

# ImageJ/Java Cheat Sheet

© 2018 Wilhelm Burger, <http://www.imagingbook.com> (October 19, 2018)

## Images

### Types of images (image “processors”)

**ImageProcessor** (class): super-class (abstract)  
**ByteProcessor** (class): unsigned 8-bit values (0, ..., 255)  
**ShortProcessor** (class): unsigned 16-bit values (0, ..., 65535)  
**FloatProcessor** (class): 32-bit floating-point values  
**ColorProcessor** (class): 24-bit RGB color values (int)

### Creating images (constructors)

```
ByteProcessor(int w, int h)
ByteProcessor(int w, int h, byte[] pixels)
ByteProcessor(java.awt.Image im)

ShortProcessor(int w, int h)
ShortProcessor(int w, int h, short[] p, ColorModel c)
ShortProcessor(java.awt.BufferedImage im)

FloatProcessor(int w, int h)
FloatProcessor(int w, int h, float[] pixels)
FloatProcessor(int w, int h, double[] pixels)
FloatProcessor(float[][] pixels)
FloatProcessor(int[][] pixels)

ColorProcessor(int w, int h)
ColorProcessor(int w, int h, int[] pixels)
ColorProcessor(java.awt.Image im)
```

### Creating from / duplicating an existing image – **ImageProcessor**

```
ImageProcessor createProcessor(int w, int h)
ImageProcessor duplicate()
```

### Converting images – **ImageProcessor**

```
ByteProcessor convertToByteProcessor()
ByteProcessor convertToByteProcessor(boolean scale)
ShortProcessor convertToShortProcessor()
ShortProcessor convertToShortProcessor(boolean scale)
FloatProcessor convertToFloatProcessor()
ColorProcessor convertToColorProcessor()
```

## Image parameters – `ImageProcessor`

```
int getHeight()  
int getWidth()  
int getPixelCount()  
int getInterpolationMethod()  
void setInterpolationMethod(int method)
```

## Displaying images (example)

```
ImageProcessor ip = new ByteProcessor(400, 300);  
new ImagePlus("The Title", ip).show();
```

## Accessing pixels

### Single-pixel access – `ImageProcessor`

```
int get(int index)  
void set(int index, int val)  
int get(int x, int y)  
void set(int x, int y, int val)  
float getf(int index)  
float setf(int index, float val)  
float getf(int x, int y)  
float setf(int x, int y, float val)  
int getPixel(int x, int y)  
void putPixel(int x, int y, int val)  
int[] getPixel(int x, int y, int[] rgb)  
void putPixel(int x, int y, int[] rgb)  
float getPixelValue(int x, int y)  
void putPixelValue(int x, int y, double val)  
double getInterpolatedPixel(double x, double y)
```

### Multi-pixel access – `ImageProcessor`

```
Object getPixels() – needs cast to num. 1D array  
Object getPixelsCopy()  
void setPixels(Object pixels)  
int[][] getIntArray()  
float[][] getFloatArray()  
void getColumn(int x, int y, int[] data, int length)  
void putColumn(int x, int y, int[] data, int length)  
void getRow(int x, int y, int[] data, int length)  
void putRow(int x, int y, int[] data, int length)  
float[] getRow(int x, int y, float[] data, int length)  
void putRow(int x, int y, float[] data, int length)  
double[] getLine(double x1, y1, x2, y2)  
void insert(ImageProcessor ip, int x, int y)
```

### RGB color images only – [ColorProcessor](#)

```
void getRGB(byte[] R, byte[] G, byte[] B)
void setRGB(byte[] R, byte[] G, byte[] B)
void getHSB(byte[] H, byte[] S, byte[] B)
void setHSB(byte[] H, byte[] S, byte[] B)
FloatProcessorgetBrightness()
void setBrightness(FloatProcessorfp)
```

## Image Stacks

### Retrieving/setting stacks – [ImagePlus](#)

```
ImageStack createEmptyStack()
ImageStack getImageStack()
ImageStack getStack()
int getStackSize()
void setStack(String title, ImageStack stack)
```

### Creating new stacks – [ImageStack](#)

```
ImageStack(int w, int h)
ImageStack(int w, int h, int size)
ImageStack(int w, int h, ColorModel cm)
static ImageStack create(int w, int h, int size, int depth)
```

### Querying stack properties – [ImageStack](#)

```
int getWidth()
int getHeight()
int getSize()
ColorModel getColorModel()
int getBitDepth()
public boolean isRGB()
public boolean isHSB()
public boolean isLab()
boolean isVirtual()
String getSliceLabel(int n)
String[] getSliceLabels()
String getShortSliceLabel(int n)
```

### Stack operations – [ImageStack](#)

```
void addSlice(ImageProcessor ip)
void addSlice(String label, ImageProcessor ip)
void addSlice(String label, ImageProcessor ip, int n)
void deleteSlice(int n)
void deleteLastSlice()
ImageProcessor getProcessor(int n)
void setProcessor(ImageProcessor ip, int n)
void update(ImageProcessor ip)
```

## Handling images and windows

### Creating/duplicating images – `ImagePlus`

```
ImagePlus(String title, ImageProcessor ip)
ImagePlus(String title, ImageStack st)
ImagePlus duplicate()
```

### Image properties – `ImagePlus`

```
int getBitDepth()
int getHeight()
int getWidth()
int getID()
FileInfo getFileInfo()
FileInfo getOriginalFileInfo()
```

### Displaying/updating images – `ImagePlus`

```
int show()
int show(String title)
int updateAndDraw()
int updateAndRepaintWindow()
```

