, IdxT >::ScaledImageT = hypercube::Hypercube<FloatT>`

Definition at line 36 of file `Boxxer3D.h`.

7.2.2.7 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Boxxer3D< FloatT, IdxT >::VecT = arma::Col<FloatT>`

Definition at line 32 of file Boxxer3D.h.

### 7.2.3 Constructor & Destructor Documentation

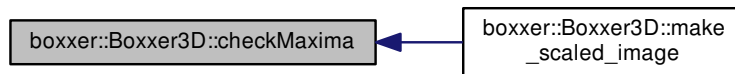
7.2.3.1 `template<class FloatT = float, class IdxT = uint32_t> boxxer::Boxxer3D< FloatT, IdxT >::Boxxer3D ( const IVecT & size, const MatT & sigma )`

### 7.2.4 Member Function Documentation

7.2.4.1 `template<class FloatT = float, class IdxT = uint32_t> static void boxxer::Boxxer3D< FloatT, IdxT >::checkMaxima ( const ImageStackT & im, IMatT & maxima, VecT & max_vals ) [static]`

Referenced by `boxxer::Boxxer3D< FloatT, IdxT >::make_scaled_image()`.

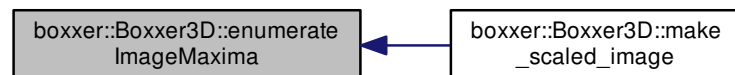
Here is the caller graph for this function:



7.2.4.2 `template<class FloatT = float, class IdxT = uint32_t> static IdxT boxxer::Boxxer3D< FloatT, IdxT >::enumerateImageMaxima ( const ImageStackT & im, IMatT & maxima, VecT & max_vals, IdxT neighborhood_size ) [static]`

Referenced by `boxxer::Boxxer3D< FloatT, IdxT >::make_scaled_image()`.

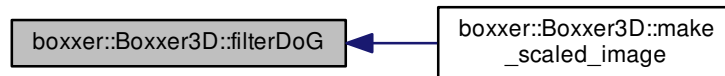
Here is the caller graph for this function:



7.2.4.3 `template<class FloatT = float, class IdxT = uint32_t> static void boxxer::Boxxer3D< FloatT, IdxT >::filterDoG ( const ImageStackT & im, ImageStackT & fim, const Vect & sigma, FloatT sigma_ratio ) [static]`

Referenced by `boxxer::Boxxer3D< FloatT, IdxT >::make_scaled_image()`.

Here is the caller graph for this function:



7.2.4.4 `template<class FloatT = float, class IdxT = uint32_t> static void boxxer::Boxxer3D< FloatT, IdxT >::filterGauss ( const ImageStackT & im, ImageStackT & fim, const Vect & sigma ) [static]`

Referenced by `boxxer::Boxxer3D< FloatT, IdxT >::make_scaled_image()`.

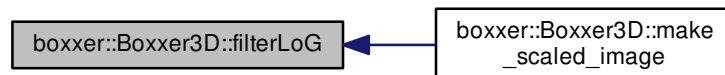
Here is the caller graph for this function:



7.2.4.5 `template<class FloatT = float, class IdxT = uint32_t> static void boxxer::Boxxer3D< FloatT, IdxT >::filterLoG ( const ImageStackT & im, ImageStackT & fim, const Vect & sigma ) [static]`

Referenced by `boxxer::Boxxer3D< FloatT, IdxT >::make_scaled_image()`.

Here is the caller graph for this function:



7.2.4.6 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Boxxer3D< FloatT, IdxT >::filterScaledDoG ( const ImageT & im, ScaledImageT & fim )`

7.2.4.7 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Boxxer3D< FloatT, IdxT >::filterScaledLoG ( const ImageT & im, ScaledImageT & fim )`

7.2.4.8 `template<class FloatT = float, class IdxT = uint32_t> ImageT boxxer::Boxxer3D< FloatT, IdxT >::make_image ( ) const [inline]`

Definition at line 55 of file Boxxer3D.h.

References `boxxer::Boxxer3D< FloatT, IdxT >::imsz`.

7.2.4.9 `template<class FloatT = float, class IdxT = uint32_t> ImageStackT boxxer::Boxxer3D< FloatT, IdxT >::make_image_stack ( IdxT nT ) const [inline]`

Definition at line 56 of file Boxxer3D.h.

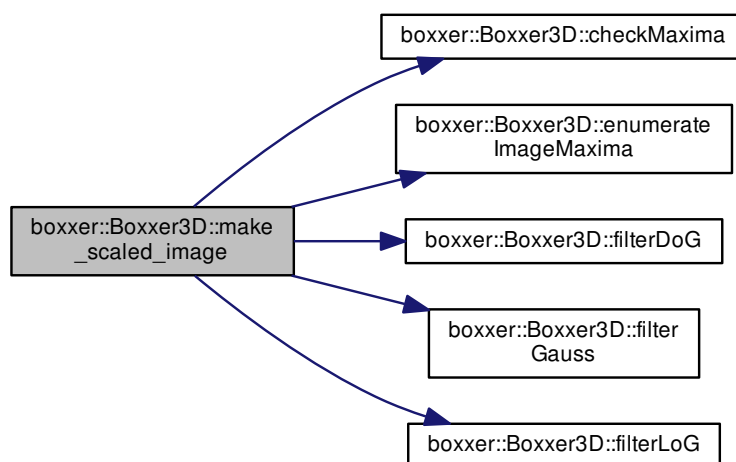
References `boxxer::Boxxer3D< FloatT, IdxT >::imsz`.

7.2.4.10 `template<class FloatT = float, class IdxT = uint32_t> ScaledImageT boxxer::Boxxer3D< FloatT, IdxT >::make_scaled_image ( ) const [inline]`

Definition at line 57 of file Boxxer3D.h.

References `boxxer::Boxxer3D< FloatT, IdxT >::checkMaxima()`, `boxxer::Boxxer3D< FloatT, IdxT >::enumerateImageMaxima()`, `boxxer::Boxxer3D< FloatT, IdxT >::filterDoG()`, `boxxer::Boxxer3D< FloatT, IdxT >::filterGauss()`, `boxxer::Boxxer3D< FloatT, IdxT >::filterLoG()`, and `boxxer::Boxxer3D< FloatT, IdxT >::imsz`.

Here is the call graph for this function:



7.2.4.11 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Boxxer3D< FloatT, IdxT >::scaleSpaceDoGMaxima ( const ImageStackT & im, IMatT & maxima, VecT & max_vals, IdxT neighborhood_size, IdxT scale_neighborhood_size )`

7.2.4.12 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Boxxer3D< FloatT, IdxT >::scaleSpaceLoGMaxima ( const ImageStackT & im, IMatT & maxima, VecT & max_vals, IdxT neighborhood_size, IdxT scale_neighborhood_size )`

7.2.4.13 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Boxxer3D< FloatT, IdxT >::setDoGSigmaRatio ( FloatT sigma_ratio )`

## 7.2.5 Member Data Documentation

7.2.5.1 `template<class FloatT = float, class IdxT = uint32_t> const FloatT boxxer::Boxxer3D< FloatT, IdxT >::DefaultSigmaRatio [static]`

Definition at line 38 of file `Boxxer3D.h`.

7.2.5.2 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::Boxxer3D< FloatT, IdxT >::dim [static]`

Definition at line 39 of file `Boxxer3D.h`.

7.2.5.3 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::Boxxer3D< FloatT, IdxT >::imsize`

Definition at line 42 of file `Boxxer3D.h`.

Referenced by `boxxer::Boxxer3D< FloatT, IdxT >::make_image()`, `boxxer::Boxxer3D< FloatT, IdxT >::make_image←_stack()`, and `boxxer::Boxxer3D< FloatT, IdxT >::make_scaled_image()`.

7.2.5.4 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Boxxer3D< FloatT, IdxT >::nScales`

Definition at line 41 of file `Boxxer3D.h`.

7.2.5.5 `template<class FloatT = float, class IdxT = uint32_t> MatT boxxer::Boxxer3D< FloatT, IdxT >::sigma`

Definition at line 43 of file `Boxxer3D.h`.

7.2.5.6 `template<class FloatT = float, class IdxT = uint32_t> FloatT boxxer::Boxxer3D< FloatT, IdxT >::sigma_ratio`

Definition at line 45 of file `Boxxer3D.h`.

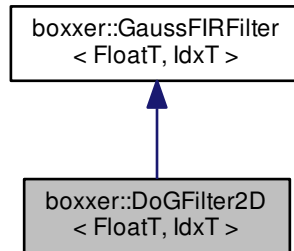
The documentation for this class was generated from the following file:

- [Boxxer3D.h](#)

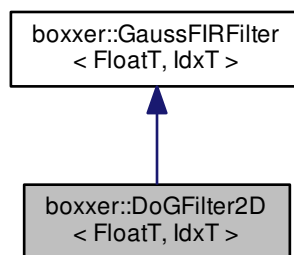
## 7.3 boxxer::DoGFilter2D&lt; FloatT, IdxT &gt; Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/GaussFilter.h>
```

Inheritance diagram for boxxer::DoGFilter2D< FloatT, IdxT >:



Collaboration diagram for boxxer::DoGFilter2D< FloatT, IdxT >:



## Public Types

- using `IVecT` = typename `GaussFIRFilter< FloatT, IdxT >::IVecT`
- using `VecT` = typename `GaussFIRFilter< FloatT, IdxT >::VecT`
- using `ImageT` = `arma::Mat< FloatT >`
- using `MatT` = `arma::Mat< FloatT >`

**Public Member Functions**

- [DoGFilter2D](#) (const [IVecT](#) &size, const [VecT](#) &sigma, [FloatT](#) sigma\_ratio)
- [DoGFilter2D](#) (const [IVecT](#) &size, const [VecT](#) &sigma, [FloatT](#) sigma\_ratio, const [IVecT](#) &kernel\_hw)
- void [set\\_kernel\\_hw](#) (const [IVecT](#) &kernel\_half\_width)
- void [set\\_sigma\\_ratio](#) ([FloatT](#) sigma\_ratio)
- [ImageT](#) [make\\_image](#) () const
- void [filter](#) (const [ImageT](#) &im, [ImageT](#) &out)
- void [test\\_filter](#) (const [ImageT](#) &im)

**Static Public Member Functions**

- static [VecT](#) [compute\\_Gauss\\_FIR\\_kernel](#) ([FloatT](#) sigma, [IdxT](#) hw)
- static [VecT](#) [compute\\_Log\\_FIR\\_kernel](#) ([FloatT](#) sigma, [IdxT](#) hw)

**Public Attributes**

- [FloatT](#) sigma\_ratio
- [IdxT](#) dim
- [IVecT](#) size
- [VecT](#) sigma
- [IVecT](#) hw

**Static Protected Attributes**

- static const [IdxT](#) max\_kernel\_hw
- static const [FloatT](#) default\_sigma\_hw\_ratio

**Friends**

- template<class [FloatT\\_](#), class [IdxT\\_](#) >  
std::ostream & [operator<<](#) (std::ostream &out, const [DoGFilter2D](#)< [FloatT\\_](#), [IdxT\\_](#) > &filt)

**7.3.1 Detailed Description**

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::DoGFilter2D< FloatT, IdxT >
```

Definition at line 70 of file GaussFilter.h.

**7.3.2 Member Typedef Documentation**

**7.3.2.1** template<class [FloatT](#) = float, class [IdxT](#) = uint32\_t> using boxxer::DoGFilter2D< [FloatT](#), [IdxT](#) >::ImageT = arma::Mat<[FloatT](#)>

Definition at line 75 of file GaussFilter.h.



7.3.2.2 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::DoGFilter2D< FloatT, IdxT >::IVecT = typename GaussFIRFilter<FloatT,IdxT>::IVecT`

Definition at line 73 of file GaussFilter.h.

7.3.2.3 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFIRFilter< FloatT, IdxT >::MatT = arma::Mat<FloatT> [inherited]`

Definition at line 26 of file GaussFilter.h.

7.3.2.4 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::DoGFilter2D< FloatT, IdxT >::VecT = typename GaussFIRFilter<FloatT,IdxT>::VecT`

Definition at line 74 of file GaussFilter.h.

