| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/CropImageFilter.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/awt/image/ConvolveOp.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/DataBuffer.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/CropImageFilter.html)    [**NO FRAMES**](http://docs.google.com/CropImageFilter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: FIELD | [CONSTR](#17dp8vu) | [METHOD](#26in1rg) |

## **java.awt.image**

Class CropImageFilter

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.awt.image.ImageFilter](http://docs.google.com/java/awt/image/ImageFilter.html)  
 **java.awt.image.CropImageFilter**

**All Implemented Interfaces:** [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html)

public class **CropImageFilter**extends [ImageFilter](http://docs.google.com/java/awt/image/ImageFilter.html)

An ImageFilter class for cropping images. This class extends the basic ImageFilter Class to extract a given rectangular region of an existing Image and provide a source for a new image containing just the extracted region. It is meant to be used in conjunction with a FilteredImageSource object to produce cropped versions of existing images.

**See Also:**[FilteredImageSource](http://docs.google.com/java/awt/image/FilteredImageSource.html), [ImageFilter](http://docs.google.com/java/awt/image/ImageFilter.html)

| **Field Summary** | |
| --- | --- |

| **Fields inherited from class java.awt.image.**[**ImageFilter**](http://docs.google.com/java/awt/image/ImageFilter.html) |
| --- |
| [consumer](http://docs.google.com/java/awt/image/ImageFilter.html#consumer) |

| **Fields inherited from interface java.awt.image.**[**ImageConsumer**](http://docs.google.com/java/awt/image/ImageConsumer.html) |
| --- |
| [COMPLETESCANLINES](http://docs.google.com/java/awt/image/ImageConsumer.html#COMPLETESCANLINES), [IMAGEABORTED](http://docs.google.com/java/awt/image/ImageConsumer.html#IMAGEABORTED), [IMAGEERROR](http://docs.google.com/java/awt/image/ImageConsumer.html#IMAGEERROR), [RANDOMPIXELORDER](http://docs.google.com/java/awt/image/ImageConsumer.html#RANDOMPIXELORDER), [SINGLEFRAME](http://docs.google.com/java/awt/image/ImageConsumer.html#SINGLEFRAME), [SINGLEFRAMEDONE](http://docs.google.com/java/awt/image/ImageConsumer.html#SINGLEFRAMEDONE), [SINGLEPASS](http://docs.google.com/java/awt/image/ImageConsumer.html#SINGLEPASS), [STATICIMAGEDONE](http://docs.google.com/java/awt/image/ImageConsumer.html#STATICIMAGEDONE), [TOPDOWNLEFTRIGHT](http://docs.google.com/java/awt/image/ImageConsumer.html#TOPDOWNLEFTRIGHT) |

| **Constructor Summary** | |
| --- | --- |
| [**CropImageFilter**](http://docs.google.com/java/awt/image/CropImageFilter.html#CropImageFilter(int,%20int,%20int,%20int))(int x, int y, int w, int h)            Constructs a CropImageFilter that extracts the absolute rectangular region of pixels from its source Image as specified by the x, y, w, and h parameters. |

| **Method Summary** | |
| --- | --- |
| void | [**setDimensions**](http://docs.google.com/java/awt/image/CropImageFilter.html#setDimensions(int,%20int))(int w, int h)            Override the source image's dimensions and pass the dimensions of the rectangular cropped region to the ImageConsumer. |
| void | [**setPixels**](http://docs.google.com/java/awt/image/CropImageFilter.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int))(int x, int y, int w, int h, [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model, byte[] pixels, int off, int scansize)            Determine whether the delivered byte pixels intersect the region to be extracted and passes through only that subset of pixels that appear in the output region. |
| void | [**setPixels**](http://docs.google.com/java/awt/image/CropImageFilter.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20int%5B%5D,%20int,%20int))(int x, int y, int w, int h, [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model, int[] pixels, int off, int scansize)            Determine if the delivered int pixels intersect the region to be extracted and pass through only that subset of pixels that appear in the output region. |
| void | [**setProperties**](http://docs.google.com/java/awt/image/CropImageFilter.html#setProperties(java.util.Hashtable))([Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> props)            Passes along the properties from the source object after adding a property indicating the cropped region. |

| **Methods inherited from class java.awt.image.**[**ImageFilter**](http://docs.google.com/java/awt/image/ImageFilter.html) |
| --- |
| [clone](http://docs.google.com/java/awt/image/ImageFilter.html#clone()), [getFilterInstance](http://docs.google.com/java/awt/image/ImageFilter.html#getFilterInstance(java.awt.image.ImageConsumer)), [imageComplete](http://docs.google.com/java/awt/image/ImageFilter.html#imageComplete(int)), [resendTopDownLeftRight](http://docs.google.com/java/awt/image/ImageFilter.html#resendTopDownLeftRight(java.awt.image.ImageProducer)), [setColorModel](http://docs.google.com/java/awt/image/ImageFilter.html#setColorModel(java.awt.image.ColorModel)), [setHints](http://docs.google.com/java/awt/image/ImageFilter.html#setHints(int)) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Constructor Detail** |
| --- |

### CropImageFilter

public **CropImageFilter**(int x,  
 int y,  
 int w,  
 int h)

Constructs a CropImageFilter that extracts the absolute rectangular region of pixels from its source Image as specified by the x, y, w, and h parameters.

