

Table of contents

Introduction	3
Welcome to JMicroVision.....	3
Program History	4
Sending Bug Reports.....	7
Support and Contacting the Author	8
Getting Started	9
System Requirements	9
Getting Help	9
Optimizing Performance	10
Shortcuts	11
Supported File Types	12
Menu Commands	17
Main window.....	17
File Menu (Main).....	18
Options Menu (Main).....	19
Windows Menu (Main)	20
Help Menu (Main)	20
File Menu (Project)	21
Project Menu (Project).....	22
User Interface.....	23
Project Window	23
Project Toolbar.....	24
Image List Editor	25
Display	26
Lens	28
Multiview.....	28
Image Rectification	29
Project Preferences	30
Data Viewer.....	30
Area Editor	32
Histogram Thresholding.....	33
Image Types.....	34
Coloring Objects.....	35
Labeling	35
Class Editor	36
Feret Diameters	37
Context.....	37
Texture.....	38
Tools.....	41
General Description	41
Note	41
Spatial Calibration	42
1D Measurement	44
2D Measurement	46
Object Extraction.....	47

Magic Wand	50
Classification.....	51
Object Descriptors	52
Point Counting	55
Background	57
Background Descriptors.....	57
Profile	58
Image Factory.....	59
Image Operations	61
Segmentation	61
k-means clustering	62
Watershed.....	63
Arithmetic	64
Logic	65
Classic Filtering	66
Nonlinear Filtering	68
Gradient Filtering.....	69
Binary.....	70
Morphology	71
Pseudo color	72
Channel	73
Enhancement.....	73
Rasterize Drawings.....	74
Plugins.....	75
Glossary.....	76
Index.....	82

Introduction

Welcome to JMicroVision



► User-friendly software for analyzing large images



- JMicroVision is an image analysis toolbox for measuring and quantifying components of high-definition images. The program contains most of the common image processing operations, has an efficient visualization system and innovative features. It contains tools to quantify either manually or automatically.
- JMicroVision has a simple and intuitive user interface with powerful features. Using it does not require to be a specialist in image analysis.
- JMicroVision has been developed especially to analyze high definition images of rock thin sections, but it can easily be used in other domains.

► Main Features

- Read images in TIFF, BMP, FlashPiX, GIF, JPEG, PNG, and PNM formats
- Efficient visualization system
- Quantify components: objects or background
- Object analysis (size, shape, orientation, texture ...)
- Object classification

- Image processing (binary and morphology operations, filtering, segmentation...)
- Image rectification (geometric corrections by control points)
- Digital point counting
- Tools for data collection in one or two dimensions
- Image annotation and description card
- Profile (variation of granulometry, density, objects or background)
- Save all measures, data, calibration and preferences in a single project file

Program History



What's new in the release 1.2.2?

- Fixed issues: 29 to 39 (see <http://www.jmicrovision.com/mantis/>)
- Spatial Calibration: Custom calibration, shift the image origin, change the Y axis direction
- Profile tool optimization
- Create a slide show
- Slide show and Multiview configuration saved in the project file
- Modify object shape: Smooth, polygonal approximation and convex hull
- 1D Measurement: Polyline and freehand line, more parameters, show profile and statistics
- 2D Measurement: More statistics on histogram
- Color Parameter for Note, 1D et 2D Measurement
- Channel operations : IHS transform
- Improvement of morphological operations
- Custom persistent filters for Classical filtering and Morphology
- Custom filters and templates are recovered from the last installed version.
- Background: get real surface
- Plugins: Create your own image processing functions
- Error messages of JMicroVision sent to the file "erroLog.txt"
- Native launcher for Windows, Linux and Solaris (resolves the JRE path and the JRE parameters)
- File association (*.jmv) on Windows
- Online help has been actualized

What's new in the release 1.2.0?

