

# JAVA DIP - ENHANCING IMAGE BRIGHTNESS

[http://www.tutorialspoint.com/java\\_dip/enhancing\\_image\\_brightness.htm](http://www.tutorialspoint.com/java_dip/enhancing_image_brightness.htm)

Copyright © tutorialspoint.com

In this chapter we enhance the brightness of an image by multiplying each pixel of the image with an alpha value and adding another beta value to it.

We **OpenCV** function **convertTo** that does the above operation automatically. It can be found under **Mat** package. Its syntax is given below:

```
int alpha = 2;  
int beta = 50;  
sourceImage.convertTo(destination, rtype , alpha, beta);
```

The parameters are described below:

Sr.No.	Parameters
1	<b>destination</b> It is destination image.
2	<b>rtype</b> It is desired output matrix type or, rather the depth, since the number of channels are the same as the input has. if rtype is negative, the output matrix will have the same type as the input.
3	<b>alpha</b> It is optional scale factor.
4	<b>beta</b> It is optional delta added to the scaled values.

Apart from the convertTo method, there are other methods provided by the Mat class. They are described briefly:

Sr.No.	Methods
1	<b>adjustROI</b> <i>intdtop, intdbottom, intdleft, intdright</i> It adjusts a submatrix size and position within the parent matrix.
2	<b>copyTo</b> <i>Matm</i> It copies the matrix to another one.
3	<b>diag</b>

It extracts a diagonal from a matrix, or creates a diagonal matrix.

4

**dot***Matm*

It computes a dot-product of two vectors.

5

**reshape***intcn*

It changes the shape and/or the number of channels of a 2D matrix without copying the data.

6

**submat***RangerowRange, RangecolRange*

It extracts a rectangular sub matrix.

## Example

The following example demonstrates the use of Mat class to enhance brightness of an image:

```
import org.opencv.core.Core;
import org.opencv.core.Mat;
import org.opencv.highgui.Highgui;

public class Main {
    static int width;
    static int height;
    static double alpha = 2;
    static double beta = 50;

    public static void main( String[] args ) {

        try{
            System.loadLibrary( Core.NATIVE_LIBRARY_NAME );
            Mat source =
Highgui.imread("digital_image_processing.jpg",Highgui.CV_LOAD_IMAGE_COLOR);
            Mat destination = new Mat(source.rows(),source.cols(),

            source.type());
            source.convertTo(destination, -1, alpha, beta);
            Highgui.imwrite("brightWithAlpha2Beta50.jpg", destination);

        }catch (Exception e) {
            System.out.println("error: " + e.getMessage());
        }
    }
}
```

## Output

When you execute the given code, the following output is seen:

### Original Image





Enhanced Bright Image **Alpha=1 & Beta=50**



Enhanced Bright Image **Alpha=2 & Beta=50**





Loading [MathJax]/jax/output/HTML-CSS/jax.js