



CFX_JImage Version 1.4

Copyright Digital Crew Ltd 2004. All rights reserved.
Digital Crew , Pembroke House, Pembroke St, Ireland.
+353 21 4277778. info@digital-crew.com



INTRODUCTION	4
ACKNOWLEDGEMENTS	4
INSTALLATION	4
IMAGE FORMATS	5
SUPPORTED READ FORMATS	5
SUPPORTED WRITE FORMATS	5
GETTING INFORMATION	6
ACTION="GETSIZE"	6
ACTION="GETVERSION"	6
RESIZING OPERATIONS	7
ACTION="RESIZE"	7
ACTION="RESIZEBESTFIT"	9
SIMPLE TRANSFORMATION OPERATIONS	10
ACTION="ROTATELEFT90"	10
ACTION="ROTATERIGHT90"	11
ACTION="ROTATE180"	12
ACTION="FLIP"	13
ACTION="MIRROR"	14
ACTION="CROP"	15
ACTION="SHEAR"	16
COLOR OPERATIONS	17
ACTION="COLOURIZE"	17
ACTION="ADJUSTBRIGHTNESS"	19
ACTION="ADJUSTCONTRAST"	20
ACTION="GAMMACORRECTION"	21
ACTION="GRAYSCALE"	22
ACTION="INVERT"	23
FILTERS	24
ACTION="FILTER"	24
ACTION="FILTER", FILTER="BLUR"	25
ACTION="FILTER", FILTER="EDGE DETECT"	26
ACTION="FILTER", FILTER="EMBOSS"	27
ACTION="FILTER", FILTER="HORIZONTAL PREWITT"	28
ACTION="FILTER", FILTER="HORIZONTAL SOBEL"	29
ACTION="FILTER", FILTER="LITHOGRAPH"	30
ACTION="FILTER", FILTER="MAXIMUM"	31
ACTION="FILTER", FILTER="MEAN"	32
ACTION="FILTER", FILTER="MEDIAN"	33
ACTION="FILTER", FILTER="MINIMUM"	34
ACTION="FILTER", FILTER="OIL"	35
ACTION="FILTER", FILTER="PSYCHEDELIC DISTILLATION"	36
ACTION="FILTER", FILTER="SHARPEN"	37
ACTION="FILTER", FILTER="VERTICAL PREWITT"	38

ACTION="FILTER", FILTER="VERTICAL SOBEL"	39
MISCELLANEOUS.....	40
ACTION="DRAWTEXT"	40
ACTION="BORDER"	41
ACTION="DRAWRECT"	42
ACTION="FILLRECT"	43
ACTION="DRAWOVAL"	44
ACTION="FILLOVAL"	45
ACTION="DRAWLINE"	46
ACTION="FILLROUNDRECT"	47

Introduction

CFX_JImage is a powerful single pass image manipulation CFX tag for ColdFusion. It can read files in 13 different formats and many more sub-formats and can automatically save files to 9 different formats (jpeg, gif, png, bmp, pnm, palm, pbm, pgm, ppm). When saving as a GIF, which must use a color palette, CFX_JImage will automatically determine the lowest number of colors to use when saving the file.

CFX_JImage is fast!

If you have any problems using CFX_JImage or if you would like to see some functionality added, please don't hesitate to contact us via email: support@digital-crew.com.

Digital Crew, Pembroke House, Pembroke Street, Cork, Ireland
ph. +353 21 4277778 fax. +353 21 4277778

Acknowledgements

CFX_JImage uses the excellent Java Imaging Utilities Library. The Java Imaging Utilities Library is embedded in the JImage.jar. The JIU library is also provided as-is for reference and to satisfy the terms of the GPL license under which it is distributed. More information is available at <http://sourceforge.net/projects/jiu/>.

Installation

1. Copy file JImage.jar (or JImagePreview.jar if using preview version) to "c:\CFusion\Java\classes".
2. Open up the ColdFusion CFIDE for the server in your browser and login, then select "JVM and Java Settings" from the menu. Where you see the class path value append ";c:\CFusion\Java\classes\JImage.jar" to the value. Click update to save the new value.
3. Go to "CFX Tags" and click to register a new Java CFX. Enter "JImage.jar" (or JImagePreview.jar if using preview version) as the class file and enter "CFX_JImage" as the CFX name.
4. Stop and restart the ColdFusion server.

That's it - CFX_JImage should now work!

Run test.cfm to verify that CFX_JImage is installed.

Image Formats

If CFX_JImage cannot save or load a file with a particular file extension type it will throw an error. When saving it will automatically choose the file format based on the file extension of the output filename.

Supported Read Formats

This is a list of the image file formats that JImage can read:

- BMP
- FlashPix
- GIF
- JPEG
- PNG
- PNM
- TIFF
- WBMP
- PALM
- PNM
- PBM
- PGM
- PPM

Supported Write Formats

This is a list of the image file formats that JImage can write an output image as:

- BMP
- GIF
- JPEG
- PNG
- PNM
- TIFF
- PALM
- PNM
- PBM
- PGM
- PPM

Getting Information

Action="GetSize"

The "GetSize" action simply returns the image's width and height in the variables "JImage.ImageWidth" and "JImage.ImageHeight". The image file is not modified.

Note that the "JImage.ImageWidth" and "JImage.ImageHeight" variables are also set after performing any other image operation.

Syntax

```
<cfx_JImage action="getSize" file="FULL_PATH_TO_FILE">
```

Example

```
<cfx_JImage action="getSize" file="#baseImage#">
```

Result

```
JImage.imageWidth: #JImage.ImageWidth#  
JImage.imageHeight: #JImage.ImageHeight#
```

Action="GetVersion"

This simply returns the current JImage version number into the variable "JImage.Version".

Syntax

```
<cfx_JImage action="getVersion">
```

Example

```
<cfx_JImage action="getVersion">
```

Result

```
JImageVersion: #JImage.Version#
```

Resizing Operations

Action="Resize"

The "Resize" action will stretch will load an image and stretch it to the given width and height. If only the "width" attribute is provided, the height will be automatically calculated and vice-versa. If the "outFile" parameter is not provided, the result will be saved back to the same file.

