| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BufferedImageOp.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/BufferedImageFilter.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/BufferStrategy.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/BufferedImageOp.html)    [**NO FRAMES**](http://docs.google.com/BufferedImageOp.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **java.awt.image**

Interface BufferedImageOp

**All Known Implementing Classes:** [AffineTransformOp](http://docs.google.com/java/awt/image/AffineTransformOp.html), [ColorConvertOp](http://docs.google.com/java/awt/image/ColorConvertOp.html), [ConvolveOp](http://docs.google.com/java/awt/image/ConvolveOp.html), [LookupOp](http://docs.google.com/java/awt/image/LookupOp.html), [RescaleOp](http://docs.google.com/java/awt/image/RescaleOp.html)

public interface **BufferedImageOp**

This interface describes single-input/single-output operations performed on BufferedImage objects. It is implemented by AffineTransformOp, ConvolveOp, ColorConvertOp, RescaleOp, and LookupOp. These objects can be passed into a BufferedImageFilter to operate on a BufferedImage in the ImageProducer-ImageFilter-ImageConsumer paradigm.

Classes that implement this interface must specify whether or not they allow in-place filtering-- filter operations where the source object is equal to the destination object.

This interface cannot be used to describe more sophisticated operations such as those that take multiple sources. Note that this restriction also means that the values of the destination pixels prior to the operation are not used as input to the filter operation.

**See Also:**[BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html), [BufferedImageFilter](http://docs.google.com/java/awt/image/BufferedImageFilter.html), [AffineTransformOp](http://docs.google.com/java/awt/image/AffineTransformOp.html), [BandCombineOp](http://docs.google.com/java/awt/image/BandCombineOp.html), [ColorConvertOp](http://docs.google.com/java/awt/image/ColorConvertOp.html), [ConvolveOp](http://docs.google.com/java/awt/image/ConvolveOp.html), [LookupOp](http://docs.google.com/java/awt/image/LookupOp.html), [RescaleOp](http://docs.google.com/java/awt/image/RescaleOp.html)

| **Method Summary** | |
| --- | --- |
| [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) | [**createCompatibleDestImage**](http://docs.google.com/java/awt/image/BufferedImageOp.html#createCompatibleDestImage(java.awt.image.BufferedImage,%20java.awt.image.ColorModel))([BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) src, [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) destCM)            Creates a zeroed destination image with the correct size and number of bands. |
| [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) | [**filter**](http://docs.google.com/java/awt/image/BufferedImageOp.html#filter(java.awt.image.BufferedImage,%20java.awt.image.BufferedImage))([BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) src, [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) dest)            Performs a single-input/single-output operation on a BufferedImage. |
| [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**getBounds2D**](http://docs.google.com/java/awt/image/BufferedImageOp.html#getBounds2D(java.awt.image.BufferedImage))([BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) src)            Returns the bounding box of the filtered destination image. |
| [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) | [**getPoint2D**](http://docs.google.com/java/awt/image/BufferedImageOp.html#getPoint2D(java.awt.geom.Point2D,%20java.awt.geom.Point2D))([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) srcPt, [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) dstPt)            Returns the location of the corresponding destination point given a point in the source image. |
| [RenderingHints](http://docs.google.com/java/awt/RenderingHints.html) | [**getRenderingHints**](http://docs.google.com/java/awt/image/BufferedImageOp.html#getRenderingHints())()            Returns the rendering hints for this operation. |

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

### filter

[BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) **filter**([BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) src,  
 [BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) dest)

Performs a single-input/single-output operation on a BufferedImage. If the color models for the two images do not match, a color conversion into the destination color model is performed. If the destination image is null, a BufferedImage with an appropriate ColorModel is created.

An IllegalArgumentException may be thrown if the source and/or destination image is incompatible with the types of images $ allowed by the class implementing this filter.

**Parameters:**src - The BufferedImage to be filtereddest - The BufferedImage in which to store the results$ **Returns:**The filtered BufferedImage. **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - If the source and/or destination image is not compatible with the types of images allowed by the class implementing this filter.

### getBounds2D

[Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) **getBounds2D**([BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) src)

Returns the bounding box of the filtered destination image. An IllegalArgumentException may be thrown if the source image is incompatible with the types of images allowed by the class implementing this filter.

**Parameters:**src - The BufferedImage to be filtered **Returns:**The Rectangle2D representing the destination image's bounding box.

### createCompatibleDestImage

[BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) **createCompatibleDestImage**([BufferedImage](http://docs.google.com/java/awt/image/BufferedImage.html) src,  
 [ColorModel](http://docs.google.com/java/awt/image/ColorModel.html) destCM)

Creates a zeroed destination image with the correct size and number of bands. An IllegalArgumentException may be thrown if the source image is incompatible with the types of images allowed by the class implementing this filter.

**Parameters:**src - The BufferedImage to be filtereddestCM - ColorModel of the destination. If null, the ColorModel of the source is used. **Returns:**The zeroed destination image.

### getPoint2D

[Point2D](http://docs.google.com/java/awt/geom/Point2D.html) **getPoint2D**([Point2D](http://docs.google.com/java/awt/geom/Point2D.html) srcPt,  
 [Point2D](http://docs.google.com/java/awt/geom/Point2D.html) dstPt)

Returns the location of the corresponding destination point given a point in the source image. If dstPt is specified, it is used to hold the return value.

**Parameters:**srcPt - the Point2D that represents the point in the source imagedstPt - The Point2D in which to store the result **Returns:**The Point2D in the destination image that corresponds to the specified point in the source image.

### getRenderingHints

[RenderingHints](http://docs.google.com/java/awt/RenderingHints.html) **getRenderingHints**()

Returns the rendering hints for this operation.

**Returns:**The RenderingHints object for this BufferedImageOp. Returns null if no hints have been set.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BufferedImageOp.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/BufferedImageFilter.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/image/BufferStrategy.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/image/BufferedImageOp.html)    [**NO FRAMES**](http://docs.google.com/BufferedImageOp.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[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).