- Fixed issues: 21, 23, 24, 25, 26, 27 and 28 (see <http://www.jmicrovision.com/mantis/>)
- Improvement of the image visualization system
- Time consuming processes in separate threads (stop a process by clicking on its progress bar)
- Multiview (1-8 images) of any image of the project (custom or synchronized zoom)
- Object separation module (based on a new algorithm) (experimental tool)
- Profile in all directions (+ color mode)
- Image rectification by control points (stack images representing the same scene but have a different size, orientation or deformation)
- More shortcuts: arrow keystrokes for moving the image or the selected drawings, mouse wheel for zooming, space bar for moving the image with the hand
- Export project in SVG (Scalable Vector Graphics format)
- Image Factory as new Tool, with a lot of new operations :
 - Arithmetic operations (add, subtract, multiply and divide)
 - Logical operations (not, and, or and xor)
 - Binary (hole fill, border kill, Euclidian distance map, reconstruction) (requires binary images)
 - Morphology with customized structuring element (opening, closing, erosion, dilation, external gradient, internal gradient, white tophat, black tophat)
 - Channel operation (extraction and merging)
 - Convolution filtering (blur, sharpen, edge, Gaussian ... and custom filters)
 - Nonlinear filtering (median, min, max and Kuwahara)
 - Gradient filtering (Sobel, Roberts, Prewitt, Freichen)
 - Enhancement (manual enhancement, automatic and equalized levels, background subtraction)
 - Pseudo color (15 palettes)
 - Segmentation (simple binarization, color and gray intensity threshold, k-means clustering, watershed segmentation by markers)
 - Rasterize drawings (convert vectorial objects to bitmap)
 - Plugin (add your own image processing functions, documentation available soon)
 - Icon before the image name to identify the image type (Color, gray or binary)
- Create an image from data
- New project from another project (import preferences, image operations...)
- Magic wand to create objects (experimental tool)
- Define working areas (ellipse and polygon) that can be used in most tools

- More palettes for coloring objects from a parameter
- Moves automatically the image when drawing hitting a border (option)
- Antialiasing option for drawings
- Round shape lens option
- Fix drawing refreshing with small or high zoom factor
- Calibration module improvement
- Find automatically the images linked to the project when they are in the same directory (if files have been moved)

What's new in the release 1.1.1?

- Fixed issues: 4, 16, 17, 19 and 20 (see <http://www.jmicrovision.com/mantis/>)
- Complete online help
- Java runtime 1.5 update 3 included
- Possibility to use the OpenGL acceleration
- Improved navigation for small or very high images
- Error messages when the project is not saved or loaded properly
- Resolve image paths in the project
- Read 16 and 32 bits per channel images (converted in 8 bits/channel)
- Fixed some minors bugs

What's new in the release 1.1.0?

- Online help (not complete)
- Java runtime 1.5 included
- New icons and shortcuts in the interface
- Define a working area (to get results from it)
- Draw within the lens
- Open images or projects by dragging them inside the workspace
- Reopen a project even when the main image is not available
- 5 automatic thresholds
- Customize the display of date and numbers
- Change the measure unit in real-time (also integrates Anglo-Saxon units)
- More exporting data formats
- Template to save settings of Object Extraction
- Texture descriptors
- Vertical Profile (shows the vertical variation of granulometry, objects or background)

- Image Factory (uses several operators, e.g. median filter to reduce noise of an image)

What's new in the release 1.0.1?

- Resolved issues: 2, 3, 5, 7, 8, 9, 12, 13, 14 and 15 (see <http://www.jmicrovision.com/mantis/>)
- Some other minor bugs resolved (in Classification and Point Counting)
- The tiled TIFF format has been improved (it uses compression, image loading is faster and the display is much more fluid when moving the image)
- Manage the memory allocated to JMicroVision and the cache memory of the image tiles.
- New installer (Installshield X multiplatform)
- Start JMicroVision with or without the console

Version 1.00 Beta - 15th June 2004

First public release on the Internet

January 2004

JMicroVision was released to limited audience for evaluation and testing purposes.

Mars 2002


JMicroVision development started

Sending Bug Reports



If you encounter a problem with the program that may be a result of a programming error, you can report it to the bug tracker system at the following address:
<http://www.jmicrovision.com/mantis/>

While all bug reports are welcome, they need to contain relevant information if they are to be of any use, especially all necessary details on how to repeat the problem.

-  **Note:** Before sending a bug report, always make sure that you are using the latest release, or at least check the [development history](#) or search with the bug tracker system if the issue was already addressed in one of the subsequent releases.