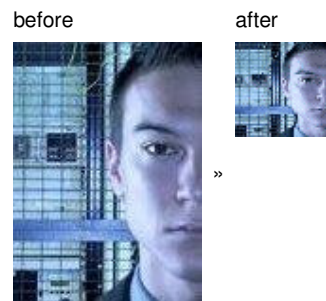
Syntax

```
<cfx_JImage action="resize" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] [width="WIDTH"] [height="HEIGHT"]>
```

Example 1

```
<cfx_JImage action="resize" file="#baseImage#"  
outfile="#path#example_resize_50x50.png" width="50" height="50">
```

Result 1



Example 2

```
<cfx_JImage action="resize" file="#baseImage#"  
outfile="#path#example_resize_120.png" width="120">
```

Result 2



Example 3

```
<cfx_JImage action="resize" file="#baseImage#"
outfile="#path#example_resize_300_90.jpg" width="300" height="90">
```

Result 3

before



after



»

Action="ResizeBestFit"

The "ResizeBestFit" action will resize an image the given image but will intelligently crop off the outer edges of the image to provide the best possible fit. In general when resizing images to match a specific width and height, this is the action to use. Both the width and height attributes should be passed.

Syntax

```
<cfx_JImage action="resizeBestFit" file="FULL_PATH_TO_FILE"
[outfile="FULL_PATH_TO_FILE"] width="WIDTH" height="HEIGHT">
```

Example

```
<cfx_JImage action="ResizeBestFit" file="#baseImage#"
outfile="#path#example_resizeBestFit_250x200.jpg" width="250" height="200">
```

Result

before



after



»

Simple Transformation Operations

Action="RotateLeft90"

Rotates the image 90° to the left, anti-clockwise.

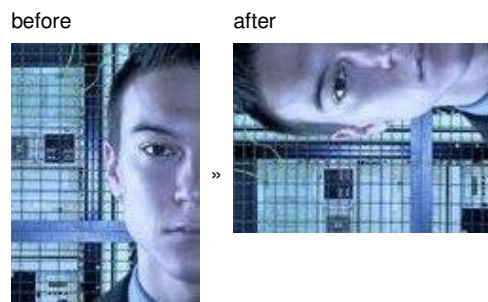
Syntax

```
<cfx_JImage action="rotateLeft90" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"]>
```

Example

```
<cfx_JImage action="rotateLeft90" file="#baseImage#"  
outfile="#path#example_rotate_left90.png">
```

Result



Action="RotateRight90"

Rotates the image 90° to the right, clockwise.

Syntax

```
<cfx_JImage action="rotateRight90" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"]>
```

Example

```
<cfx_JImage action="rotateRight90" file="#baseImage#"  
outfile="#path#example_rotate_left90.png">
```

Result

before



after



»

Action="Rotate180"

Rotates the image 180°.

Syntax

```
<cfx_JImage action="rotate180" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"]>
```

Example

```
<cfx_JImage action="rotate180" file="#baseImage#"  
outfile="#path#example_rotate_180.bmp">
```

Result

before



after



»

Action="Flip"

Flips the image upside-down.

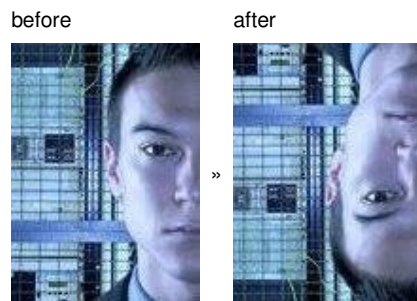
Syntax

```
<cfx_JImage action="flip" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"]>
```

Example

```
<cfx_JImage action="flip" file="#baseImage#" outfile="#path#example_flip.jpg">
```

Result



Action="Mirror"

Mirrors the image.

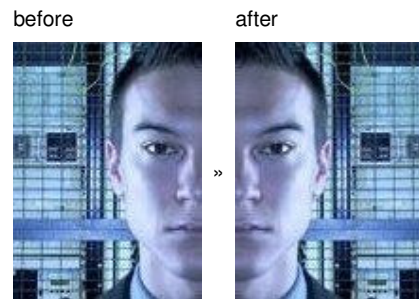
Syntax

```
<cfx_JImage action="mirror" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"]>
```

Example

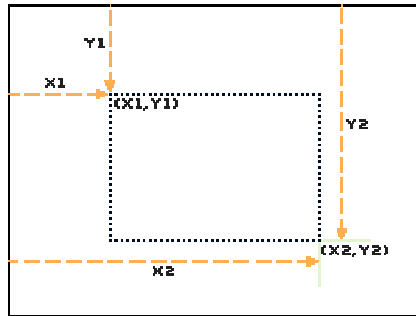
```
<cfx_JImage action="mirror" file="#baseImage#" outfile="#path#example_mirror.png">
```

Result



Action="Crop"

Given two sets of coordinates, this action cuts the corresponding section out of the image. If X1 or Y1 are less than 0, they are reset to 0. If X2 or Y2 are bigger than the width or height of the image accordingly, they are set to the width and height accordingly. X2 must be bigger than X1. Y2 must be bigger than Y1.



Syntax

```
<cfx_JImage action="crop" file="FULL_PATH_TO_FILE"
[outfile="FULL_PATH_TO_FILE"]
x1="[TOP_LEFT_X_OFFSET]"
y1="[TOP_LEFT_Y_OFFSET]"
x2="[BOTTOM_RIGHT_X_OFFSET]"
y2="[BOTTOM_RIGHT_Y_OFFSET]" >
```

Example

```
<cfx_JImage action="crop" file="#baseImage#" outfile="#path#example_mirror.png"
x1="20" y1="20" x2="100" y2="100" >
```

Result



Action="Shear"

Shears the image.

