

JAVA DIP - IMAGE COMPRESSION TECHNIQUE

http://www.tutorialspoint.com/java_dip/image_compression_technique.htm

Copyright © tutorialspoint.com

An image can easily be compressed and stored through Java. Compression of image involves converting an image into jpg and storing it.

In order to compress an image, we read the image and convert into BufferedImage object.

Further, we get an ImageWriter from **getImageWritersByFormatName** method found in the ImageIO class. From this ImageWriter, create an **ImageWriteParam** object. Its syntax is given below:

```
Iterator<ImageWriter> list = ImageIO.getImageWritersByFormatName("jpg");
ImageWriteParam obj = writer_From_List.getDefaultWriteParam();
```

From this ImageWriteParam object, you can set the compression by calling these two methods which are **setCompressionMode** and **setCompressionQuality**. Their syntaxes are as given below:

```
obj.setCompressionMode(ImageWriteParam.MODE_EXPLICIT);
obj.setCompressionQuality(0.05f);
```

The setCompressionMode method takes Mode_EXPLICIT as the parameter. Some of the other MODES are described briefly:

Sr.No.	Modes
1	MODE_DEFAULT It is a constant value that may be passed into methods to enable that feature for future writes.
2	MODE_DISABLED It is a constant value that may be passed into methods to disable that feature for future writes.
3	MODE_EXPLICIT It is a constant value that may be passed into methods to enable that feature for future writes.

Apart from the compressions methods, there are other methods provided by the ImageWriteParam class. They are described briefly:

Sr.No.	Methods
1	canOffsetTiles It returns true if the writer can perform tiling with non-zero grid offsets while writing.
2	getBitRate <i>floatquality</i>

It returns a float indicating an estimate of the number of bits of output data for each bit of input image data at the given quality level.

3

getLocale

It returns the currently set Locale, or null if only a default Locale is supported.

4

isCompressionLossless

It returns true if the current compression type provides lossless compression.

5

unsetCompression

It removes any previous compression type and quality settings.

6

unsetTiling

It removes any previous tile grid parameters specified by calls to setTiling.

Example

The following example demonstrates the use of ImageWriteParam class to compress an image:

```
import java.io.*;
import java.util.*;
import java.awt.image.*;

import javax.imageio.*;
import javax.imageio.stream.*;

class Compression {

    public static void main(String[] args) throws IOException {

        File input = new File("digital_image_processing.jpg");
        BufferedImage image = ImageIO.read(input);

        File compressedImageFile = new File("compress.jpg");
        OutputStream os = new FileOutputStream(compressedImageFile);

        Iterator<ImageWriter> writers = ImageIO.getImageWritersByFormatName("jpg");
        ImageWriter writer = (ImageWriter) writers.next();

        ImageOutputStream ios = ImageIO.createImageOutputStream(os);
        writer.setOutput(ios);

        ImageWriteParam param = writer.getDefaultWriteParam();

        param.setCompressionMode(ImageWriteParam.MODE_EXPLICIT);
        param.setCompressionQuality(0.05f);
        writer.write(null, new IIOMemoryImage(image, null, null), param);

        os.close();
        ios.close();
        writer.dispose();
    }
}
```

Output

When you execute the given code, it compresses the image **digital_image_processing.jpg** to its equivalent compressed image and writes it on the hard disk with the name **compress.jpg**.

Original Image



Compressed Image - Quality Factor: 0.05



Compressed Image - Quality Factor: 0.5





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