► Guidelines to follow when submitting a bug report

- Please, indicate the version of JMicroVision used. If you have downloaded a version without JRE and JAI, indicate the version of JRE and JAI (find JRE version in the menu Help - About JMicroVision... - System Information - java.vm.version).
- Your operating system type and version.
- Quick summary of your hardware (CPU type, speed, memory installed, video device type, other details that may be relevant to the problem).
- How does the problem exhibit: send the error messages of the errorLog.txt file located in the installed directory of JMicroVision (find it in the menu Help - About

JMicroVision... - System Information - user.dir) or, if the file does not contain error messages, make a short description of what has happened.

- What were the actions preceding the problem, when did the problem occur.
- Any of your own observations related to the problem (anything else you may have noticed).

▶▶ **See also:**

[Program History](#)

Support and Contacting the Author



For downloading the latest release, help and resources, please go to the website.

▶ **Web Site**

- <http://www.jmicrovision.com>
- Bug Reports: <http://www.jmicrovision.com/mantis/>
- Feature Requests: <http://www.jmicrovision.com/mantis/>

▶ **Email Addresses**

- Suggestions or Comments: comments@JMicroVision.com
- Website Issues: webmaster@JMicroVision.com
- Any Issue or Question about JMicroVision: support@JMicroVision.com

▶ **Mailing List**

Subscribe to the mailing list on the website for information about JMicroVision system updates and events (<http://www.jmicrovision.com/download.htm>).

▶ **Mailing Address**

Nicolas Roduit
Department of Geology
13, rue des Maraîchers
1205 Geneva
Switzerland

▶▶ **See also:**

[Sending Bug Reports](#)

Getting Started

System Requirements



- Any platform supporting Java 1.5 and JAI ([Java Advanced Imaging](#)). However Windows, Linux, Solaris and Mac OS X (partially) have a native acceleration that improves the performance.
- 256 MB RAM (512 MB is recommended for handling large images).
- 60-90 MB available disk space (includes Java Virtual Machine).
Only 3.2 MB on Mac OS X (Java Virtual Machine is already included in the system).

▶▶ See also:

[JMicroVision Performance](#)

Getting Help



▶ Help on the Web

For downloading the latest release, help and resources: <http://www.jmicrovision.com>
Notes or issues in the bug tracker system: <http://www.jmicrovision.com/mantis/>

▶ Technical Support

JMicroVision is free, therefore no official technical support is available, but it is possible to submit your questions or issues at support@JMicroVision.com.

▶ Online Application Help

From the JMicroVision **Help** menu, select **Contents & Index**.

Note: Press the Help button in the program to view a help topic directly related to the current window.

Click:	To:
Contents	View the table of contents of the online Help. Click each book to display pages that link to topics, and click each page to display the corresponding topic in the right pane.
Index	Search for specific words or phrases or select from a list of index keywords. Click the keyword to display the corresponding topic in the right pane.
Search	Locate words or phrases within the content of your topics. Type the word or phrase in the text field, press ENTER , and select the topic you want from the list of topics.
Glossary	Display a list of words, short phrases, and their definitions related to JMicroVision. When you select a term from the Term list, its corresponding definition is displayed in Definition .

►► See also:

[Contacting the author](#)


Optimizing Performance




1) Memory management

The **initial Java heap size** sets the amount of memory that is initially reserved for the application, while the **maximum Java heap size** determines the largest amount of memory the application will be allowed to consume. Optimal settings for these options are dependent on several issues such as the Java runtime you are using and the actual physical configuration of your machine.

Open the JMicroVision.ini file located in the install directory with a text editor or start the Configuration Wizard in graphical mode (see above for each platform) and change the values of the heap size. Default values (64 for initial and 384 for maximum) are suitable for a machine with 512 MB of RAM. For instance, if you have a computer with 256 MB of RAM, the values must be changed to 32 MB (initial) and 128 (maximum) to avoid memory swapping. But with 128 MB of maximum heap size, some operations like converting or saving images may result in a "out of memory" message of the Java runtime.