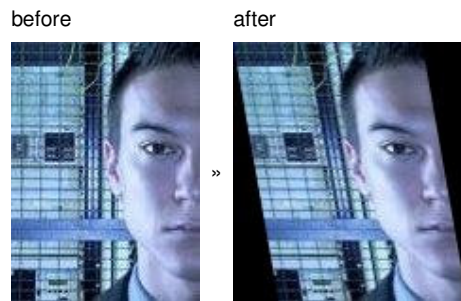
Syntax

```
<cfx_JImage action="shear" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] angle="[ANGLE_TO_SHEER_BY]">
```

Example

```
<cfx_JImage action="shear" file="#baseImage#" outfile="#path#example_shear_10.jpg"  
angle="10">
```

Result



Color Operations

Action="Colourize"

This changes the hue of the image making the entire image a shade of one color. The hue must be a value between 0 and 256. Attribute "hue" must be passed.

The "saturation" value is optional and defaults to 255. Saturation must be a value between 0 and 255.

The "brightness" value is optional and defaults to 0. Brightness must be a value between -100 and 100. If the brightness is negative, the image gets darker; if the brightness value is positive the image get brighter.



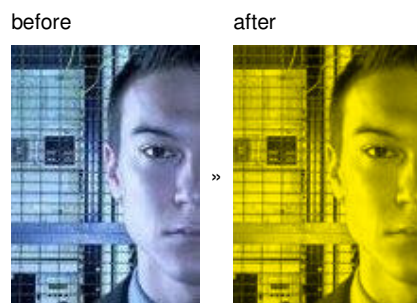
Syntax

```
<cfx_JImage action="Colourize" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] hue="[HUE_VALUE]" [saturation="[SATURATION]"]  
[brightness="[BRIGHTNESS]"] >
```

Example 1

```
<cfx_JImage action="Colourize" file="#baseImage#"  
outfile="#path#example_colourize_40.png" hue="40">
```

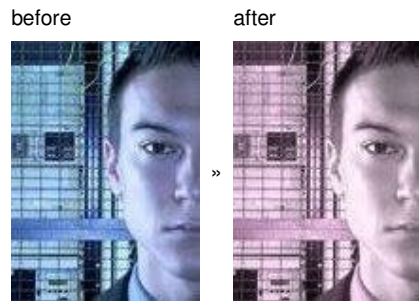
Result 1



Example 2

```
<cfx_JImage action="Colourize" file="#baseImage#"
outfile="#path#example_colourize_220_sat_50.jpg" hue="220" saturation="50">
```

Result 2



Action="AdjustBrightness"

This changes the brightness of the image.

The required "brightness" attribute must be an integer value between -100 and 100. If the brightness is negative, the image gets darker; if the brightness value is positive the image gets brighter.

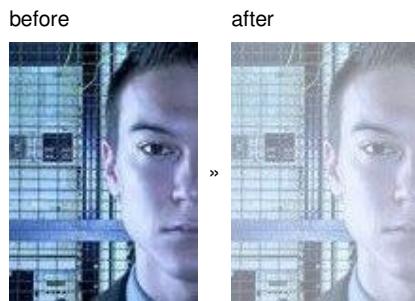
Syntax

```
<cfx_JImage action="AdjustBrightness" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] brightness="[BRIGHTNESS]">
```

Example 1

```
<cfx_JImage action="AdjustBrightness" file="#baseImage#"  
outfile="#path#example_adjustbrightness_50.jpg" brightness="50">
```

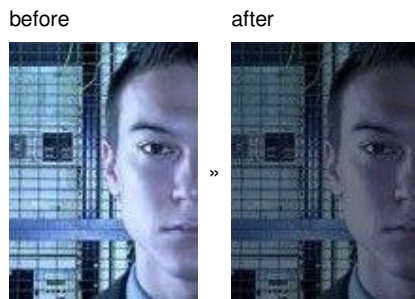
Result 1



Example 2

```
<cfx_JImage action="AdjustBrightness" file="#baseImage#"  
outfile="#path#example_adjustbrightness_-50.jpg" brightness="-50">
```

Result 2



Action="AdjustContrast"

This changes the contrast of the image.

The required "contrast" attribute must be an integer value between -100 and 100.

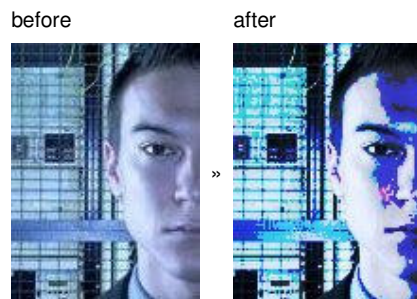
Syntax

```
<cfx_JImage action="AdjustContrast" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] contrast="[contrast]">
```

Example 1

```
<cfx_JImage action="AdjustContrast" file="#baseImage#"  
outfile="#path#example_adjustcontrast_50.jpg" contrast="50">
```

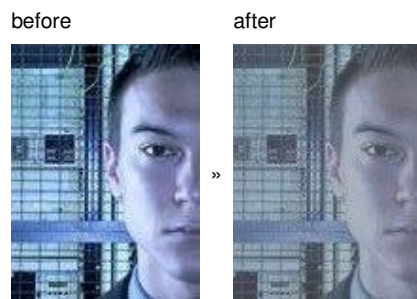
Result 1



Example 2

```
<cfx_JImage action="Adjustcontrast" file="#baseImage#"  
outfile="#path#example_adjustcontrast_-50.jpg" contrast="-50">
```

Result 2



Action="GammaCorrection"

This applies gamma correction to an image.
The required "gamma" attribute must be an decimal number value between 0.1 and 10.0.