### 7.3.3 Constructor & Destructor Documentation

7.3.3.1 `template<class FloatT = float, class IdxT = uint32_t> boxxer::DoGFilter2D< FloatT, IdxT >::DoGFilter2D ( const IVecT & size, const VecT & sigma, FloatT sigma_ratio )`

7.3.3.2 `template<class FloatT = float, class IdxT = uint32_t> boxxer::DoGFilter2D< FloatT, IdxT >::DoGFilter2D ( const IVecT & size, const VecT & sigma, FloatT sigma_ratio, const IVecT & kernel_hw )`

### 7.3.4 Member Function Documentation

7.3.4.1 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Gauss_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

7.3.4.2 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Log_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

7.3.4.3 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::DoGFilter2D< FloatT, IdxT >::filter ( const ImageT & im, ImageT & out )`

7.3.4.4 `template<class FloatT = float, class IdxT = uint32_t> ImageT boxxer::DoGFilter2D< FloatT, IdxT >::make_image ( ) const [inline]`

Definition at line 83 of file GaussFilter.h.

References `boxxer::GaussFIRFilter< FloatT, IdxT >::size`.

7.3.4.5 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::DoGFilter2D< FloatT, IdxT >::set_kernel_hw ( const IVecT & kernel_half_width ) [virtual]`

Implements [boxxer::GaussFIRFilter< FloatT, IdxT >](#).

7.3.4.6 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::DoGFilter2D< FloatT, IdxT >::set_sigma_ratio ( FloatT sigma_ratio )`

7.3.4.7 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::DoGFilter2D< FloatT, IdxT >::test_filter ( const ImageT & im )`

### 7.3.5 Friends And Related Function Documentation

7.3.5.1 `template<class FloatT = float, class IdxT = uint32_t> template<class FloatT_, class IdxT_ > std::ostream& operator<< ( std::ostream & out, const DoGFilter2D< FloatT_, IdxT_ > & filt ) [friend]`

### 7.3.6 Member Data Documentation

7.3.6.1 `template<class FloatT = float, class IdxT = uint32_t> const FloatT boxxer::GaussFIRFilter< FloatT, IdxT >::default_sigma_hw_ratio [static], [protected], [inherited]`

Definition at line 41 of file GaussFilter.h.

7.3.6.2 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::dim [inherited]`

Definition at line 28 of file GaussFilter.h.

7.3.6.3 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::hw [inherited]`

Definition at line 31 of file GaussFilter.h.

7.3.6.4 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::max_kernel_hw [static], [protected], [inherited]`

Definition at line 40 of file GaussFilter.h.

7.3.6.5 `template<class FloatT = float, class IdxT = uint32_t> VecT boxxer::GaussFIRFilter< FloatT, IdxT >::sigma [inherited]`

Definition at line 30 of file GaussFilter.h.

7.3.6.6 `template<class FloatT = float, class IdxT = uint32_t> FloatT boxxer::DoGFilter2D< FloatT, IdxT >::sigma_ratio`

Definition at line 77 of file GaussFilter.h.

7.3.6.7 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::size`  
[inherited]

Definition at line 29 of file GaussFilter.h.

Referenced by `boxxer::GaussFilter2D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::LoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::GaussFilter3D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter3D< FloatT, IdxT >::make_image()`, and `boxxer::LoGFilter3D< FloatT, IdxT >::make_image()`.

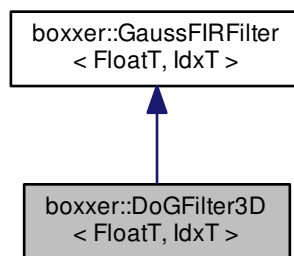
The documentation for this class was generated from the following file:

- [GaussFilter.h](#)

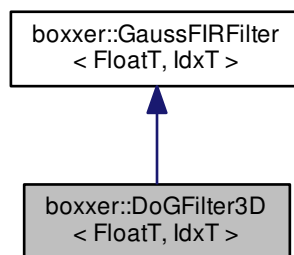
## 7.4 boxxer::DoGFilter3D< FloatT, IdxT > Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/GaussFilter.h>
```

Inheritance diagram for `boxxer::DoGFilter3D< FloatT, IdxT >`:



Collaboration diagram for `boxxer::DoGFilter3D< FloatT, IdxT >`:



### Public Types

- using [IVecT](#) = typename [GaussFIRFilter](#)< FloatT, IdxT >::IVecT
- using [VecT](#) = typename [GaussFIRFilter](#)< FloatT, IdxT >::VecT
- using [ImageT](#) = arma::Cube< FloatT >
- using [MatT](#) = arma::Mat< FloatT >

### Public Member Functions

- [DoGFilter3D](#) (const [IVecT](#) &size, const [VecT](#) &sigma, FloatT sigma\_ratio)
- [DoGFilter3D](#) (const [IVecT](#) &size, const [VecT](#) &sigma, FloatT sigma\_ratio, const [IVecT](#) &kernel\_hw)
- void [set\\_kernel\\_hw](#) (const [IVecT](#) &kernel\_half\_width)
- void [set\\_sigma\\_ratio](#) (FloatT sigma\_ratio)
- [ImageT](#) [make\\_image](#) () const
- void [filter](#) (const [ImageT](#) &im, [ImageT](#) &out)
- void [test\\_filter](#) (const [ImageT](#) &im)

### Static Public Member Functions

- static [VecT](#) [compute\\_Gauss\\_FIR\\_kernel](#) (FloatT sigma, IdxT hw)
- static [VecT](#) [compute\\_Log\\_FIR\\_kernel](#) (FloatT sigma, IdxT hw)

### Public Attributes

- FloatT [sigma\\_ratio](#)
- IdxT [dim](#)
- [IVecT](#) [size](#)
- [VecT](#) [sigma](#)
- [IVecT](#) [hw](#)

### Static Protected Attributes

- static const IdxT [max\\_kernel\\_hw](#)
- static const FloatT [default\\_sigma\\_hw\\_ratio](#)

### Friends

- template<class FloatT\_, class IdxT\_>  
std::ostream & [operator<<](#) (std::ostream &out, const [GaussFilter3D](#)< FloatT\_, IdxT\_ > &filt)

### 7.4.1 Detailed Description

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::DoGFilter3D< FloatT, IdxT >
```

Definition at line 150 of file GaussFilter.h.

## 7.4.2 Member Typedef Documentation

7.4.2.1 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::DoGFilter3D< FloatT, IdxT >::ImageT = arma::Cube<FloatT>`

Definition at line 155 of file GaussFilter.h.

7.4.2.2 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::DoGFilter3D< FloatT, IdxT >::IVecT = typename GaussFIRFilter<FloatT,IdxT>::IVecT`

Definition at line 153 of file GaussFilter.h.

7.4.2.3 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFIRFilter< FloatT, IdxT >::MatT = arma::Mat<FloatT> [inherited]`

Definition at line 26 of file GaussFilter.h.

7.4.2.4 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::DoGFilter3D< FloatT, IdxT >::VecT = typename GaussFIRFilter<FloatT,IdxT>::VecT`

Definition at line 154 of file GaussFilter.h.

## 7.4.3 Constructor &amp; Destructor Documentation

7.4.3.1 `template<class FloatT = float, class IdxT = uint32_t> boxxer::DoGFilter3D< FloatT, IdxT >::DoGFilter3D ( const IVecT & size, const VecT & sigma, FloatT sigma_ratio )`

7.4.3.2 `template<class FloatT = float, class IdxT = uint32_t> boxxer::DoGFilter3D< FloatT, IdxT >::DoGFilter3D ( const IVecT & size, const VecT & sigma, FloatT sigma_ratio, const IVecT & kernel_hw )`

## 7.4.4 Member Function Documentation

7.4.4.1 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Gauss_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

7.4.4.2 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Log_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

7.4.4.3 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::DoGFilter3D< FloatT, IdxT >::filter ( const ImageT & im, ImageT & out )`

7.4.4.4 `template<class FloatT = float, class IdxT = uint32_t> ImageT boxxer::DoGFilter3D< FloatT, IdxT >::make_image ( ) const [inline]`

Definition at line 163 of file GaussFilter.h.

References `boxxer::GaussFIRFilter< FloatT, IdxT >::size`.

7.4.4.5 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::DoGFilter3D< FloatT, IdxT >::set_kernel_hw ( const IVecT & kernel_half_width ) [virtual]`

Implements [boxxer::GaussFIRFilter< FloatT, IdxT >](#).

7.4.4.6 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::DoGFilter3D< FloatT, IdxT >::set_sigma_ratio ( FloatT sigma_ratio )`

7.4.4.7 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::DoGFilter3D< FloatT, IdxT >::test_filter ( const ImageT & im )`

#### 7.4.5 Friends And Related Function Documentation

7.4.5.1 `template<class FloatT = float, class IdxT = uint32_t> template<class FloatT_, class IdxT_> std::ostream& operator<< ( std::ostream & out, const GaussFilter3D< FloatT_, IdxT_> & filt ) [friend]`

#### 7.4.6 Member Data Documentation

7.4.6.1 `template<class FloatT = float, class IdxT = uint32_t> const FloatT boxxer::GaussFIRFilter< FloatT, IdxT >::default_sigma_hw_ratio [static], [protected], [inherited]`

Definition at line 41 of file GaussFilter.h.

7.4.6.2 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::dim [inherited]`

Definition at line 28 of file GaussFilter.h.

7.4.6.3 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::hw [inherited]`

Definition at line 31 of file GaussFilter.h.

7.4.6.4 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::max_kernel_hw [static], [protected], [inherited]`

Definition at line 40 of file GaussFilter.h.

7.4.6.5 `template<class FloatT = float, class IdxT = uint32_t> VecT boxxer::GaussFIRFilter< FloatT, IdxT >::sigma [inherited]`

Definition at line 30 of file GaussFilter.h.

7.4.6.6 `template<class FloatT = float, class IdxT = uint32_t> FloatT boxxer::DoGFilter3D< FloatT, IdxT >::sigma_ratio`

Definition at line 157 of file GaussFilter.h.

7.4.6.7 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::size`  
[inherited]

Definition at line 29 of file GaussFilter.h.

Referenced by `boxxer::GaussFilter2D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::LoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::GaussFilter3D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter3D< FloatT, IdxT >::make_image()`, and `boxxer::LoGFilter3D< FloatT, IdxT >::make_image()`.

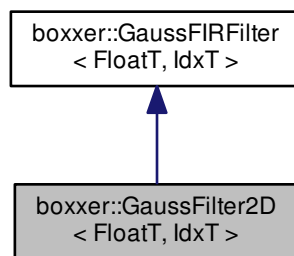
The documentation for this class was generated from the following file:

- [GaussFilter.h](#)

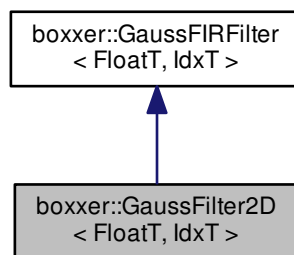
## 7.5 boxxer::GaussFilter2D< FloatT, IdxT > Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/GaussFilter.h>
```

Inheritance diagram for `boxxer::GaussFilter2D< FloatT, IdxT >`:



Collaboration diagram for `boxxer::GaussFilter2D< FloatT, IdxT >`:



### Public Types

- using `IVecT` = typename `GaussFIRFilter< FloatT, IdxT >::IVecT`
- using `VecT` = typename `GaussFIRFilter< FloatT, IdxT >::VecT`
- using `ImageT` = `arma::Mat< FloatT >`
- using `MatT` = `arma::Mat< FloatT >`

### Public Member Functions

- `GaussFilter2D` (const `IVecT` &`size`, const `VecT` &`sigma`)
- `GaussFilter2D` (const `IVecT` &`size`, const `VecT` &`sigma`, const `IVecT` &`kernel_hw`)
- void `set_kernel_hw` (const `IVecT` &`kernel_half_width`)
- `ImageT` `make_image` () const
- void `filter` (const `ImageT` &`im`, `ImageT` &`out`)
- void `test_filter` (const `ImageT` &`im`)

### Static Public Member Functions

- static `VecT` `compute_Gauss_FIR_kernel` (FloatT `sigma`, IdxT `hw`)
- static `VecT` `compute_Log_FIR_kernel` (FloatT `sigma`, IdxT `hw`)

### Public Attributes

- IdxT `dim`
- `IVecT` `size`
- `VecT` `sigma`
- `IVecT` `hw`

### Static Protected Attributes

- static const IdxT `max_kernel_hw`
- static const FloatT `default_sigma_hw_ratio`

### Friends

- template<class FloatT\_, class IdxT\_>  
std::ostream & `operator<<` (std::ostream &`out`, const `GaussFilter2D< FloatT_, IdxT_ >` &`filt`)

### 7.5.1 Detailed Description

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::GaussFilter2D< FloatT, IdxT >
```

#### 2D Filters

Definition at line 47 of file `GaussFilter.h`.