**Parameters:**x - the x location of the top of the rectangle to be extractedy - the y location of the top of the rectangle to be extractedw - the width of the rectangle to be extractedh - the height of the rectangle to be extracted

| **Method Detail** |
| --- |

### setProperties

public void **setProperties**([Hashtable](http://docs.google.com/java/util/Hashtable.html)<?,?> props)

Passes along the properties from the source object after adding a property indicating the cropped region. This method invokes super.setProperties, which might result in additional properties being added.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setProperties](http://docs.google.com/java/awt/image/ImageConsumer.html#setProperties(java.util.Hashtable)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html)**Overrides:**[setProperties](http://docs.google.com/java/awt/image/ImageFilter.html#setProperties(java.util.Hashtable)) in class [ImageFilter](http://docs.google.com/java/awt/image/ImageFilter.html) **Parameters:**props - the properties from the source object

### setDimensions

public void **setDimensions**(int w,  
 int h)

Override the source image's dimensions and pass the dimensions of the rectangular cropped region to the ImageConsumer.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setDimensions](http://docs.google.com/java/awt/image/ImageConsumer.html#setDimensions(int,%20int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html)**Overrides:**[setDimensions](http://docs.google.com/java/awt/image/ImageFilter.html#setDimensions(int,%20int)) in class [ImageFilter](http://docs.google.com/java/awt/image/ImageFilter.html) **Parameters:**w - the width of the source imageh - the height of the source image**See Also:**[ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html)

### setPixels

public void **setPixels**(int x,  
 int y,  
 int w,  
 int h,  
 [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model,  
 byte[] pixels,  
 int off,  
 int scansize)

Determine whether the delivered byte pixels intersect the region to be extracted and passes through only that subset of pixels that appear in the output region.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setPixels](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html)**Overrides:**[setPixels](http://docs.google.com/java/awt/image/ImageFilter.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int)) in class [ImageFilter](http://docs.google.com/java/awt/image/ImageFilter.html) **Parameters:**x - the X coordinate of the upper-left corner of the area of pixels to be sety - the Y coordinate of the upper-left corner of the area of pixels to be setw - the width of the area of pixelsh - the height of the area of pixelsmodel - the specified ColorModelpixels - the array of pixelsoff - the offset into the pixels arrayscansize - the distance from one row of pixels to the next in the pixels array**See Also:**[ImageConsumer.setPixels(int, int, int, int, java.awt.image.ColorModel, byte[], int, int)](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int))

### setPixels

public void **setPixels**(int x,  
 int y,  
 int w,  
 int h,  
 [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) model,  
 int[] pixels,  
 int off,  
 int scansize)

Determine if the delivered int pixels intersect the region to be extracted and pass through only that subset of pixels that appear in the output region.

Note: This method is intended to be called by the ImageProducer of the Image whose pixels are being filtered. Developers using this class to filter pixels from an image should avoid calling this method directly since that operation could interfere with the filtering operation.

**Specified by:**[setPixels](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20int%5B%5D,%20int,%20int)) in interface [ImageConsumer](http://docs.google.com/java/awt/image/ImageConsumer.html)**Overrides:**[setPixels](http://docs.google.com/java/awt/image/ImageFilter.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20int%5B%5D,%20int,%20int)) in class [ImageFilter](http://docs.google.com/java/awt/image/ImageFilter.html) **Parameters:**x - the X coordinate of the upper-left corner of the area of pixels to be sety - the Y coordinate of the upper-left corner of the area of pixels to be setw - the width of the area of pixelsh - the height of the area of pixelsmodel - the specified ColorModelpixels - the array of pixelsoff - the offset into the pixels arrayscansize - the distance from one row of pixels to the next in the pixels array**See Also:**[ImageConsumer.setPixels(int, int, int, int, java.awt.image.ColorModel, byte[], int, int)](http://docs.google.com/java/awt/image/ImageConsumer.html#setPixels(int,%20int,%20int,%20int,%20java.awt.image.ColorModel,%20byte%5B%5D,%20int,%20int))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/CropImageFilter.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/awt/image/ConvolveOp.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/DataBuffer.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/CropImageFilter.html)    [**NO FRAMES**](http://docs.google.com/CropImageFilter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#2et92p0) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: FIELD | [CONSTR](#17dp8vu) | [METHOD](#26in1rg) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).