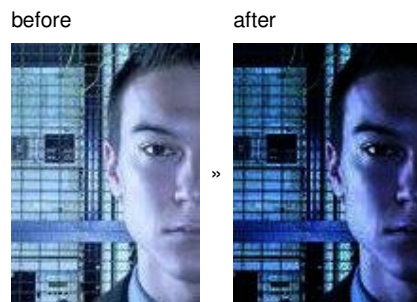
Syntax

```
<cfx_JImage action="GammaCorrection" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] gamma="[GAMMA_CORRECTION_VALUE]">
```

Example 1

```
<cfx_JImage action="GammaCorrection" file="#baseImage#"  
outfile="#path#example_gammaCorrection_50.jpg" gamma="0.36">
```

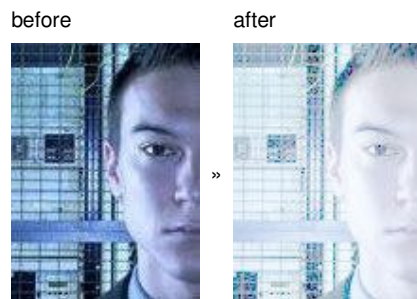
Result 1



Example 2

```
<cfx_JImage action="gammaCorrection" file="#baseImage#"  
outfile="#path#example_gammaCorrection_4.8.jpg" gamma="4.8">
```

Result 2



Action="Grayscale"

Makes an image gray.

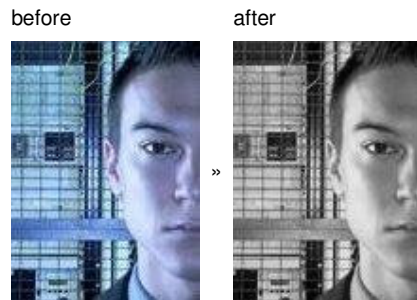
Syntax

```
<cfx_JImage action="grayscale" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"]>
```

Example

```
<cfx_JImage action="grayscale" file="#baseImage#"  
outfile="#path#example_grayscale.gif">
```

Result



Action="Invert"

Inverts the colors of an image.

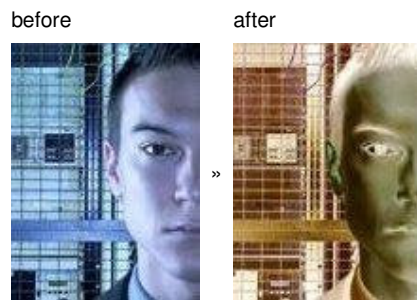
Syntax

```
<cfx_JImage action="Invert" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"]>
```

Example

```
<cfx_JImage action="Invert" file="#baseImage#" outfile="#path#example_invert.png">
```

Result



Filters

Action="Filter"

The "filter" action takes the image and applies a any one of a number of filter types to the image. The attribute "filter" is required. The following filters are supported:

- Blur
- Edge detect
- Emboss
- Horizontal Prewitt
- Horizontal Sobel
- Lithograph
- Maximum
- Mean
- Median
- Minimum
- Oil
- Psychedelic Distillation
- Sharpen
- Vertical Prewitt
- Vertical Sobel

Syntax

```
<cfx_JImage action="filter" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] filter="[FILTER_TO_APPLY]">
```

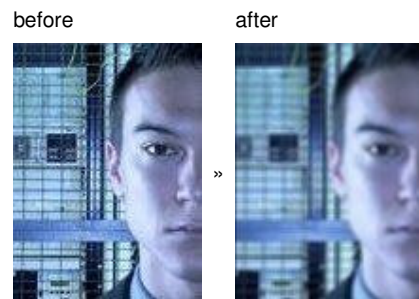
Action="Filter", Filter="Blur"

This filter applies a blur effect on the image.

Blur Filter Example

```
<cfx_JImage action="filter" filter="blur" file="#baseImage#"
outfile="#path#example_filter_blur.jpg">
```

Blur Filter Result



Action="Filter", Filter="Edge Detect"

This filter applies an edge detect filter on the image.

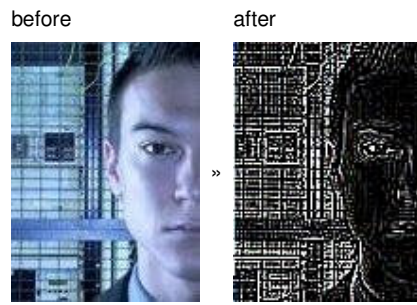
Syntax

```
<cfx_JImage action="filter" filter="edge detect" file="#baseImage#"
outfile="#path#example_filter_blur.jpg">
```

Edge Detect Filter Example

```
<cfx_JImage action="filter" filter="edge detect" file="#baseImage#"
outfile="#path#example_filter_edgeDetect.jpg">
```

Edge Detect Filter Result



Action="Filter", Filter="Emboss"

This filter applies an edge detect filter on the image.

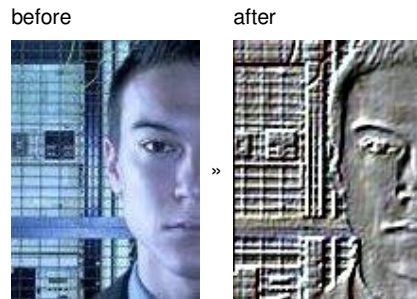
Syntax

```
<cfx_JImage action="filter" filter="emboss" file="#baseImage#"
outfile="#path#example_filter_emboss.jpg">
```

Edge Detect Filter Example

```
<cfx_JImage action="filter" filter="emboss" file="#baseImage#"
outfile="#path#example_filter_emboss.jpg">
```

Emboss Filter Result



Action="Filter", Filter="Horizontal Prewitt"

This filter applies a horizontal prewitt filter on the image.

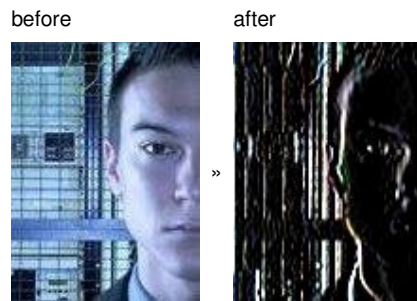
Syntax

```
<cfx_JImage action="filter" filter="Horizontal Prewitt" file="#baseImage#"
outfile="#path#example_filter_horizontal_prewitt.jpg">
```

Horizontal Prewitt Filter Example

```
<cfx_JImage action="filter" filter="Horizontal Prewitt" file="#baseImage#"
outfile="#path#example_filter_horizontal_prewitt.jpg">
```

Horizontal Prewitt Filter Result



Action="Filter", Filter="Horizontal Sobel"

This filter applies a horizontal sobel filter on the image.