### 7.5.2 Member Typedef Documentation

**7.5.2.1** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFilter2D< FloatT, IdxT >::ImageT = arma::Mat<FloatT>`

Definition at line 52 of file GaussFilter.h.

**7.5.2.2** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFilter2D< FloatT, IdxT >::IVecT = typename GaussFIRFilter<FloatT,IdxT>::IVecT`

Definition at line 50 of file GaussFilter.h.

**7.5.2.3** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFIRFilter< FloatT, IdxT >::MatT = arma::Mat<FloatT> [inherited]`

Definition at line 26 of file GaussFilter.h.

**7.5.2.4** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFilter2D< FloatT, IdxT >::VecT = typename GaussFIRFilter<FloatT,IdxT>::VecT`

Definition at line 51 of file GaussFilter.h.

### 7.5.3 Constructor & Destructor Documentation

**7.5.3.1** `template<class FloatT = float, class IdxT = uint32_t> boxxer::GaussFilter2D< FloatT, IdxT >::GaussFilter2D ( const IVecT & size, const VecT & sigma )`

**7.5.3.2** `template<class FloatT = float, class IdxT = uint32_t> boxxer::GaussFilter2D< FloatT, IdxT >::GaussFilter2D ( const IVecT & size, const VecT & sigma, const IVecT & kernel_hw )`

### 7.5.4 Member Function Documentation

**7.5.4.1** `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Gauss_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

**7.5.4.2** `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Log_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

**7.5.4.3** `template<class FloatT = float, class IdxT = uint32_t> void boxxer::GaussFilter2D< FloatT, IdxT >::filter ( const ImageT & im, ImageT & out )`

**7.5.4.4** `template<class FloatT = float, class IdxT = uint32_t> ImageT boxxer::GaussFilter2D< FloatT, IdxT >::make_image ( )const [inline]`

Definition at line 57 of file GaussFilter.h.

References `boxxer::GaussFIRFilter< FloatT, IdxT >::size`.

7.5.4.5 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::GaussFilter2D< FloatT, IdxT >::set_kernel_hw ( const IVecT & kernel_half_width ) [virtual]`

Implements [boxxer::GaussFIRFilter< FloatT, IdxT >](#).

7.5.4.6 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::GaussFilter2D< FloatT, IdxT >::test_filter ( const ImageT & im )`

## 7.5.5 Friends And Related Function Documentation

7.5.5.1 `template<class FloatT = float, class IdxT = uint32_t> template<class FloatT_, class IdxT_> std::ostream& operator<< ( std::ostream & out, const GaussFilter2D< FloatT_, IdxT_> & filt ) [friend]`

## 7.5.6 Member Data Documentation

7.5.6.1 `template<class FloatT = float, class IdxT = uint32_t> const FloatT boxxer::GaussFIRFilter< FloatT, IdxT >::default_sigma_hw_ratio [static], [protected], [inherited]`

Definition at line 41 of file GaussFilter.h.

7.5.6.2 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::dim [inherited]`

Definition at line 28 of file GaussFilter.h.

7.5.6.3 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::hw [inherited]`

Definition at line 31 of file GaussFilter.h.

7.5.6.4 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::max_kernel_hw [static], [protected], [inherited]`

Definition at line 40 of file GaussFilter.h.

7.5.6.5 `template<class FloatT = float, class IdxT = uint32_t> VecT boxxer::GaussFIRFilter< FloatT, IdxT >::sigma [inherited]`

Definition at line 30 of file GaussFilter.h.

7.5.6.6 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::size [inherited]`

Definition at line 29 of file GaussFilter.h.

Referenced by `boxxer::GaussFilter2D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::LoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::GaussFilter3D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter3D< FloatT, IdxT >::make_image()`, and `boxxer::LoGFilter3D< FloatT, IdxT >::make_image()`.

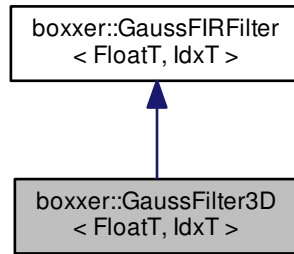
The documentation for this class was generated from the following file:

- [GaussFilter.h](#)

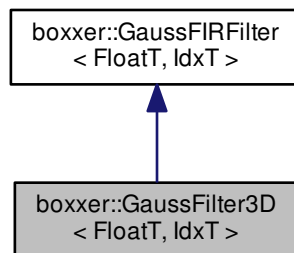
## 7.6 boxxer::GaussFilter3D&lt; FloatT, IdxT &gt; Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/GaussFilter.h>
```

Inheritance diagram for boxxer::GaussFilter3D< FloatT, IdxT >:



Collaboration diagram for boxxer::GaussFilter3D< FloatT, IdxT >:



## Public Types

- using [IVecT](#) = typename [GaussFIRFilter](#)< FloatT, IdxT >::IVecT
- using [VecT](#) = typename [GaussFIRFilter](#)< FloatT, IdxT >::VecT
- using [ImageT](#) = arma::Cube< FloatT >
- using [MatT](#) = arma::Mat< FloatT >

**Public Member Functions**

- [GaussFilter3D](#) (const [IVecT](#) &size, const [VecT](#) &sigma)
- [GaussFilter3D](#) (const [IVecT](#) &size, const [VecT](#) &sigma, const [IVecT](#) &kernel\_hw)
- void [set\\_kernel\\_hw](#) (const [IVecT](#) &kernel\_half\_width)
- [ImageT](#) [make\\_image](#) () const
- void [filter](#) (const [ImageT](#) &im, [ImageT](#) &out)
- void [test\\_filter](#) (const [ImageT](#) &im)

**Static Public Member Functions**

- static [VecT](#) [compute\\_Gauss\\_FIR\\_kernel](#) (FloatT sigma, IdxT hw)
- static [VecT](#) [compute\\_Log\\_FIR\\_kernel](#) (FloatT sigma, IdxT hw)

**Public Attributes**

- IdxT [dim](#)
- [IVecT](#) [size](#)
- [VecT](#) [sigma](#)
- [IVecT](#) [hw](#)

**Static Protected Attributes**

- static const IdxT [max\\_kernel\\_hw](#)
- static const FloatT [default\\_sigma\\_hw\\_ratio](#)

**Friends**

- template<class FloatT\_, class IdxT\_ >  
std::ostream & [operator<<](#) (std::ostream &out, const [GaussFilter3D](#)< FloatT\_, IdxT\_ > &filt)

**7.6.1 Detailed Description**

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::GaussFilter3D< FloatT, IdxT >
```

**3D Filters**

Definition at line 126 of file GaussFilter.h.

**7.6.2 Member Typedef Documentation**

**7.6.2.1** template<class FloatT = float, class IdxT = uint32\_t> using boxxer::GaussFilter3D< FloatT, IdxT >::ImageT = arma::Cube<FloatT>

Definition at line 131 of file GaussFilter.h.

7.6.2.2 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFilter3D< FloatT, IdxT >::IVecT = typename GaussFIRFilter<FloatT,IdxT>::IVecT`

Definition at line 129 of file GaussFilter.h.

7.6.2.3 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFIRFilter< FloatT, IdxT >::MatT = arma::Mat<FloatT> [inherited]`

Definition at line 26 of file GaussFilter.h.

7.6.2.4 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFilter3D< FloatT, IdxT >::VecT = typename GaussFIRFilter<FloatT,IdxT>::VecT`

Definition at line 130 of file GaussFilter.h.

### 7.6.3 Constructor & Destructor Documentation

7.6.3.1 `template<class FloatT = float, class IdxT = uint32_t> boxxer::GaussFilter3D< FloatT, IdxT >::GaussFilter3D ( const IVecT & size, const VecT & sigma )`

7.6.3.2 `template<class FloatT = float, class IdxT = uint32_t> boxxer::GaussFilter3D< FloatT, IdxT >::GaussFilter3D ( const IVecT & size, const VecT & sigma, const IVecT & kernel_hw )`

### 7.6.4 Member Function Documentation

7.6.4.1 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Gauss_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

7.6.4.2 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Log_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

7.6.4.3 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::GaussFilter3D< FloatT, IdxT >::filter ( const ImageT & im, ImageT & out )`

7.6.4.4 `template<class FloatT = float, class IdxT = uint32_t> ImageT boxxer::GaussFilter3D< FloatT, IdxT >::make_image ( )const [inline]`

Definition at line 136 of file GaussFilter.h.

References `boxxer::GaussFIRFilter< FloatT, IdxT >::size`.

7.6.4.5 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::GaussFilter3D< FloatT, IdxT >::set_kernel_hw ( const IVecT & kernel_half_width ) [virtual]`

Implements [boxxer::GaussFIRFilter< FloatT, IdxT >](#).

7.6.4.6 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::GaussFilter3D< FloatT, IdxT >::test_filter ( const ImageT & im )`

## 7.6.5 Friends And Related Function Documentation

7.6.5.1 `template<class FloatT = float, class IdxT = uint32_t> template<class FloatT_, class IdxT_ > std::ostream& operator<< ( std::ostream & out, const GaussFilter3D< FloatT_, IdxT_ > & filt ) [friend]`

## 7.6.6 Member Data Documentation

7.6.6.1 `template<class FloatT = float, class IdxT = uint32_t> const FloatT boxxer::GaussFIRFilter< FloatT, IdxT >::default_sigma_hw_ratio [static], [protected], [inherited]`

Definition at line 41 of file GaussFilter.h.

7.6.6.2 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::dim [inherited]`

Definition at line 28 of file GaussFilter.h.

7.6.6.3 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::hw [inherited]`

Definition at line 31 of file GaussFilter.h.

7.6.6.4 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::max_kernel_hw [static], [protected], [inherited]`

Definition at line 40 of file GaussFilter.h.

7.6.6.5 `template<class FloatT = float, class IdxT = uint32_t> VecT boxxer::GaussFIRFilter< FloatT, IdxT >::sigma [inherited]`

Definition at line 30 of file GaussFilter.h.

7.6.6.6 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::size [inherited]`

Definition at line 29 of file GaussFilter.h.

Referenced by `boxxer::GaussFilter2D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::LoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::GaussFilter3D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter3D< FloatT, IdxT >::make_image()`, and `boxxer::LoGFilter3D< FloatT, IdxT >::make_image()`.

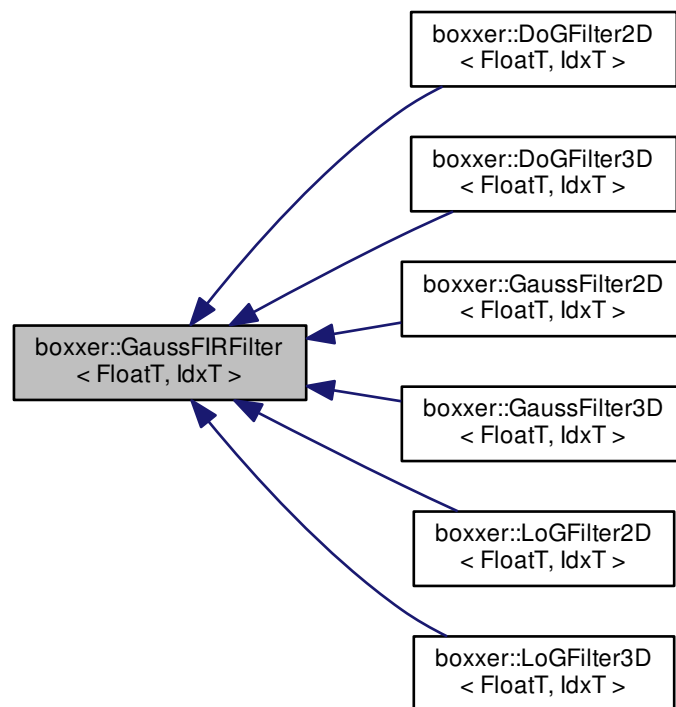
The documentation for this class was generated from the following file:

- [GaussFilter.h](#)

## 7.7 boxxer::GaussFIRFilter&lt; FloatT, IdxT &gt; Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/GaussFilter.h>
```

Inheritance diagram for boxxer::GaussFIRFilter< FloatT, IdxT >:



## Public Types

- using [IVecT](#) = arma::Col< IdxT >
- using [VecT](#) = arma::Col< FloatT >
- using [MatT](#) = arma::Mat< FloatT >

## Public Member Functions

- [GaussFIRFilter](#) (IdxT [dim](#), const [IVecT](#) &[size](#), const [VecT](#) &[sigma](#))
- virtual void [set\\_kernel\\_hw](#) (const [IVecT](#) &kernel\_half\_width)=0

## Static Public Member Functions

- static [VecT](#) [compute\\_Gauss\\_FIR\\_kernel](#) (FloatT [sigma](#), IdxT [hw](#))
- static [VecT](#) [compute\\_Log\\_FIR\\_kernel](#) (FloatT [sigma](#), IdxT [hw](#))

## Public Attributes

- `IdxT dim`
- `IVecT size`
- `VecT sigma`
- `IVecT hw`

## Static Protected Attributes

- static const `IdxT max_kernel_hw`
- static const `FloatT default_sigma_hw_ratio`

### 7.7.1 Detailed Description

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::GaussFIRFilter< FloatT, IdxT >
```

#### Base filters

Definition at line 21 of file GaussFilter.h.