 **Note:** These changes take effect when you restart JMicroVision. Before editing the JMicroVision.ini file, it is safer to make a backup. In Generic or Other Platforms distribution, the JMicroVision.ini does not exist, only the command parameters (-Xms and -Xmx) define the heap size values.

 **Note:** On Mac OS X, the JMicroVision.ini does not exist. Control-click on JMicroVision in the Finder and choose "Show Package Contents". This will open a folder that contains a folder called "Contents". Open the **Info.plist** file in a text editor and change the heap size values (-Xms and -Xmx).

The **cache memory for tiles** is the memory used to store the image tiles for future requests. Each tile can be processed and stored separately. Previously allocated tiles may also be re-used to save memory. Tiling improves the performance and facilitates the use of multiple threads for computation.


When loading an image, JMicroVision asks to convert the image to a **tiled TIFF format**, which is a format with a tiled data structure. As a result, each tile can be loaded separately in memory (without a tiled format, the whole image would be loaded).


The value of the **cache memory for tiles** essentially depends on the maximum Java heap size. The best value is the **maximum Java heap size** minus the memory required by the application (generally 20 to 100 MB, it can be more when converting very large images; cache memory used by images is not included). The **used memory** is the total memory used by the application.

2) Using OpenGL for Java 2D

J2SE 5.0 includes a new OpenGL-based pipeline for Java 2D. This pipeline provides hardware acceleration for simple rendering operations (text, images, lines and filled primitives) as well as those that involve complex transforms, paints, composites and clips. This pipeline is available on Solaris, Linux and Windows platforms (not supported in J2SE 5.0 Release 1 on Mac OS X) and is currently disabled by default (-Dsun.java2d.opengl=false).

To enable the OpenGL acceleration, open the JMicroVision.ini file located in the install directory with a text editor or start the Configuration Wizard in graphical mode (see installation instructions on the website) and change the value "false" by "true" (-Dsun.java2d.opengl=true).

 **Note:** These changes take effect when you restart JMicroVision. Before editing the JMicroVision.ini file, it is safer to make a backup. In Generic or Other Platforms distribution, the JMicroVision.ini does not exist, use the VM option command "-Dsun.java2d.opengl=true".

 **Note:** Some graphics drivers might cause rendering artifacts or very slow rendering in JMicroVision. Try to install the latest OpenGL drivers provided by your graphics hardware manufacturer. If it doesn't work, set back the option to "-Dsun.java2d.opengl=false".

Shortcuts



Shortcut	Action
Arrows	Move the image (when the hand is selected) Move the selected drawings
Delete	Delete the selected drawings or an item of a list

Double left click	Show properties of a drawing or item of a list
Ctrl + A	Select all (drawings, table elements or list elements)
Ctrl + C	Copy table elements
Ctrl + left click	Resize 3d charts
mouse wheel	Zoom in or zoom out any image view
Right click	Open the context menu for the thumbnail, the drawings and the image views
Shift + left click	Rotate 3d charts Add or remove a specific drawing to the selection
Space bar	Activate the hand for moving the image

Supported File Types



1) Supported file formats for loading images

The following formats are supported: **BMP, FlashPIX, GIF, JPEG, PNG, PNM** and **TIFF**. Transparency or alpha channel is removed when loading a format that uses transparency. Images with 16 bits per channel are converted in 8 bits per channel.

1.1 BMP (.bmp)

The BMP reader can read Version 2.x, 3.x and some 4.x BMP images. BMP images with 1, 4, 8, 24 bits can be read with this reader. A support for 16 and 32 bit images has also been implemented, although such images are not very common.

Reading of compressed BMPs is supported. BI_RGB, BI_RLE8, BI_RLE4 and BI_BITFIELDS compressions are handled.

BMP limitations:

- Only the default RGB color space is supported.
- Alpha channels are not supported.

1.2 FlashPIX (.fpx)

A limited FlashPIX reader is provided that is capable of extracting a single resolution from a FlashPIX image file. Only simple FlashPix files are decoded properly.

1.3 GIF (.gif)

GIF files are read using the classes of the Java 2 SDK.

1.4 JPEG (.jpg, .jpeg, .jpe)

JPEG files are read using the classes of the Java 2 SDK.