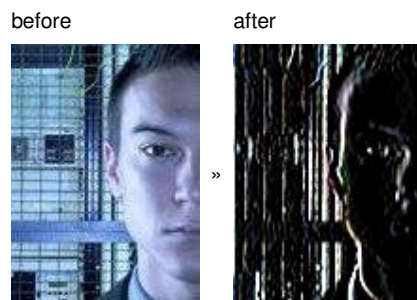
Syntax

```
<cfx_JImage action="filter" filter="Horizontal Sobel" file="#baseImage#"
outfile="#path#example_filter_horizontal_sobel">
```

Horizontal Sobel Filter Example

```
<cfx_JImage action="filter" filter="Horizontal Sobel" file="#baseImage#"
outfile="#path#example_filter_horizontal_sobel">
```

Horizontal Prewitt Filter Result



Action="Filter", Filter="Lithograph"

This filter applies a lithograph filter on the image.

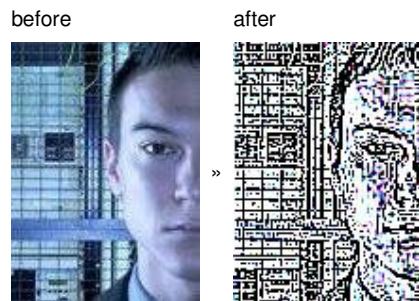
Syntax

```
<cfx_JImage action="filter" filter="Lithograph" file="#baseImage#"
outfile="#path#example_filter_lithograph.jpg">
```

Lithograph Filter Example

```
<cfx_JImage action="filter" filter="Lithograph" file="#baseImage#"
outfile="#path#example_filter_lithograph.jpg">
```

Lithograph Filter Result



Action="Filter", Filter="Maximum"

This filter applies a maximum filter on the image.

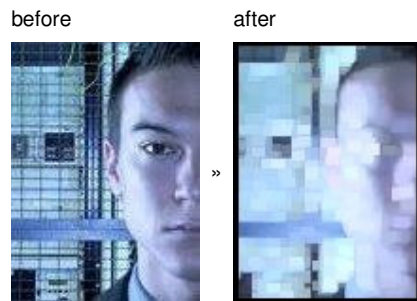
Syntax

```
<cfx_JImage action="filter" filter="maximum" file="#baseImage#"
outfile="#path#example_filter_maximum.jpg">
```

Maximum Filter Example

```
<cfx_JImage action="filter" filter="maximum" file="#baseImage#"
outfile="#path#example_filter_maximum.jpg">
```

Maximum Filter Result



Action="Filter", Filter="Mean"

This filter applies a mean filter on the image.

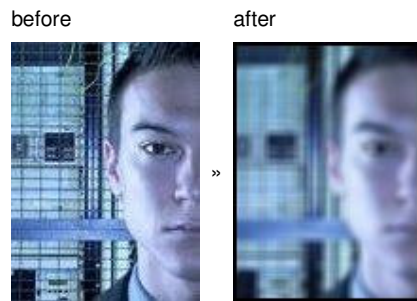
Syntax

```
<cfx_JImage action="filter" filter="mean" file="#baseImage#"
outfile="#path#example_filter_mean.jpg">
```

Mean Filter Example

```
<cfx_JImage action="filter" filter="mean" file="#baseImage#"
outfile="#path#example_filter_mean.jpg">
```

Mean Filter Result



Action="Filter", Filter="Median"

This filter applies a median filter on the image.

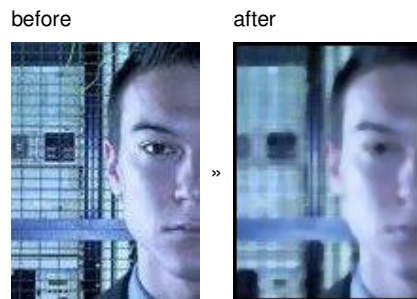
Syntax

```
<cfx_JImage action="filter" filter="median" file="#baseImage#"
outfile="#path#example_filter_median.jpg">
```

Median Filter Example

```
<cfx_JImage action="filter" filter="median" file="#baseImage#"
outfile="#path#example_filter_median.jpg">
```

Median Filter Result



Action="Filter", Filter="Minimum"

This filter applies a minimum filter on the image.

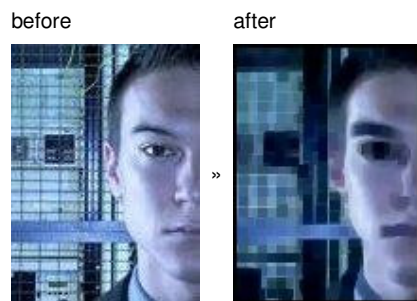
Syntax

```
<cfx_JImage action="filter" filter="minimum" file="#baseImage#"
outfile="#path#example_filter_minimum.jpg">
```

Minimum Filter Example

```
<cfx_JImage action="filter" filter="minimum" file="#baseImage#"
outfile="#path#example_filter_minimum.jpg">
```

Minimum Filter Result



Action="Filter", Filter="Oil"

This filter applies an oil filter on the image.

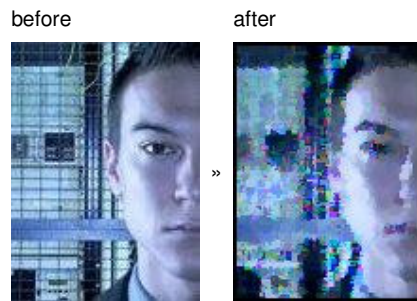
Syntax

```
<cfx_JImage action="filter" filter="oil" file="#baseImage#"
  outfile="#path#example_filter_oil.jpg">
```

Oil Filter Example

```
<cfx_JImage action="filter" filter="oil" file="#baseImage#"
  outfile="#path#example_filter_oil.jpg">
```

Oil Filter Result



Action="Filter", Filter="Psychedelic Distillation"

This filter applies a psychedelic distillation filter on the image.