### 7.7.2 Member Typedef Documentation

**7.7.2.1** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFIRFilter< FloatT, IdxT >::IVecT = arma::Col<IdxT>`

Definition at line 24 of file GaussFilter.h.

**7.7.2.2** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFIRFilter< FloatT, IdxT >::MatT = arma::Mat<FloatT>`

Definition at line 26 of file GaussFilter.h.

**7.7.2.3** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFIRFilter< FloatT, IdxT >::VecT = arma::Col<FloatT>`

Definition at line 25 of file GaussFilter.h.



## 7.7.3 Constructor &amp; Destructor Documentation

7.7.3.1 `template<class FloatT = float, class IdxT = uint32_t> boxxer::GaussFIRFilter< FloatT, IdxT >::GaussFIRFilter ( IdxT dim, const IVecT & size, const VecT & sigma )`

## 7.7.4 Member Function Documentation

7.7.4.1 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Gauss_FIR_kernel ( FloatT sigma, IdxT hw ) [static]`

7.7.4.2 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Log_FIR_kernel ( FloatT sigma, IdxT hw ) [static]`

7.7.4.3 `template<class FloatT = float, class IdxT = uint32_t> virtual void boxxer::GaussFIRFilter< FloatT, IdxT >::set_kernel_hw ( const IVecT & kernel_half_width ) [pure virtual]`

Implemented in [boxxer::LoGFilter3D< FloatT, IdxT >](#), [boxxer::DoGFilter3D< FloatT, IdxT >](#), [boxxer::GaussFilter3D< FloatT, IdxT >](#), [boxxer::LoGFilter2D< FloatT, IdxT >](#), [boxxer::DoGFilter2D< FloatT, IdxT >](#), and [boxxer::GaussFilter2D< FloatT, IdxT >](#).

## 7.7.5 Member Data Documentation

7.7.5.1 `template<class FloatT = float, class IdxT = uint32_t> const FloatT boxxer::GaussFIRFilter< FloatT, IdxT >::default_sigma_hw_ratio [static], [protected]`

Definition at line 41 of file GaussFilter.h.

7.7.5.2 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::dim`

Definition at line 28 of file GaussFilter.h.

7.7.5.3 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::hw`

Definition at line 31 of file GaussFilter.h.

7.7.5.4 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::max_kernel_hw [static], [protected]`

Definition at line 40 of file GaussFilter.h.

7.7.5.5 `template<class FloatT = float, class IdxT = uint32_t> VecT boxxer::GaussFIRFilter< FloatT, IdxT >::sigma`

Definition at line 30 of file GaussFilter.h.

#### 7.7.5.6 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::size`

Definition at line 29 of file GaussFilter.h.

Referenced by `boxxer::GaussFilter2D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::LoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::GaussFilter3D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter3D< FloatT, IdxT >::make_image()`, and `boxxer::LoGFilter3D< FloatT, IdxT >::make_image()`.

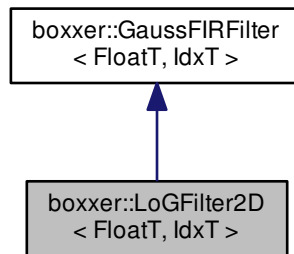
The documentation for this class was generated from the following file:

- [GaussFilter.h](#)

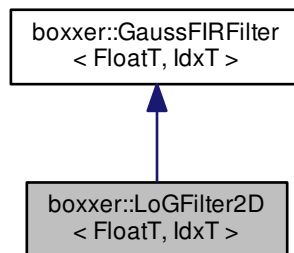
### 7.8 `boxxer::LoGFilter2D< FloatT, IdxT >` Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/GaussFilter.h>
```

Inheritance diagram for `boxxer::LoGFilter2D< FloatT, IdxT >`:



Collaboration diagram for `boxxer::LoGFilter2D< FloatT, IdxT >`:



### Public Types

- using [IVecT](#) = typename [GaussFIRFilter](#)< FloatT, IdxT >::IVecT
- using [VecT](#) = typename [GaussFIRFilter](#)< FloatT, IdxT >::VecT
- using [ImageT](#) = arma::Mat< FloatT >
- using [MatT](#) = arma::Mat< FloatT >

### Public Member Functions

- [LoGFilter2D](#) (const [IVecT](#) &size, const [VecT](#) &sigma)
- [LoGFilter2D](#) (const [IVecT](#) &size, const [VecT](#) &sigma, const [IVecT](#) &kernel\_hw)
- void [set\\_kernel\\_hw](#) (const [IVecT](#) &kernel\_half\_width)
- [ImageT](#) [make\\_image](#) () const
- void [filter](#) (const [ImageT](#) &im, [ImageT](#) &out)
- void [test\\_filter](#) (const [ImageT](#) &im)

### Static Public Member Functions

- static [VecT](#) [compute\\_Gauss\\_FIR\\_kernel](#) (FloatT sigma, IdxT hw)
- static [VecT](#) [compute\\_Log\\_FIR\\_kernel](#) (FloatT sigma, IdxT hw)

### Public Attributes

- IdxT [dim](#)
- [IVecT](#) [size](#)
- [VecT](#) [sigma](#)
- [IVecT](#) [hw](#)

### Static Protected Attributes

- static const IdxT [max\\_kernel\\_hw](#)
- static const FloatT [default\\_sigma\\_hw\\_ratio](#)

### Friends

- template<class FloatT\_, class IdxT\_>  
std::ostream & [operator<<](#) (std::ostream &out, const [LoGFilter2D](#)< FloatT\_, IdxT\_ > &filt)

#### 7.8.1 Detailed Description

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::LoGFilter2D< FloatT, IdxT >
```

Definition at line 98 of file GaussFilter.h.

## 7.8.2 Member Typedef Documentation

**7.8.2.1** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::LoGFilter2D< FloatT, IdxT >::ImageT = arma::Mat<FloatT>`

Definition at line 103 of file GaussFilter.h.

**7.8.2.2** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::LoGFilter2D< FloatT, IdxT >::IVecT = typename GaussFIRFilter<FloatT,IdxT>::IVecT`

Definition at line 101 of file GaussFilter.h.

**7.8.2.3** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFIRFilter< FloatT, IdxT >::MatT = arma::Mat<FloatT> [inherited]`

Definition at line 26 of file GaussFilter.h.

**7.8.2.4** `template<class FloatT = float, class IdxT = uint32_t> using boxxer::LoGFilter2D< FloatT, IdxT >::VecT = typename GaussFIRFilter<FloatT,IdxT>::VecT`

Definition at line 102 of file GaussFilter.h.

## 7.8.3 Constructor & Destructor Documentation

**7.8.3.1** `template<class FloatT = float, class IdxT = uint32_t> boxxer::LoGFilter2D< FloatT, IdxT >::LoGFilter2D ( const IVecT & size, const VecT & sigma )`

**7.8.3.2** `template<class FloatT = float, class IdxT = uint32_t> boxxer::LoGFilter2D< FloatT, IdxT >::LoGFilter2D ( const IVecT & size, const VecT & sigma, const IVecT & kernel_hw )`

## 7.8.4 Member Function Documentation

**7.8.4.1** `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Gauss_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

**7.8.4.2** `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Log_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

**7.8.4.3** `template<class FloatT = float, class IdxT = uint32_t> void boxxer::LoGFilter2D< FloatT, IdxT >::filter ( const ImageT & im, ImageT & out )`

**7.8.4.4** `template<class FloatT = float, class IdxT = uint32_t> ImageT boxxer::LoGFilter2D< FloatT, IdxT >::make_image ( ) const [inline]`

Definition at line 108 of file GaussFilter.h.

References `boxxer::GaussFIRFilter< FloatT, IdxT >::size`.

7.8.4.5 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::LoGFilter2D< FloatT, IdxT >::set_kernel_hw ( const IVecT & kernel_half_width ) [virtual]`

Implements [boxxer::GaussFIRFilter< FloatT, IdxT >](#).

7.8.4.6 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::LoGFilter2D< FloatT, IdxT >::test_filter ( const ImageT & im )`

## 7.8.5 Friends And Related Function Documentation

7.8.5.1 `template<class FloatT = float, class IdxT = uint32_t> template<class FloatT_, class IdxT_> std::ostream& operator<< ( std::ostream & out, const LoGFilter2D< FloatT_, IdxT_> & filt ) [friend]`

## 7.8.6 Member Data Documentation

7.8.6.1 `template<class FloatT = float, class IdxT = uint32_t> const FloatT boxxer::GaussFIRFilter< FloatT, IdxT >::default_sigma_hw_ratio [static], [protected], [inherited]`

Definition at line 41 of file GaussFilter.h.

7.8.6.2 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::dim [inherited]`

Definition at line 28 of file GaussFilter.h.

7.8.6.3 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::hw [inherited]`

Definition at line 31 of file GaussFilter.h.

7.8.6.4 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::max_kernel_hw [static], [protected], [inherited]`

Definition at line 40 of file GaussFilter.h.

7.8.6.5 `template<class FloatT = float, class IdxT = uint32_t> VecT boxxer::GaussFIRFilter< FloatT, IdxT >::sigma [inherited]`

Definition at line 30 of file GaussFilter.h.

7.8.6.6 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::size [inherited]`

Definition at line 29 of file GaussFilter.h.

Referenced by `boxxer::GaussFilter2D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::LoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::GaussFilter3D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter3D< FloatT, IdxT >::make_image()`, and `boxxer::LoGFilter3D< FloatT, IdxT >::make_image()`.

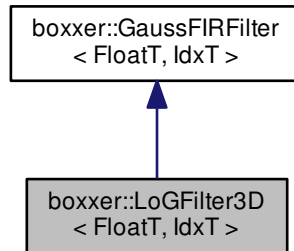
The documentation for this class was generated from the following file:

- [GaussFilter.h](#)

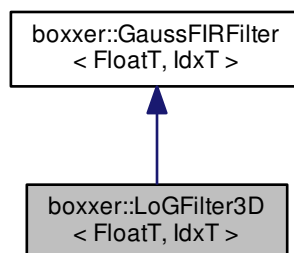
## 7.9 boxxer::LoGFilter3D< FloatT, IdxT > Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/GaussFilter.h>
```

Inheritance diagram for boxxer::LoGFilter3D< FloatT, IdxT >:



Collaboration diagram for boxxer::LoGFilter3D< FloatT, IdxT >:



### Public Types

- using [IVecT](#) = typename [GaussFIRFilter](#)< FloatT, IdxT >::IVecT
- using [VecT](#) = typename [GaussFIRFilter](#)< FloatT, IdxT >::VecT
- using [ImageT](#) = arma::Cube< FloatT >
- using [MatT](#) = arma::Mat< FloatT >

### Public Member Functions

- [LoGFilter3D](#) (const [IVecT](#) &size, const [VecT](#) &sigma)
- [LoGFilter3D](#) (const [IVecT](#) &size, const [VecT](#) &sigma, const [IVecT](#) &kernel\_hw)
- void [set\\_kernel\\_hw](#) (const [IVecT](#) &kernel\_half\_width)
- [ImageT](#) [make\\_image](#) () const
- void [filter](#) (const [ImageT](#) &im, [ImageT](#) &out)
- void [test\\_filter](#) (const [ImageT](#) &im)

### Static Public Member Functions

- static [VecT](#) [compute\\_Gauss\\_FIR\\_kernel](#) (FloatT sigma, IdxT hw)
- static [VecT](#) [compute\\_Log\\_FIR\\_kernel](#) (FloatT sigma, IdxT hw)

### Public Attributes

- IdxT [dim](#)
- [IVecT](#) [size](#)
- [VecT](#) [sigma](#)
- [IVecT](#) [hw](#)

### Static Protected Attributes

- static const IdxT [max\\_kernel\\_hw](#)
- static const FloatT [default\\_sigma\\_hw\\_ratio](#)

### Friends

- template<class FloatT\_, class IdxT\_>  
std::ostream & [operator<<](#) (std::ostream &out, const [LoGFilter3D](#)< FloatT\_, IdxT\_ > &filt)

#### 7.9.1 Detailed Description

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::LoGFilter3D< FloatT, IdxT >
```

Definition at line 178 of file GaussFilter.h.