### Locking/unlocking images – `ImagePlus`

```
boolean lock()
boolean lockSilently()
void unlock()
```

### Clipboard operations – `ImagePlus`

```
void copy(boolean cut)
void paste()
static ImagePlus getClipboard()
```

### Window management – `WindowManager`

```
static boolean closeAllWindows()
static ImagePlus getCurrentImage()
static ImageWindow getCurrentWindow()
static ImagePlus getImage(int imageID)
static ImagePlus getImage(String title)
static int getImageCount()
static ImagePlus getTempCurrentImage()
static int getWindowCount()
static void putBehind()
static void repaintImageWindows()
static void setCurrentWindow(ImageWindow win)
static void setTempCurrentImage(ImagePlus im)
```

## Loading/saving images – IJ

```
static void open()  
static void open(String path)  
static ImagePlus openImage(String path)  
static void save(String path)  
static void saveAs(String format, String path)
```

## Image statistics

### Histograms – ImageProcessor

```
int[] getHistogram()  
double getHistogramMax()  
double getHistogramMin()  
int getHistogramSize()  
void setHistogramRange(double min, double max)  
int setHistogramSize(int size)
```

### General statistics – ImageProcessor

```
ImageStatistics getStats()
```

### Image display, lookup tables – ImageProcessor

```
double getmaxValue()  
double getminValue()  
void resetMinAndMax()  
void setMinAndMax(double min, double max)  
LUT getLut()  
void setLut(LUT)  
void invertLut()  
boolean isDefaultLut()  
boolean isInvertedLut()  
boolean isColorLut()  
boolean isPseudoColorLut()
```

## Point operations

### Single image – ImageProcessor

```
void abs()  
void add(int value)  
void add(double value)  
void and(int value)  
void applyTable(int[] lut)  
void gamma(double g)  
void log()  
void max(double value)  
void min(double value)  
void multiply(double value)
```

```
void noise(double r)
void or(int value)
void sqr()
void sqrt()
void threshold(int th)
void xor(int value)
```

### Multiple Images – **ImageProcessor**

```
void copyBits(ImageProcessor ip, int xloc, int yloc, int mode)
with mode = ADD, AND, AVERAGE, COPY, COPY_INVERTED, DIFFERENCE, DIVIDE, MAX, MIN, MULTIPLY, OR, SUBTRACT, XOR (defined by interface Blitter)
```

### Filters

#### **ImageProcessor:**

```
void blurGaussian(double sigma)
void convolve(float[] kernel, int w, int h)
void convolve3x3(int[] kernel)
void dilate()
void erode()
void findEdges()
void medianFilter()
void smooth()
void sharpen()
```

#### **GaussianBlur:**

```
void blur(ImageProcessor ip, double radius)
```

#### **RankFilters:**

```
void rank(ImageProcessor ip, double radius, int type)
with type = MEAN, MIN, MAX, VARIANCE, MEDIAN, OUTLIERS, DESPECKLE, OPEN, CLOSE (defined by class RankFilters).
```

### Geometric operations – **ImageProcessor**

```
ImageProcessor crop()
void flipHorizontal()
void flipVertical()
boolean getInterpolate()
double getInterpolatedPixel(double x, double y)
double getInterpolatedRGBPixel(double x, double y)
ImageProcessor resize(int width, int height)
void rotate(double angle)
ImageProcessor rotateLeft()
ImageProcessor rotateRight()
void scale(double xScale, double yScale)
void setBackgroundValue(double value)
void setInterpolate(boolean interpolate)
```

## Graphic/drawing operations – `ImageProcessor`

```
void drawDot(int x, int y)
void drawLine(int x1, int y1, int x2, int y2)
void drawPixel(int x, int y)
void drawRect(int x, int y, int width, int height)
void drawString(String s)
void drawString(String s, int x, int y)
void fill()
void fill(int[] mask)
void getStringWidth(String s)
void insert(ImageProcessor ip, int x, int y)
void lineTo(int x, int y)
void moveTo(int x, int y)
void setAntialiasedText(boolean antialiasedText)
void setClipRect(Rectangle clipRect)
void setColor(Color color)
void setFont(Font font)
void setJustification(int justification)
void setLineWidth(int width)
void setValue(double value)
```

## Regions of interest

### `ImageProcessor`:

```
int[] getMask()
Rectangle getRoi()
void setRoi(Rectangle roi)
void setRoi(int x, int y, int width, int height)
void setMask(int[] mask)
```

### `ImagePlus`:

```
int[] getMask()
Roi getRoi()
void killRoi()
void setRoi(Rectangle roi)
void setRoi(Roi roi)
void setRoi(int x, int y, int width, int height)
```

## System information (`IJ`)

```
static boolean is64Bit()
static boolean isJava16()
static boolean isJava17()
static boolean isJava18()
static boolean isJava19()
```

```
static boolean isLinux()
static boolean isMacintosh()
static boolean isWindows()
static String getFullVersion()
static String getVersion()
static String getDirectory(String target)
static long currentMemory()
static String freeMemory()
static long maxMemory()
static Dimension getScreenSize()
```

---

Color codes used above:

**class/interface**

**constructor method**

**static method**

**non-static method**

**constant**

sample code

---

Links:

<https://imagej.nih.gov/ij/index.html> (ImageJ Home)

<https://imagej.nih.gov/ij/developer/api/index.html> (ImageJ API Docs)

<https://imagej.nih.gov/ij/developer/source/index.html> (ImageJ Source)

<http://imagejdocu.tudor.lu/> (ImageJ Wiki)

<https://imagingbook.com/> (ImagingBook Home)