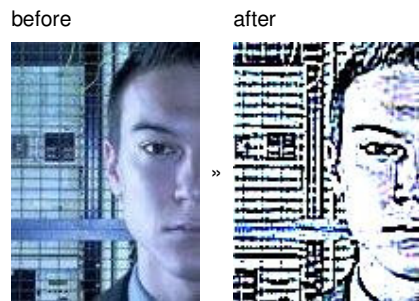
Syntax

```
<cfx_JImage action="filter" filter="psychedelic distillation" file="#baseImage#"
outfile="#path#example_filter_psychedelic_distillation.jpg">
```

Psychedelic Distillation Filter Example

```
<cfx_JImage action="filter" filter="psychedelic distillation" file="#baseImage#"
outfile="#path#example_filter_psychedelic_distillation.jpg">
```

Psychedelic Distillation Filter Result



Action="Filter", Filter="Sharpen"

This filter applies a sharpen filter on the image.

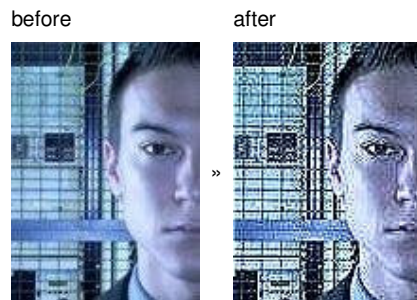
Syntax

```
<cfx_JImage action="filter" filter="sharpen" file="#baseImage#"
outfile="#path#example_filter_sharpen.jpg">
```

Sharpen Filter Example

```
<cfx_JImage action="filter" filter="sharpen" file="#baseImage#"
outfile="#path#example_filter_sharpen.jpg">
```

Sharpen Filter Result



Action="Filter", Filter="Vertical Prewitt"

This filter applies a vertical prewitt filter on the image.

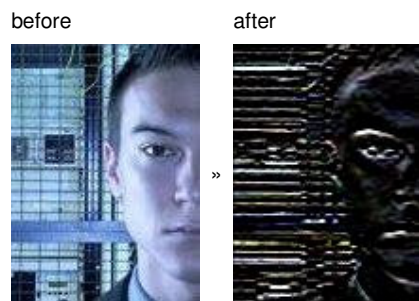
Syntax

```
<cfx_JImage action="filter" filter="vertical prewitt" file="#baseImage#"
outfile="#path#example_filter_vertical_prewitt.jpg">
```

Vertical Prewitt Filter Example

```
<cfx_JImage action="filter" filter="vertical prewitt" file="#baseImage#"
outfile="#path#example_filter_vertical_prewitt.jpg">
```

Vertical Prewitt Filter Result



Action="Filter", Filter="Vertical Sobel"

This filter applies a vertical sobel filter on the image.

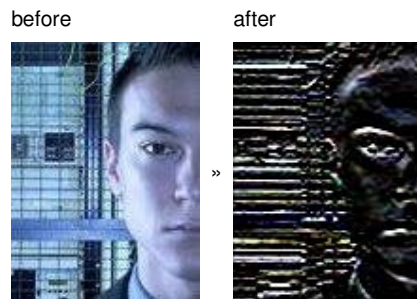
Syntax

```
<cfx_JImage action="filter" filter="vertical sobel" file="#baseImage#"
outfile="#path#example_filter_vertical_sobel.jpg">
```

Vertical Sobel Filter Example

```
<cfx_JImage action="filter" filter="vertical sobel" file="#baseImage#"
outfile="#path#example_filter_vertical_sobel.jpg">
```

Vertical Sobel Filter Result



Miscellaneous

Action="DrawText"

The "drawtext" action allows you to draw text on top of an image.

The "x" and "y" attributes are used to specify where on the image the text should be drawn from and correspond to the bottom left corner of the first character. The default "x" and "y" attribute values are 0 and 20 respectively.



Antialiasing is off by default. You can turn on antialiasing by passing antialiasing="yes".

if the specified font cannot be found, the default system font will be used instead.

Using the "transparency" attribute you can make the text partially transparent. The default value for transparency is 0.

Use the "color" attribute to specify the color to use when drawing the font. This value must be passed using the standard HTML color hexadecimal format such as "#00FF00" for red. The default color is black.

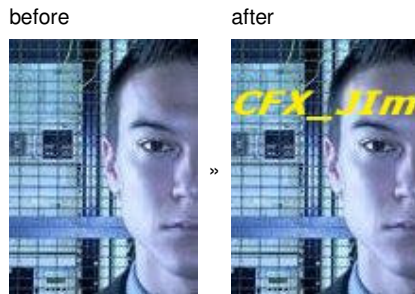
Syntax

```
<cfx_JImage action="drawText" file="FULL_PATH_TO_FILE"
[outfile="FULL_PATH_TO_FILE"] text="[TEXT_TO_WRITE]" [x="X_OFFSET"]
[y="[Y_OFFSET]"] [font="FONT_TO_USE"] [fontSize="FONT_SIZE"]
[fontItalic="BOOLEAN_FONT_ITALIC"] [color="HTML_COLOR_CODE"]
[transparency="TRANSPARENCY_VALUE"] [antialiasing="YES/NO"]
>
```

Example

```
<cfx_JImage action="drawText" file="#baseImage#"
outfile="#path#example_drawText_JImage_Rocks.jpg" text="CFX_JImage Rocks!" x="0"
y="40" font="verdana" fontsize="18" color="##FEEF00" fontbold="yes" fontitalic="yes"
antialiasing="yes" >
```

Result



Action="Border"

The "border" action draws a rectangular border around the image.