#### 7.9.2 Member Typedef Documentation

7.9.2.1 template<class FloatT = float, class IdxT = uint32\_t> using boxxer::LoGFilter3D< FloatT, IdxT >::ImageT = arma::Cube<FloatT>

Definition at line 183 of file GaussFilter.h.

7.9.2.2 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::LoGFilter3D< FloatT, IdxT >::IVecT = typename GaussFIRFilter<FloatT,IdxT>::IVecT`

Definition at line 181 of file GaussFilter.h.

7.9.2.3 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::GaussFIRFilter< FloatT, IdxT >::MatT = arma::Mat<FloatT> [inherited]`

Definition at line 26 of file GaussFilter.h.

7.9.2.4 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::LoGFilter3D< FloatT, IdxT >::VecT = typename GaussFIRFilter<FloatT,IdxT>::VecT`

Definition at line 182 of file GaussFilter.h.

### 7.9.3 Constructor & Destructor Documentation

7.9.3.1 `template<class FloatT = float, class IdxT = uint32_t> boxxer::LoGFilter3D< FloatT, IdxT >::LoGFilter3D ( const IVecT & size, const VecT & sigma )`

7.9.3.2 `template<class FloatT = float, class IdxT = uint32_t> boxxer::LoGFilter3D< FloatT, IdxT >::LoGFilter3D ( const IVecT & size, const VecT & sigma, const IVecT & kernel_hw )`

### 7.9.4 Member Function Documentation

7.9.4.1 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Gauss_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

7.9.4.2 `template<class FloatT = float, class IdxT = uint32_t> static VecT boxxer::GaussFIRFilter< FloatT, IdxT >::compute_Log_FIR_kernel ( FloatT sigma, IdxT hw ) [static],[inherited]`

7.9.4.3 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::LoGFilter3D< FloatT, IdxT >::filter ( const ImageT & im, ImageT & out )`

7.9.4.4 `template<class FloatT = float, class IdxT = uint32_t> ImageT boxxer::LoGFilter3D< FloatT, IdxT >::make_image ( ) const [inline]`

Definition at line 188 of file GaussFilter.h.

References `boxxer::GaussFIRFilter< FloatT, IdxT >::size`.

7.9.4.5 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::LoGFilter3D< FloatT, IdxT >::set_kernel_hw ( const IVecT & kernel_half_width ) [virtual]`

Implements [boxxer::GaussFIRFilter< FloatT, IdxT >](#).



7.9.4.6 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::LoGFilter3D< FloatT, IdxT >::test_filter ( const ImageT & im )`

## 7.9.5 Friends And Related Function Documentation

7.9.5.1 `template<class FloatT = float, class IdxT = uint32_t> template<class FloatT_, class IdxT_ > std::ostream& operator<< ( std::ostream & out, const LoGFilter3D< FloatT_, IdxT_ > & filt )` [*friend*]

## 7.9.6 Member Data Documentation

7.9.6.1 `template<class FloatT = float, class IdxT = uint32_t> const FloatT boxxer::GaussFIRFilter< FloatT, IdxT >::default_sigma_hw_ratio` [*static*], [*protected*], [*inherited*]

Definition at line 41 of file GaussFilter.h.

7.9.6.2 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::dim` [*inherited*]

Definition at line 28 of file GaussFilter.h.

7.9.6.3 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::hw` [*inherited*]

Definition at line 31 of file GaussFilter.h.

7.9.6.4 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::GaussFIRFilter< FloatT, IdxT >::max_kernel_hw` [*static*], [*protected*], [*inherited*]

Definition at line 40 of file GaussFilter.h.

7.9.6.5 `template<class FloatT = float, class IdxT = uint32_t> VecT boxxer::GaussFIRFilter< FloatT, IdxT >::sigma` [*inherited*]

Definition at line 30 of file GaussFilter.h.

7.9.6.6 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::GaussFIRFilter< FloatT, IdxT >::size` [*inherited*]

Definition at line 29 of file GaussFilter.h.

Referenced by `boxxer::GaussFilter2D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::LoGFilter2D< FloatT, IdxT >::make_image()`, `boxxer::GaussFilter3D< FloatT, IdxT >::make_image()`, `boxxer::DoGFilter3D< FloatT, IdxT >::make_image()`, and `boxxer::LoGFilter3D< FloatT, IdxT >::make_image()`.

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

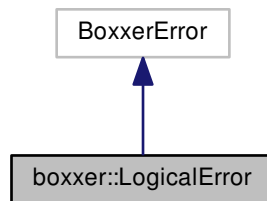
- [GaussFilter.h](#)

## 7.10 boxxer::LogicalError Struct Reference

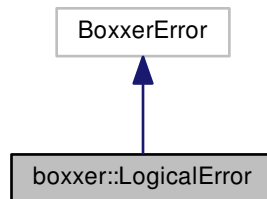
Internal logical error. Bad logic or broken promises.

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/BoxxerError.h>
```

Inheritance diagram for boxxer::LogicalError:



Collaboration diagram for boxxer::LogicalError:



### Public Member Functions

- [LogicalError](#) (std::string message)

#### 7.10.1 Detailed Description

Internal logical error. Bad logic or broken promises.

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

## 7.10.2 Constructor &amp; Destructor Documentation

7.10.2.1 boxxer::LogicalError::LogicalError ( std::string *message* ) [inline]

Definition at line 34 of file BoxxerError.h.

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

- [BoxxerError.h](#)

## 7.11 boxxer::Maxima2D&lt; FloatT, IdxT &gt; Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/Maxima.h>
```

## Public Types

- using [IVecT](#) = arma::Col< IdxT >
- using [IMatT](#) = arma::Mat< IdxT >
- using [VecT](#) = arma::Col< FloatT >
- using [ImageT](#) = arma::Mat< FloatT >

## Public Member Functions

- [Maxima2D](#) (const [IVecT](#) &sizeX, IdxT [boxsize](#)=[MinBoxsize](#))
- IdxT [find\\_maxima](#) (const [ImageT](#) &im)
- IdxT [find\\_maxima](#) (const [ImageT](#) &im, [IMatT](#) &maxima\_out, [VecT](#) &max\_vals\_out)
- void [read\\_maxima](#) (IdxT Nmaxima, [IMatT](#) &maxima\_out, [VecT](#) &max\_vals\_out) const
- void [test\\_maxima](#) (const [ImageT](#) &im)
- bool [check\\_maxima](#) (const [ImageT](#) &im, IdxT x, IdxT y, IdxT neighborhoodSize=[MinBoxsize](#))

## Public Attributes

- [IVecT](#) [size](#)
- IdxT [boxsize](#)

## Static Public Attributes

- static const IdxT [MinBoxsize](#)
- static const IdxT [Ndim](#)

## 7.11.1 Detailed Description

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::Maxima2D< FloatT, IdxT >
```

Definition at line 16 of file Maxima.h.

### 7.11.2 Member Typedef Documentation

7.11.2.1 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Maxima2D< FloatT, IdxT >::ImageT = arma::Mat<FloatT>`

Definition at line 22 of file Maxima.h.

7.11.2.2 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Maxima2D< FloatT, IdxT >::IMatT = arma::Mat<IdxT>`

Definition at line 20 of file Maxima.h.

7.11.2.3 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Maxima2D< FloatT, IdxT >::IVecT = arma::Col<IdxT>`

Definition at line 19 of file Maxima.h.

7.11.2.4 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Maxima2D< FloatT, IdxT >::VecT = arma::Col<FloatT>`

Definition at line 21 of file Maxima.h.

### 7.11.3 Constructor & Destructor Documentation

7.11.3.1 `template<class FloatT = float, class IdxT = uint32_t> boxxer::Maxima2D< FloatT, IdxT >::Maxima2D ( const IVecT & sizeX, IdxT boxsize = MinBoxsize )`

### 7.11.4 Member Function Documentation

7.11.4.1 `template<class FloatT = float, class IdxT = uint32_t> bool boxxer::Maxima2D< FloatT, IdxT >::check_maxima ( const ImageT & im, IdxT x, IdxT y, IdxT neighborhoodSize = MinBoxsize )`

7.11.4.2 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Maxima2D< FloatT, IdxT >::find_maxima ( const ImageT & im )`

7.11.4.3 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Maxima2D< FloatT, IdxT >::find_maxima ( const ImageT & im, IMatT & maxima_out, VecT & max_vals_out )`

7.11.4.4 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Maxima2D< FloatT, IdxT >::read_maxima ( IdxT Nmaxima, IMatT & maxima_out, VecT & max_vals_out ) const`

7.11.4.5 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Maxima2D< FloatT, IdxT >::test_maxima ( const ImageT & im )`

### 7.11.5 Member Data Documentation

7.11.5.1 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Maxima2D< FloatT, IdxT >::boxsize`

Definition at line 27 of file Maxima.h.

7.11.5.2 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::Maxima2D< FloatT, IdxT >::MinBoxsize`  
`[static]`

Definition at line 23 of file Maxima.h.

7.11.5.3 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::Maxima2D< FloatT, IdxT >::Ndim`  
`[static]`

Definition at line 24 of file Maxima.h.

7.11.5.4 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::Maxima2D< FloatT, IdxT >::size`

Definition at line 26 of file Maxima.h.

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

- [Maxima.h](#)

## 7.12 boxxer::Maxima3D< FloatT, IdxT > Class Template Reference

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/Maxima.h>
```

### Public Types

- using [IVecT](#) = arma::Col< IdxT >
- using [IMatT](#) = arma::Mat< IdxT >
- using [ICubeT](#) = arma::Cube< IdxT >
- using [VecT](#) = arma::Col< FloatT >
- using [ImageT](#) = arma::Cube< FloatT >

### Public Member Functions

- [Maxima3D](#) (const [IVecT](#) &size, IdxT boxsize=[MinBoxsize](#))
- IdxT [find\\_maxima](#) (const [ImageT](#) &im)
- IdxT [find\\_maxima](#) (const [ImageT](#) &im, [IMatT](#) &maxima\_out, [VecT](#) &max\_vals\_out)
- void [read\\_maxima](#) ([IMatT](#) &maxima\_out, [VecT](#) &max\_vals\_out) const
- void [test\\_maxima](#) (const [ImageT](#) &im)
- bool [check\\_maxima](#) (const [ImageT](#) &im, IdxT x, IdxT y, IdxT z, IdxT neighborhoodSize=[MinBoxsize](#))

### Public Attributes

- [IVecT](#) size
- IdxT boxsize

## Static Public Attributes

- static const IdxT [MinBoxsize](#)
- static const IdxT [Ndim](#)

### 7.12.1 Detailed Description

```
template<class FloatT = float, class IdxT = uint32_t>
class boxxer::Maxima3D< FloatT, IdxT >
```

Definition at line 52 of file Maxima.h.

### 7.12.2 Member Typedef Documentation

7.12.2.1 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Maxima3D< FloatT, IdxT >::ICubeT = arma::Cube<IdxT>`

Definition at line 57 of file Maxima.h.

7.12.2.2 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Maxima3D< FloatT, IdxT >::ImageT = arma::Cube<FloatT>`

Definition at line 59 of file Maxima.h.

7.12.2.3 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Maxima3D< FloatT, IdxT >::IMatT = arma::Mat<IdxT>`

Definition at line 56 of file Maxima.h.

7.12.2.4 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Maxima3D< FloatT, IdxT >::IVecT = arma::Col<IdxT>`

Definition at line 55 of file Maxima.h.