1.5 PNG (.png)

All PNG files (8 and 24 bit palettes).

1.6 PNM (.pgm, .pnm, .ppm)

Standard PNM files, including PBM, PGM, and PPM images of both ASCII and raw formats.

1.7 TIFF (.tif, .tiff)


The TIFF decoder supports all of the baseline TIFF 6.0 specifications. The TIFF decoder also supports several extensions including CCITT bilevel encodings, Tiled Images, Sample Format (integral and floating point). TIFF G3, G4, PackBits, LZW, JPEG-in-TIFF and DEFLATE compression types are understood by the TIFF decoder.

2) File formats for saving images

2.1 Tiled TIFF (.tif, .tiff)

Tiled TIFF is the best format to visualize or analyze images in JMicroVision. It also keeps information of the acquisition device. Images can be saved in the tiled TIFF format:

- 1) From the file menu [Convert Images to Tiled TIFF](#)
- 2) When they are loaded, if the image length or the image width is larger the 768 pixels and if the format is not already tiled TIFF, a popup dialog asks you to optimize the image.
- 3) Press the save button in the [Image List Editor](#).

 **Note:** The tiled TIFF format is very suitable for large images (see JMicroVision Preferences - [Memory](#)). It also offers a faster rendering comparing to the other formats.

2.2 TIFF (.tif, .tiff)

Standard TIFF format (not tiled) for saving images produced by [Image Factory](#) or for [exporting a snapshot of the project rendering](#).


2.3 PNG (.png)

Image format for saving charts, histograms and the images produced by [Image Factory](#) or for [exporting a snapshot of the project rendering](#).

3) File format for loading and saving projects

3.1 JMicroVision Project (.jmv)

Format of JMicroVision Project files, that includes all the drawings, general descriptions, calibration, image references, image operations and preferences.

 **Note:** The additional descriptors cannot be included ([Context](#), [Feret](#), [Texture](#) and [Background](#)).

4) Vector graphics format for exporting the current project

4.1 Compressed SVG (.svgz)

The SVG (Scalable Vector Graphics) file format is defined in the W3C specification, SVGZ is the "gzip" compression (an open-source variant of Zip compression) of the SVG file. If the displayed image is exported, a directory called "images_ " + "the name of the project" is created and the image will be copied in it.

Most of the major vector graphics editors can read SVG (Adobe Illustrator, Corel Draw, ArcGIS 9.2, GIMP, OpenOffice...).

5) File format for loading and saving a list of classes

5.1 Component list (.pcl)

The Component list file allows loading and saving your own list of components for [Point Counting](#), [Object Extraction](#) or markers in [Image Factory](#).

6) File format for loading and saving a description card

6.1 General Description list (.gdl)

The [General Description](#) list file allows loading and saving your own list of descriptions.

7) File format for loading and saving a list of learning elements

7.1 Learning list (.lng)

The Learning list file allows saving learning elements, so that they can later be applied or added to another [supervised classification](#).

8) File format for loading and saving a calibration

8.1 Calibration ratio (.cal)

The [Calibration](#) ratio file allows loading and saving a specific calibration ratio that corresponds to one image or to a set of images.

9) File format for saving data tables

In [Data Viewer](#), the current table can be saved in one of these formats.

9.1 CSV semicolon separator (.csv)

CSV is a standard format that can be read by software that uses a spreadsheet. Actually, it is a text file where each element of the table is separated by a semicolon and a line by a carriage return.