The "size" attributes is used to specify the size of the border to be drawn and defaults to 0. The color of the border can be passed using the "color" attribute. Also the transparency of the border can be specified.

Using the "transparency" attribute you can make the text partially transparent. The default value for transparency is 0.

Use the optional "color" attribute to specify the color to use when drawing. This value must be passed using the standard HTML color hexadecimal format such as "#00FF00" for red. The default color is black.

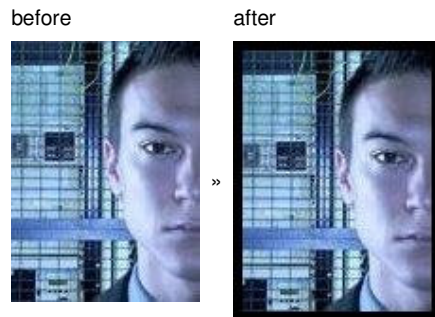
Syntax

```
<cfx_JImage action="border" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] [Size="BORDER_SIZE"]  
[color="HTML_COLOR_CODE"]  
[transparency="TRANSPARENCY_VALUE"]  
>
```

Example

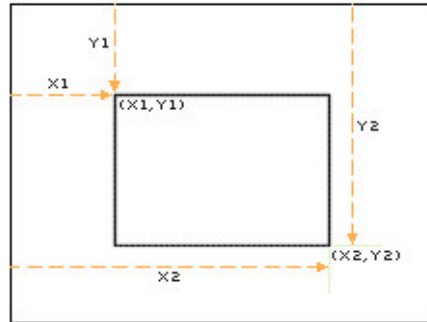
```
<cfx_JImage action="border" file="#baseImage#" outfile="#path#example_border.jpg"  
size="2">
```

Result



Action="DrawRect"

The "drawrect" action draws an empty rectangle on the image.



Using the "transparency" attribute you can make the shape partially transparent. The default value for transparency is 0.

Use the optional "color" attribute to specify the color to use when drawing the shape. This value must be passed using the standard HTML color hexadecimal format such as "#00FF00" for red. The default color is black.

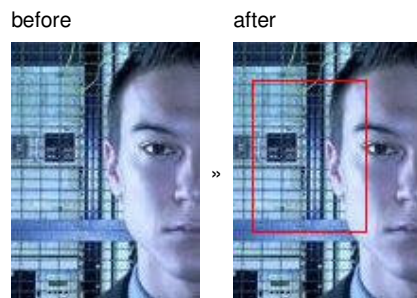
Syntax

```
<cfx_JImage action="drawrect" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] [x1="X1_OFFSET"] [y1="Y1_OFFSET"]  
[x2="X2_OFFSET"] [y2="Y2_OFFSET"] [color="HTML_COLOR_CODE"]  
[transparency="TRANSPARENCY_VALUE"]  
>
```

Example

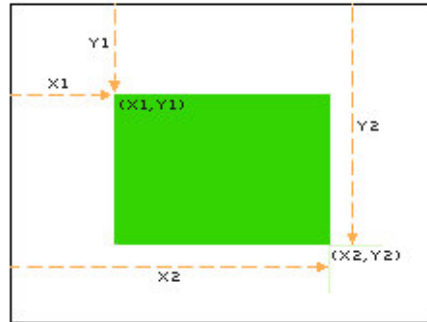
```
<cfx_JImage action="drawRect" file="#baseImage#"  
outfile="#path#example_drawRect.jpg" x1="10" y1="20" x2="60" y2="80"  
color="#FF0000">
```

Result



Action="FillRect"

The "fillrect" action draws a filled rectangle on the image.



Using the "transparency" attribute you can make the shape partially transparent. The default value for transparency is 0.

Use the optional "color" attribute to specify the color to use when drawing the shape. This value must be passed using the standard HTML color hexadecimal format such as "#00FF00" for red. The default color is black.

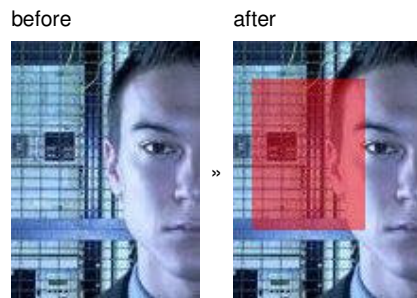
Syntax

```
<cfx_JImage action="fillrect" file="FULL_PATH_TO_FILE"
[outfile="FULL_PATH_TO_FILE"] [x1="X1_OFFSET"] [y1="[Y1_OFFSET]"]
[x2="X2_OFFSET"] [y2="Y2_OFFSET"] [color="HTML_COLOR_CODE"]
[transparency="TRANSPARENCY_VALUE"]
>
```

Example

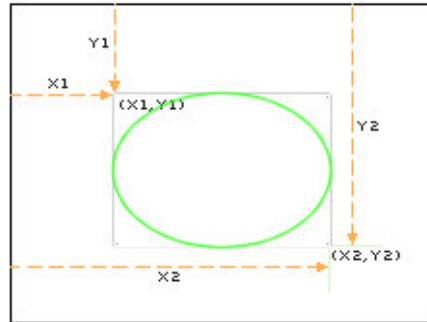
```
<cfx_JImage action="fillRect" file="#baseImage#" outfile="#path#example_fillRect.jpg"
x1="10" y1="20" x2="60" y2="80" color="#FF0000" transparency="50">
```

Result



Action="DrawOval"

The "drawoval" action draws an empty oval on the image.



Using the "transparency" attribute you can make the shape partially transparent. The default value for transparency is 0.

Use the optional "color" attribute to specify the color to use when drawing the shape. This value must be passed using the standard HTML color hexadecimal format such as "#00FF00" for red. The default color is black.