7.12.2.5 `template<class FloatT = float, class IdxT = uint32_t> using boxxer::Maxima3D< FloatT, IdxT >::VecT = arma::Col<FloatT>`

Definition at line 58 of file Maxima.h.

## 7.12.3 Constructor &amp; Destructor Documentation

7.12.3.1 `template<class FloatT = float, class IdxT = uint32_t> boxxer::Maxima3D< FloatT, IdxT >::Maxima3D ( const IVecT & size, IdxT boxsize = MinBoxsize )`

## 7.12.4 Member Function Documentation

7.12.4.1 `template<class FloatT = float, class IdxT = uint32_t> bool boxxer::Maxima3D< FloatT, IdxT >::check_maxima ( const ImageT & im, IdxT x, IdxT y, IdxT z, IdxT neighborhoodSize = MinBoxsize )`

7.12.4.2 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Maxima3D< FloatT, IdxT >::find_maxima ( const ImageT & im )`

7.12.4.3 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Maxima3D< FloatT, IdxT >::find_maxima ( const ImageT & im, IMatT & maxima_out, VecT & max_vals_out )`

7.12.4.4 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Maxima3D< FloatT, IdxT >::read_maxima ( IMatT & maxima_out, VecT & max_vals_out ) const`

7.12.4.5 `template<class FloatT = float, class IdxT = uint32_t> void boxxer::Maxima3D< FloatT, IdxT >::test_maxima ( const ImageT & im )`

## 7.12.5 Member Data Documentation

7.12.5.1 `template<class FloatT = float, class IdxT = uint32_t> IdxT boxxer::Maxima3D< FloatT, IdxT >::boxsize`

Definition at line 65 of file Maxima.h.

7.12.5.2 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::Maxima3D< FloatT, IdxT >::MinBoxsize [static]`

Definition at line 61 of file Maxima.h.

7.12.5.3 `template<class FloatT = float, class IdxT = uint32_t> const IdxT boxxer::Maxima3D< FloatT, IdxT >::Ndim [static]`

Definition at line 62 of file Maxima.h.

7.12.5.4 `template<class FloatT = float, class IdxT = uint32_t> IVecT boxxer::Maxima3D< FloatT, IdxT >::size`

Definition at line 64 of file Maxima.h.

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

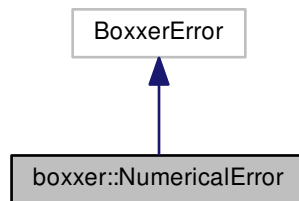
- [Maxima.h](#)

### 7.13 boxer::NumericalError Struct Reference

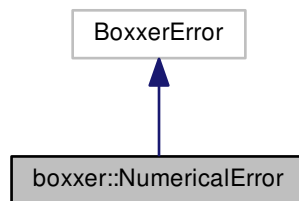
Internal numerical error.

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/BoxxerError.h>
```

Inheritance diagram for boxer::NumericalError:



Collaboration diagram for boxer::NumericalError:



#### Public Member Functions

- [NumericalError](#) (std::string message)

#### 7.13.1 Detailed Description

Internal numerical error.

Definition at line 39 of file BoxxerError.h.



## 7.13.2 Constructor &amp; Destructor Documentation

## 7.13.2.1 boxxer::NumericalError::NumericalError( std::string message ) [inline]

Definition at line 41 of file BoxxerError.h.

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

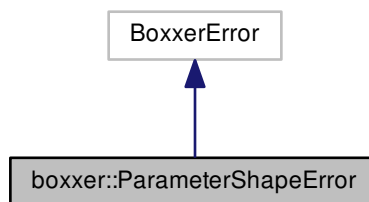
- [BoxxerError.h](#)

## 7.14 boxxer::ParameterShapeError Struct Reference

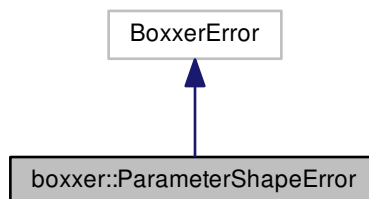
Parameters are the incorrect shape, size or number of dimensions.

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/BoxxerError.h>
```

Inheritance diagram for boxxer::ParameterShapeError:



Collaboration diagram for boxxer::ParameterShapeError:



## Public Member Functions

- [ParameterShapeError](#) (std::string message)

### 7.14.1 Detailed Description

Parameters are the incorrect shape, size or number of dimensions.

Definition at line 25 of file BoxxerError.h.

### 7.14.2 Constructor & Destructor Documentation

#### 7.14.2.1 `boxxer::ParameterShapeError::ParameterShapeError ( std::string message ) [inline]`

Definition at line 27 of file BoxxerError.h.

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

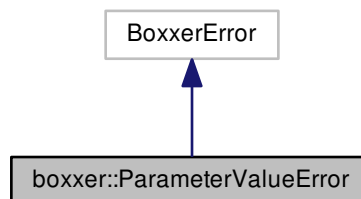
- [BoxxerError.h](#)

## 7.15 `boxxer::ParameterValueError` Struct Reference

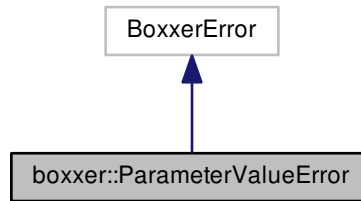
Parameter value is not valid.

```
#include </home/travis/build/markjolah/Boxxer/include/Boxxer/BoxxerError.h>
```

Inheritance diagram for `boxxer::ParameterValueError`:



Collaboration diagram for boxxer::ParameterValueError:



#### Public Member Functions

- [ParameterValueError](#) (std::string message)

##### 7.15.1 Detailed Description

Parameter value is not valid.

Definition at line 18 of file BoxxerError.h.

##### 7.15.2 Constructor & Destructor Documentation

###### 7.15.2.1 boxxer::ParameterValueError::ParameterValue ( std::string *message* ) `[inline]`

Definition at line 20 of file BoxxerError.h.

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

- [BoxxerError.h](#)

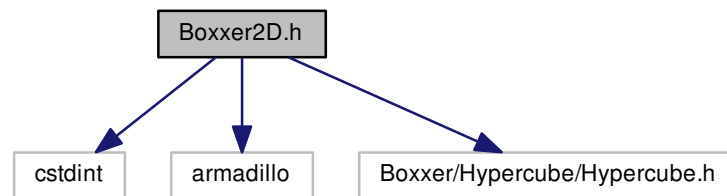
## 8 File Documentation

### 8.1 Boxxer2D.h File Reference

The class declaration for Boxxer2D.

```
#include <cstdint>
#include <armadillo>
#include "Boxxer/Hypercube/Hypercube.h"
```

Include dependency graph for Boxxer2D.h:



#### Classes

- class [boxxer::Boxxer2D< FloatT, IdxT >](#)

#### Namespaces

- [boxxer](#)

#### 8.1.1 Detailed Description

The class declaration for Boxxer2D.

#### Author

Mark J. Olah (mjo@cs.unm DOT edu)

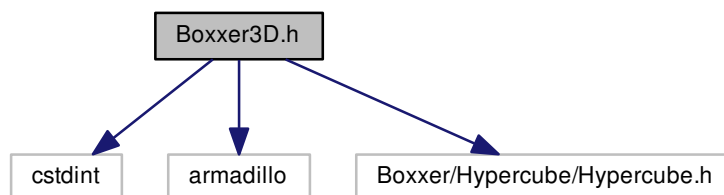
#### Date

2014-2019

## 8.2 Boxxer3D.h File Reference

The class declaration for Boxxer3D.

```
#include <stdint>
#include <armadillo>
#include "Boxxer/Hypercube/Hypercube.h"
Include dependency graph for Boxxer3D.h:
```



### Classes

- class [boxxer::Boxxer3D< FloatT, IdxT >](#)

### Namespaces

- [boxxer](#)

### 8.2.1 Detailed Description

The class declaration for Boxxer3D.

### Author

Mark J. Olah ([mjo@cs.unm.edu](mailto:mjo@cs.unm.edu))

### Date

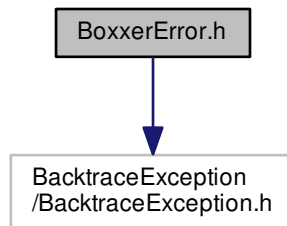
2014-2019

### 8.3 BoxxerError.h File Reference

Error handling.

```
#include "BacktraceException/BacktraceException.h"
```

Include dependency graph for BoxxerError.h:



#### Classes

- struct [boxxer::ParameterValueError](#)  
*Parameter value is not valid.*
- struct [boxxer::ParameterShapeError](#)  
*Parameters are the incorrect shape, size or number of dimensions.*
- struct [boxxer::LogicalError](#)  
*Internal logical error. Bad logic or broken promises.*
- struct [boxxer::NumericalError](#)  
*Internal numerical error.*

#### Namespaces

- [boxxer](#)

#### Typedefs

- using [boxxer::BoxxerError](#) = [backtrace\\_exception::BacktraceException](#)

#### 8.3.1 Detailed Description

Error handling.

#### Author

Mark J. Olah (mjo@cs.unm DOT edu)

#### Date

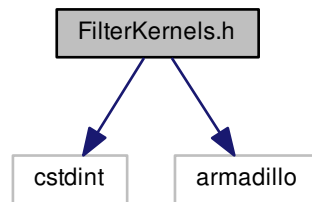
2014-2019

## 8.4 FilterKernels.h File Reference

The `boxxer::kernels` namespace - low-level Gaussian finite-impulse response filters.

```
#include <cstdlib>
#include <armadillo>
```

Include dependency graph for FilterKernels.h:



### Namespaces

- `boxxer`
- `boxxer::kernels`

### Functions

#### 1D Gauss FIR Filters

*1D Gaussian finite-impulse response filters.*

- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_1D (IntT size, const FloatT data[], FloatT fdata[], IntT hw, const FloatT kernel[])`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_1D (const arma::Col< FloatT > &data, arma::Col< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_1D_small (IntT size, const FloatT data[], FloatT fdata[], IntT hw, const FloatT kernel[])`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_1D_arma (const arma::Col< FloatT > &data, arma::Col< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_1D_inplace_arma (arma::Col< FloatT > &data, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_1D_inplace (IntT size, FloatT data[], IntT hw, const FloatT kernel[])`

## 2D Gauss FIR Filters

*2D Gaussian finite-impulse response filters.*

- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_2Dx (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_2Dx_small (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_2Dx_arma (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_2Dy (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_2Dy_rowmajor (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_2Dy_colmajor (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_2Dy_small (const arma::Mat< FloatT > &data, arma::Mat< FloatT > &fdata, const arma::Col< FloatT > &kernel)`

## 3D Gauss FIR Filters

*3D Gaussian finite-impulse response filters.*

- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_3Dx (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_3Dx_small (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_3Dy (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_3Dy_small (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_3Dz (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`
- `template<class FloatT = float, class IntT = int32_t>`  
`void boxxer::kernels::gaussFIR_3Dz_small (const arma::Cube< FloatT > &data, arma::Cube< FloatT > &fdata, const arma::Col< FloatT > &kernel)`

### 8.4.1 Detailed Description

The `boxxer::kernels` namespace - low-level Gaussian finite-impulse response filters.



**Author**

Mark J. Olah (mjo@cs.unm DOT edu)

**Date**

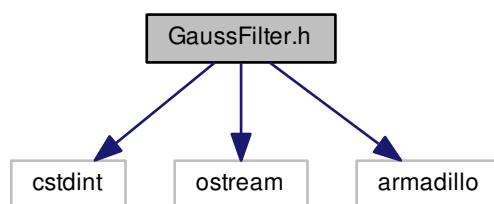
2014-2019

**8.5 GaussFilter.h File Reference**

The class declarations for Gaussian image filter classes.

```
#include <cstdlib>
#include <ostream>
#include <armadillo>
```

Include dependency graph for GaussFilter.h:

**Classes**

- class [boxxer::GaussFIRFilter< FloatT, IdxT >](#)
- class [boxxer::GaussFilter2D< FloatT, IdxT >](#)
- class [boxxer::DoGFilter2D< FloatT, IdxT >](#)
- class [boxxer::LoGFilter2D< FloatT, IdxT >](#)
- class [boxxer::GaussFilter3D< FloatT, IdxT >](#)
- class [boxxer::DoGFilter3D< FloatT, IdxT >](#)
- class [boxxer::LoGFilter3D< FloatT, IdxT >](#)

**Namespaces**

- [boxxer](#)

### 8.5.1 Detailed Description

The class declarations for Gaussian image filter classes.