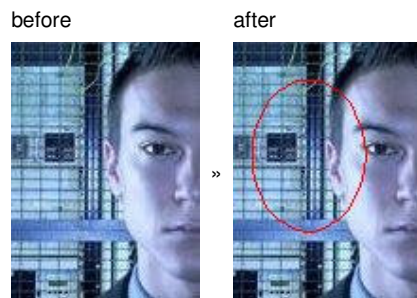
Syntax

```
<cfx_JImage action="drawoval" file="FULL_PATH_TO_FILE"
[outfile="FULL_PATH_TO_FILE"] [x1="X1_OFFSET"] [y1="[Y1_OFFSET]"]
[x2="X2_OFFSET"] [y2="Y2_OFFSET"] [color="HTML_COLOR_CODE"]
[transparency="TRANSPARENCY_VALUE"]
>
```

Example

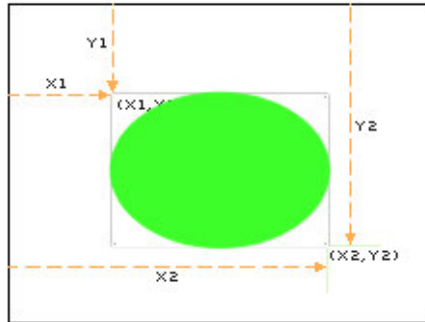
```
<cfx_JImage action="drawOval" file="#baseImage#"
outfile="#path#example_drawOval.jpg" x1="10" y1="20" x2="60" y2="80"
color="#00FF00">
```

Result



Action="FillOval"

The "filloval" action draws a filled oval on the image.



Using the "transparency" attribute you can make the shape partially transparent. The default value for transparency is 0.

Use the optional "color" attribute to specify the color to use when drawing the shape. This value must be passed using the standard HTML color hexadecimal format such as "#00FF00" for red. The default color is black.

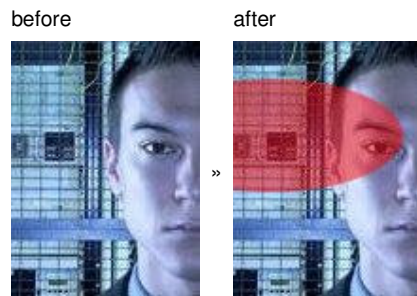
Syntax

```
<cfx_JImage action="filloval" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] [x1="X1_OFFSET"] [y1="[Y1_OFFSET]"]  
[x2="X2_OFFSET"] [y2="Y2_OFFSET"] [color="HTML_COLOR_CODE"]  
[transparency="TRANSPARENCY_VALUE"]  
>
```

Example

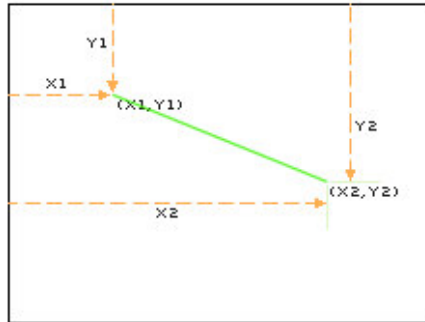
```
<cfx_JImage action="fillOval" file="#baseImage#" outfile="#path#example_fillOval.jpg"  
x1="10" y1="20" x2="60" y2="80" color="#00FF00">
```

Result



Action="DrawLine"

The "drawLine" action draws a line on the image.



Using the "transparency" attribute you can make the shape partially transparent. The default value for transparency is 0.

Use the optional "color" attribute to specify the color to use when drawing the line. This value must be passed using the standard HTML color hexadecimal format such as "#00FF00" for red. The default color is black.

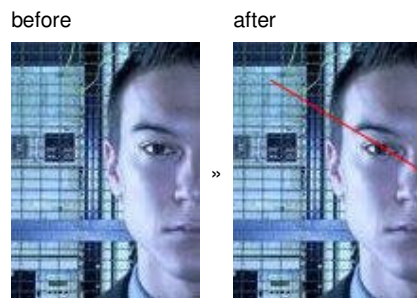
Syntax

```
<cfx_JImage action="drawline" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] [x1="X1_OFFSET"] [y1="[Y1_OFFSET]"]  
[x2="X2_OFFSET"] [y2="Y2_OFFSET"] [color="HTML_COLOR_CODE"]  
[transparency="TRANSPARENCY_VALUE"]  
>
```

Example

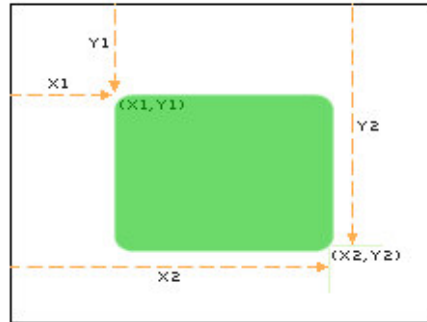
```
<cfx_JImage action="drawLine" file="#baseImage#"  
outfile="#path#example_drawLine.jpg" x1="10" y1="20" x2="60" y2="80"  
color="#00FF00">
```

Result



Action="FillRoundRect"

The "fillRoundRect" action draws a filled rounded rectangle on the image.



Use the "size" attribute to specify how round the rounded edges should be. The default "size" is 20.

Using the "transparency" attribute you can make the shape partially transparent. The default value for transparency is 0.

Use the optional "color" attribute to specify the color to use when drawing the line. This value must be passed using the standard HTML color hexadecimal format such as "#00FF00" for red. The default color is black.

Syntax

```
<cfx_JImage action="fillRoundRect" file="FULL_PATH_TO_FILE"  
[outfile="FULL_PATH_TO_FILE"] [x1="X1_OFFSET"] [y1="[Y1_OFFSET]"]  
[x2="X2_OFFSET"] [y2="Y2_OFFSET"] [color="HTML_COLOR_CODE"]  
[transparency="TRANSPARENCY_VALUE"]  
>
```

Example

```
<cfx_JImage action="fillRoundRect" file="#baseImage#"  
outfile="#path#example_fillRoundRect.jpg" x1="10" y1="20" x2="60" y2="80"  
color="#00FF00" transparency="50">
```

Result