#### Author

Mark J. Olah (mjo@cs.unm DOT edu)

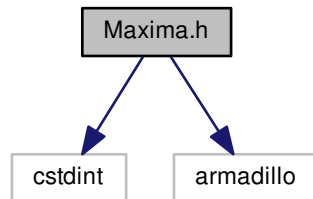
#### Date

2014-2019 These classes are meant to be a per-thread worker class or a direct interface for single threaded processes. Each object has its own local storage of which is only 1 or 2 frames in size.

## 8.6 Maxima.h File Reference

The class declaration for the local maxima finders Maxima2D and Maxima3D.

```
#include <cstdint>
#include <armadillo>
Include dependency graph for Maxima.h:
```



#### Classes

- class [boxxer::Maxima2D< FloatT, IdxT >](#)
- class [boxxer::Maxima3D< FloatT, IdxT >](#)

#### Namespaces

- [boxxer](#)

**8.6.1 Detailed Description**

The class declaration for the local maxima finders Maxima2D and Maxima3D.

**Author**

Mark J. Olah (mjo@cs.unm DOT edu)

**Date**

2014-2019

**8.7 README.md File Reference**



## Index

- boxsize
  - boxxer::Maxima2D, [56](#)
  - boxxer::Maxima3D, [59](#)
- boxxer, [4](#)
  - BoxxerError, [5](#)
- Boxxer2D.h, [64](#)
- Boxxer2D
  - boxxer::Boxxer2D, [17](#)
- Boxxer3D.h, [65](#)
- Boxxer3D
  - boxxer::Boxxer3D, [23](#)
- boxxer::Boxxer2D< FloatT, IdxT >, [14](#)
- boxxer::Boxxer2D
  - Boxxer2D, [17](#)
  - checkMaxima, [17](#)
  - DefaultSigmaRatio, [20](#)
  - dim, [20](#)
  - enumerateImageMaxima, [17](#)
  - filterDoG, [17](#)
  - filterGauss, [18](#)
  - filterLoG, [18](#)
  - filterScaledDoG, [18](#)
  - filterScaledLoG, [19](#)
  - IMatT, [16](#)
  - IVecT, [16](#)
  - ImageStackT, [16](#)
  - ImageT, [16](#)
  - imsize, [20](#)
  - make\_image, [19](#)
  - make\_image\_stack, [19](#)
  - make\_scaled\_image, [19](#)
  - make\_scaled\_image\_stack, [19](#)
  - MatT, [16](#)
  - nScales, [20](#)
  - scaleSpaceDoGMaxima, [20](#)
  - scaleSpaceLoGMaxima, [20](#)
  - ScaledImageStackT, [16](#)
  - ScaledImageT, [16](#)
  - setDoGSigmaRatio, [20](#)
  - sigma, [20](#)
  - sigma\_ratio, [20](#)
  - VecT, [17](#)
- boxxer::Boxxer3D< FloatT, IdxT >, [21](#)
- boxxer::Boxxer3D
  - Boxxer3D, [23](#)
  - checkMaxima, [23](#)
  - DefaultSigmaRatio, [26](#)
  - dim, [26](#)
  - enumerateImageMaxima, [23](#)
  - filterDoG, [23](#)
  - filterGauss, [24](#)
  - filterLoG, [24](#)
  - filterScaledDoG, [24](#)
  - filterScaledLoG, [25](#)
  - IMatT, [22](#)
  - IVecT, [22](#)
  - ImageStackT, [22](#)
  - ImageT, [22](#)
  - imsize, [26](#)
  - make\_image, [25](#)
  - make\_image\_stack, [25](#)
  - make\_scaled\_image, [25](#)
  - MatT, [22](#)
  - nScales, [26](#)
  - scaleSpaceDoGMaxima, [25](#)
  - scaleSpaceLoGMaxima, [26](#)
  - ScaledImageT, [22](#)
  - setDoGSigmaRatio, [26](#)
  - sigma, [26](#)
  - sigma\_ratio, [26](#)
  - VecT, [22](#)
- boxxer::DoGFilter2D< FloatT, IdxT >, [27](#)
- boxxer::DoGFilter2D
  - compute\_Gauss\_FIR\_kernel, [29](#)
  - compute\_LoG\_FIR\_kernel, [29](#)
  - default\_sigma\_hw\_ratio, [30](#)
  - dim, [30](#)
  - DoGFilter2D, [29](#)
  - filter, [29](#)
  - hw, [30](#)
  - IVecT, [28](#)
  - ImageT, [28](#)
  - make\_image, [29](#)
  - MatT, [29](#)
  - max\_kernel\_hw, [30](#)
  - operator<<, [30](#)
  - set\_kernel\_hw, [29](#)
  - set\_sigma\_ratio, [29](#)
  - sigma, [30](#)
  - sigma\_ratio, [30](#)
  - size, [30](#)
  - test\_filter, [30](#)
  - VecT, [29](#)
- boxxer::DoGFilter3D< FloatT, IdxT >, [31](#)
- boxxer::DoGFilter3D
  - compute\_Gauss\_FIR\_kernel, [33](#)
  - compute\_LoG\_FIR\_kernel, [33](#)
  - default\_sigma\_hw\_ratio, [34](#)
  - dim, [34](#)
  - DoGFilter3D, [33](#)
  - filter, [33](#)
  - hw, [34](#)

- IVecT, 33
- ImageT, 33
- make\_image, 33
- MatT, 33
- max\_kernel\_hw, 34
- operator<<, 34
- set\_kernel\_hw, 33
- set\_sigma\_ratio, 34
- sigma, 34
- sigma\_ratio, 34
- size, 34
- test\_filter, 34
- VecT, 33
- boxxer::GaussFIRFilter
  - compute\_Gauss\_FIR\_kernel, 45
  - compute\_Log\_FIR\_kernel, 45
  - default\_sigma\_hw\_ratio, 45
  - dim, 45
  - GaussFIRFilter, 45
  - hw, 45
  - IVecT, 44
  - MatT, 44
  - max\_kernel\_hw, 45
  - set\_kernel\_hw, 45
  - sigma, 45
  - size, 45
  - VecT, 44
- boxxer::GaussFIRFilter< FloatT, IdxT >, 43
- boxxer::GaussFilter2D< FloatT, IdxT >, 35
- boxxer::GaussFilter2D
  - compute\_Gauss\_FIR\_kernel, 37
  - compute\_Log\_FIR\_kernel, 37
  - default\_sigma\_hw\_ratio, 38
  - dim, 38
  - filter, 37
  - GaussFilter2D, 37
  - hw, 38
  - IVecT, 37
  - ImageT, 37
  - make\_image, 37
  - MatT, 37
  - max\_kernel\_hw, 38
  - operator<<, 38
  - set\_kernel\_hw, 37
  - sigma, 38
  - size, 38
  - test\_filter, 38
  - VecT, 37
- boxxer::GaussFilter3D< FloatT, IdxT >, 39
- boxxer::GaussFilter3D
  - compute\_Gauss\_FIR\_kernel, 41
  - compute\_Log\_FIR\_kernel, 41
  - default\_sigma\_hw\_ratio, 42
  - dim, 42
  - filter, 41
  - GaussFilter3D, 41
  - hw, 42
  - IVecT, 40
  - ImageT, 40
  - make\_image, 41
  - MatT, 41
  - max\_kernel\_hw, 42
  - operator<<, 42
  - set\_kernel\_hw, 41
  - sigma, 42
  - size, 42
  - test\_filter, 41
  - VecT, 41
- boxxer::LoGFilter2D< FloatT, IdxT >, 46
- boxxer::LoGFilter2D
  - compute\_Gauss\_FIR\_kernel, 48
  - compute\_Log\_FIR\_kernel, 48
  - default\_sigma\_hw\_ratio, 49
  - dim, 49
  - filter, 48
  - hw, 49
  - IVecT, 48
  - ImageT, 48
  - LoGFilter2D, 48
  - make\_image, 48
  - MatT, 48
  - max\_kernel\_hw, 49
  - operator<<, 49
  - set\_kernel\_hw, 48
  - sigma, 49
  - size, 49
  - test\_filter, 49
  - VecT, 48
- boxxer::LoGFilter3D< FloatT, IdxT >, 50
- boxxer::LoGFilter3D
  - compute\_Gauss\_FIR\_kernel, 52
  - compute\_Log\_FIR\_kernel, 52
  - default\_sigma\_hw\_ratio, 53
  - dim, 53
  - filter, 52
  - hw, 53
  - IVecT, 51
  - ImageT, 51
  - LoGFilter3D, 52
  - make\_image, 52
  - MatT, 52
  - max\_kernel\_hw, 53
  - operator<<, 53
  - set\_kernel\_hw, 52
  - sigma, 53
  - size, 53
  - test\_filter, 52
  - VecT, 52

- boxxer::LogicalError, [54](#)
  - LogicalError, [55](#)
- boxxer::Maxima2D< FloatT, IdxT >, [55](#)
- boxxer::Maxima2D
  - boxsize, [56](#)
  - check\_maxima, [56](#)
  - find\_maxima, [56](#)
  - IMatT, [56](#)
  - IVecT, [56](#)
  - ImageT, [56](#)
  - Maxima2D, [56](#)
  - MinBoxsize, [56](#)
  - Ndim, [57](#)
  - read\_maxima, [56](#)
  - size, [57](#)
  - test\_maxima, [56](#)
  - VecT, [56](#)
- boxxer::Maxima3D< FloatT, IdxT >, [57](#)
- boxxer::Maxima3D
  - boxsize, [59](#)
  - check\_maxima, [59](#)
  - find\_maxima, [59](#)
  - ICubeT, [58](#)
  - IMatT, [58](#)
  - IVecT, [58](#)
  - ImageT, [58](#)
  - Maxima3D, [59](#)
  - MinBoxsize, [59](#)
  - Ndim, [59](#)
  - read\_maxima, [59](#)
  - size, [59](#)
  - test\_maxima, [59](#)
  - VecT, [58](#)
- boxxer::NumericalError, [60](#)
  - NumericalError, [61](#)
- boxxer::ParameterShapeError, [61](#)
  - ParameterShapeError, [62](#)
- boxxer::ParameterValueError, [62](#)
  - ParameterValueError, [63](#)
- boxxer::kernels, [5](#)
  - gaussFIR\_1D\_arma, [8](#)
  - gaussFIR\_1D\_inplace, [9](#)
  - gaussFIR\_1D\_inplace\_arma, [9](#)
  - gaussFIR\_1D\_small, [9](#)
  - gaussFIR\_1D, [6](#)
  - gaussFIR\_2Dx, [10](#)
  - gaussFIR\_2Dx\_arma, [10](#)
  - gaussFIR\_2Dx\_small, [10](#)
  - gaussFIR\_2Dy, [11](#)
  - gaussFIR\_2Dy\_colmajor, [11](#)
  - gaussFIR\_2Dy\_rowmajor, [11](#)
  - gaussFIR\_2Dy\_small, [12](#)
  - gaussFIR\_3Dx, [12](#)
  - gaussFIR\_3Dx\_small, [12](#)
  - gaussFIR\_3Dy, [13](#)
  - gaussFIR\_3Dy\_small, [13](#)
  - gaussFIR\_3Dz, [13](#)
  - gaussFIR\_3Dz\_small, [14](#)
- BoxxerError
  - boxxer, [5](#)
- BoxxerError.h, [66](#)
- check\_maxima
  - boxxer::Maxima2D, [56](#)
  - boxxer::Maxima3D, [59](#)
- checkMaxima
  - boxxer::Boxxer2D, [17](#)
  - boxxer::Boxxer3D, [23](#)
- compute\_Gauss\_FIR\_kernel
  - boxxer::DoGFilter2D, [29](#)
  - boxxer::DoGFilter3D, [33](#)
  - boxxer::GaussFIRFilter, [45](#)
  - boxxer::GaussFilter2D, [37](#)
  - boxxer::GaussFilter3D, [41](#)
  - boxxer::LoGFilter2D, [48](#)
  - boxxer::LoGFilter3D, [52](#)
- compute\_Log\_FIR\_kernel
  - boxxer::DoGFilter2D, [29](#)
  - boxxer::DoGFilter3D, [33](#)
  - boxxer::GaussFIRFilter, [45](#)
  - boxxer::GaussFilter2D, [37](#)
  - boxxer::GaussFilter3D, [41](#)
  - boxxer::LoGFilter2D, [48](#)
  - boxxer::LoGFilter3D, [52](#)
- default\_sigma\_hw\_ratio
  - boxxer::DoGFilter2D, [30](#)
  - boxxer::DoGFilter3D, [34](#)
  - boxxer::GaussFIRFilter, [45](#)
  - boxxer::GaussFilter2D, [38](#)
  - boxxer::GaussFilter3D, [42](#)
  - boxxer::LoGFilter2D, [49](#)
  - boxxer::LoGFilter3D, [53](#)
- DefaultSigmaRatio
  - boxxer::Boxxer2D, [20](#)
  - boxxer::Boxxer3D, [26](#)
- dim
  - boxxer::Boxxer2D, [20](#)
  - boxxer::Boxxer3D, [26](#)
  - boxxer::DoGFilter2D, [30](#)
  - boxxer::DoGFilter3D, [34](#)
  - boxxer::GaussFIRFilter, [45](#)
  - boxxer::GaussFilter2D, [38](#)
  - boxxer::GaussFilter3D, [42](#)
  - boxxer::LoGFilter2D, [49](#)
  - boxxer::LoGFilter3D, [53](#)
- DoGFilter2D
  - boxxer::DoGFilter2D, [29](#)
- DoGFilter3D

- boxxer::DoGFilter3D, 33
- enumeratImageMaxima
  - boxxer::Boxxer2D, 17
  - boxxer::Boxxer3D, 23
- filter
  - boxxer::DoGFilter2D, 29
  - boxxer::DoGFilter3D, 33
  - boxxer::GaussFilter2D, 37
  - boxxer::GaussFilter3D, 41
  - boxxer::LoGFilter2D, 48
  - boxxer::LoGFilter3D, 52
- filterDoG
  - boxxer::Boxxer2D, 17
  - boxxer::Boxxer3D, 23
- filterGauss
  - boxxer::Boxxer2D, 18
  - boxxer::Boxxer3D, 24
- FilterKernels.h, 67
- filterLoG
  - boxxer::Boxxer2D, 18
  - boxxer::Boxxer3D, 24
- filterScaledDoG
  - boxxer::Boxxer2D, 18
  - boxxer::Boxxer3D, 24
- filterScaledLoG
  - boxxer::Boxxer2D, 19
  - boxxer::Boxxer3D, 25
- find\_maxima
  - boxxer::Maxima2D, 56
  - boxxer::Maxima3D, 59
- gaussFIR\_1D\_arma
  - boxxer::kernels, 8
- gaussFIR\_1D\_inplace
  - boxxer::kernels, 9
- gaussFIR\_1D\_inplace\_arma
  - boxxer::kernels, 9
- gaussFIR\_1D\_small
  - boxxer::kernels, 9
- gaussFIR\_1D
  - boxxer::kernels, 6
- gaussFIR\_2Dx
  - boxxer::kernels, 10
- gaussFIR\_2Dx\_arma
  - boxxer::kernels, 10
- gaussFIR\_2Dx\_small
  - boxxer::kernels, 10
- gaussFIR\_2Dy
  - boxxer::kernels, 11
- gaussFIR\_2Dy\_colmajor
  - boxxer::kernels, 11
- gaussFIR\_2Dy\_rowmajor
  - boxxer::kernels, 11
- gaussFIR\_2Dy\_small
  - boxxer::kernels, 12
- gaussFIR\_3Dx
  - boxxer::kernels, 12
- gaussFIR\_3Dx\_small
  - boxxer::kernels, 12
- gaussFIR\_3Dy
  - boxxer::kernels, 13
- gaussFIR\_3Dy\_small
  - boxxer::kernels, 13
- gaussFIR\_3Dz
  - boxxer::kernels, 13
- gaussFIR\_3Dz\_small
  - boxxer::kernels, 14
- GaussFIRFilter
  - boxxer::GaussFIRFilter, 45
- GaussFilter.h, 69
- GaussFilter2D
  - boxxer::GaussFilter2D, 37
- GaussFilter3D
  - boxxer::GaussFilter3D, 41
- hw
  - boxxer::DoGFilter2D, 30
  - boxxer::DoGFilter3D, 34
  - boxxer::GaussFIRFilter, 45
  - boxxer::GaussFilter2D, 38
  - boxxer::GaussFilter3D, 42
  - boxxer::LoGFilter2D, 49
  - boxxer::LoGFilter3D, 53
- ICubeT
  - boxxer::Maxima3D, 58
- IMatT
  - boxxer::Boxxer2D, 16
  - boxxer::Boxxer3D, 22
  - boxxer::Maxima2D, 56
  - boxxer::Maxima3D, 58
- IVecT
  - boxxer::Boxxer2D, 16
  - boxxer::Boxxer3D, 22
  - boxxer::DoGFilter2D, 28
  - boxxer::DoGFilter3D, 33
  - boxxer::GaussFIRFilter, 44
  - boxxer::GaussFilter2D, 37
  - boxxer::GaussFilter3D, 40
  - boxxer::LoGFilter2D, 48
  - boxxer::LoGFilter3D, 51
  - boxxer::Maxima2D, 56
  - boxxer::Maxima3D, 58
- ImageStackT
  - boxxer::Boxxer2D, 16
  - boxxer::Boxxer3D, 22
- ImageT
  - boxxer::Boxxer2D, 16



- boxxer::Boxxer3D, [22](#)
- boxxer::DoGFilter2D, [28](#)
- boxxer::DoGFilter3D, [33](#)
- boxxer::GaussFilter2D, [37](#)
- boxxer::GaussFilter3D, [40](#)
- boxxer::LoGFilter2D, [48](#)
- boxxer::LoGFilter3D, [51](#)
- boxxer::Maxima2D, [56](#)
- boxxer::Maxima3D, [58](#)
- imshow
  - boxxer::Boxxer2D, [20](#)
  - boxxer::Boxxer3D, [26](#)
- LoGFilter2D
  - boxxer::LoGFilter2D, [48](#)
- LoGFilter3D
  - boxxer::LoGFilter3D, [52](#)
- LogicalError
  - boxxer::LogicalError, [55](#)
- make\_image
  - boxxer::Boxxer2D, [19](#)
  - boxxer::Boxxer3D, [25](#)
  - boxxer::DoGFilter2D, [29](#)
  - boxxer::DoGFilter3D, [33](#)
  - boxxer::GaussFilter2D, [37](#)
  - boxxer::GaussFilter3D, [41](#)
  - boxxer::LoGFilter2D, [48](#)
  - boxxer::LoGFilter3D, [52](#)
- make\_image\_stack
  - boxxer::Boxxer2D, [19](#)
  - boxxer::Boxxer3D, [25](#)
- make\_scaled\_image
  - boxxer::Boxxer2D, [19](#)
  - boxxer::Boxxer3D, [25](#)
- make\_scaled\_image\_stack
  - boxxer::Boxxer2D, [19](#)
- MatT
  - boxxer::Boxxer2D, [16](#)
  - boxxer::Boxxer3D, [22](#)
  - boxxer::DoGFilter2D, [29](#)
  - boxxer::DoGFilter3D, [33](#)
  - boxxer::GaussFIRFilter, [44](#)
  - boxxer::GaussFilter2D, [37](#)
  - boxxer::GaussFilter3D, [41](#)
  - boxxer::LoGFilter2D, [48](#)
  - boxxer::LoGFilter3D, [52](#)
- max\_kernel\_hw
  - boxxer::DoGFilter2D, [30](#)
  - boxxer::DoGFilter3D, [34](#)
  - boxxer::GaussFIRFilter, [45](#)
  - boxxer::GaussFilter2D, [38](#)
  - boxxer::GaussFilter3D, [42](#)
  - boxxer::LoGFilter2D, [49](#)
  - boxxer::LoGFilter3D, [53](#)
- Maxima.h, [70](#)
- Maxima2D
  - boxxer::Maxima2D, [56](#)
- Maxima3D
  - boxxer::Maxima3D, [59](#)
- MinBoxsize
  - boxxer::Maxima2D, [56](#)
  - boxxer::Maxima3D, [59](#)
- nScales
  - boxxer::Boxxer2D, [20](#)
  - boxxer::Boxxer3D, [26](#)
- Ndim
  - boxxer::Maxima2D, [57](#)
  - boxxer::Maxima3D, [59](#)
- NumericalError
  - boxxer::NumericalError, [61](#)
- operator<<
  - boxxer::DoGFilter2D, [30](#)
  - boxxer::DoGFilter3D, [34](#)
  - boxxer::GaussFilter2D, [38](#)
  - boxxer::GaussFilter3D, [42](#)
  - boxxer::LoGFilter2D, [49](#)
  - boxxer::LoGFilter3D, [53](#)
- ParameterShapeError
  - boxxer::ParameterShapeError, [62](#)
- ParameterValueError
  - boxxer::ParameterValueError, [63](#)
- README.md, [71](#)
- read\_maxima
  - boxxer::Maxima2D, [56](#)
  - boxxer::Maxima3D, [59](#)
- scaleSpaceDoGMaxima
  - boxxer::Boxxer2D, [20](#)
  - boxxer::Boxxer3D, [25](#)
- scaleSpaceLoGMaxima
  - boxxer::Boxxer2D, [20](#)
  - boxxer::Boxxer3D, [26](#)
- ScaledImageStackT
  - boxxer::Boxxer2D, [16](#)
- ScaledImageT
  - boxxer::Boxxer2D, [16](#)
  - boxxer::Boxxer3D, [22](#)
- set\_kernel\_hw
  - boxxer::DoGFilter2D, [29](#)
  - boxxer::DoGFilter3D, [33](#)
  - boxxer::GaussFIRFilter, [45](#)
  - boxxer::GaussFilter2D, [37](#)
  - boxxer::GaussFilter3D, [41](#)
  - boxxer::LoGFilter2D, [48](#)
  - boxxer::LoGFilter3D, [52](#)

- set\_sigma\_ratio
  - boxxer::DoGFilter2D, [29](#)
  - boxxer::DoGFilter3D, [34](#)
- setDoGSigmaRatio
  - boxxer::Boxxer2D, [20](#)
  - boxxer::Boxxer3D, [26](#)
- sigma
  - boxxer::Boxxer2D, [20](#)
  - boxxer::Boxxer3D, [26](#)
  - boxxer::DoGFilter2D, [30](#)
  - boxxer::DoGFilter3D, [34](#)
  - boxxer::GaussFIRFilter, [45](#)
  - boxxer::GaussFilter2D, [38](#)
  - boxxer::GaussFilter3D, [42](#)
  - boxxer::LoGFilter2D, [49](#)
  - boxxer::LoGFilter3D, [53](#)
- sigma\_ratio
  - boxxer::Boxxer2D, [20](#)
  - boxxer::Boxxer3D, [26](#)
  - boxxer::DoGFilter2D, [30](#)
  - boxxer::DoGFilter3D, [34](#)
- size
  - boxxer::DoGFilter2D, [30](#)
  - boxxer::DoGFilter3D, [34](#)
  - boxxer::GaussFIRFilter, [45](#)
  - boxxer::GaussFilter2D, [38](#)
  - boxxer::GaussFilter3D, [42](#)
  - boxxer::LoGFilter2D, [49](#)
  - boxxer::LoGFilter3D, [53](#)
  - boxxer::Maxima2D, [57](#)
  - boxxer::Maxima3D, [59](#)
- test\_filter
  - boxxer::DoGFilter2D, [30](#)
  - boxxer::DoGFilter3D, [34](#)
  - boxxer::GaussFilter2D, [38](#)
  - boxxer::GaussFilter3D, [41](#)
  - boxxer::LoGFilter2D, [49](#)
  - boxxer::LoGFilter3D, [52](#)
- test\_maxima
  - boxxer::Maxima2D, [56](#)
  - boxxer::Maxima3D, [59](#)
- VecT
  - boxxer::Boxxer2D, [17](#)
  - boxxer::Boxxer3D, [22](#)
  - boxxer::DoGFilter2D, [29](#)
  - boxxer::DoGFilter3D, [33](#)
  - boxxer::GaussFIRFilter, [44](#)
  - boxxer::GaussFilter2D, [37](#)
  - boxxer::GaussFilter3D, [41](#)
  - boxxer::LoGFilter2D, [48](#)
  - boxxer::LoGFilter3D, [52](#)
  - boxxer::Maxima2D, [56](#)
  - boxxer::Maxima3D, [58](#)