9.2 ASCII tabulation separator (.txt)

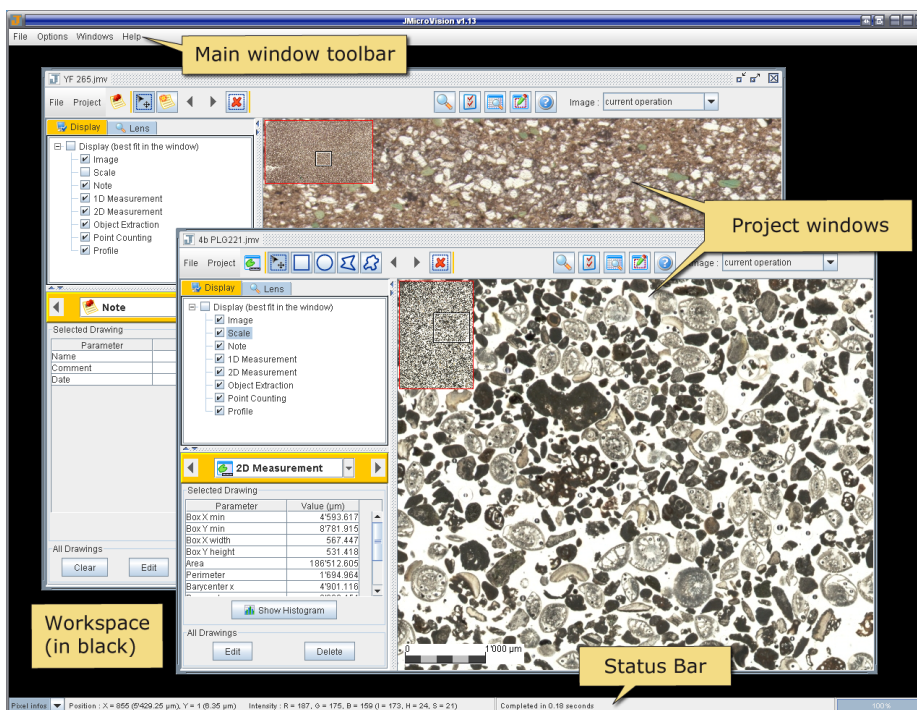
It is a simple text file where each element of the table is separated by tabulation marks and a line by a carriage return.

9.3 XML version 1.0 (.xml)

XML file defined by XML 1.0 specifications.

Menu Commands

Main window



The workspace contains the project windows (internal). All the other kinds of windows (like Data Viewer, Preferences...) can be moved out of the workspace.

1) Main Window Menu


- File Menu
- Options Menu
- Window Menu
- Help Menu

2) Project Window

See **Project Window** in User Interface.

3) Status Bar

On the left side of the status bar, the combo box allows to select between **Status** and **Pixel info** (described below). On the right side, progress bars of time consuming processes are displayed. They disappear at the end of the process. JMicroVision is a multithread application where each long process is embedded in a thread. This means that more than one process can be executed at the same time.



 **Note:** Most of the processes can be stopped by clicking on their progress bar, except for instance loading or saving images.

3.1 Status

Shows unexpected errors.

3.2 Pixel info

Shows the cursor position in pixels, and if calibrated, in the calibration unit (appears in parentheses). It also shows the intensity value of the pixel under the cursor. For color images, RGB values and IHS values in parentheses are displayed, and for gray scale or binary images, only the intensity value appears (see **Image Types**).

 **Note:** The image position (0,0) is located by default in the upper left corner . However, the origin and the direction of the y axis can be modified in the **Spatial Calibration** tool.

File Menu (Main)



1) New Project...


1.1 From Image...

Creates a new project from an image file with the current **default preferences**. At least a valid image path or the width and the height (creates a white image) must be specified. The **calibration** file path is optional.

1.2 From another Project...

Creates a new project with a new image based on another project. A dialog let you choose which data to import to new project.

 **Note:** **Areas** that have a region outside the image are automatically removed.


 **Note:** Some image operations and the backgrounds need to be recalculated. The progress of these processes appears in the status bar.

1.3 From Data...

Creates a new project with the current **default preferences** from a text file containing a data table. The character delimiting the values must be selected. After pressing **OK**, an histogram appears. For redefining the range of the values, enter a new value in the text field **Min** or **Max**, otherwise just close it.

2) Open Project...

Opens an existing project file with a ".jmv" extension.


 **Note:** If the main image link is not valid anymore, a dialog appears to resolve the image path.

 **Note:** It is also possible to open a project by dragging a file in the workspace.

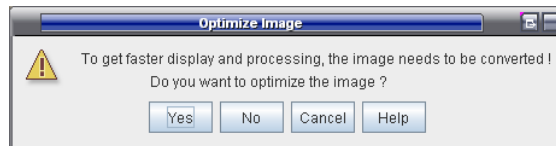
3) Open Image ...

Opens an BMP, FlashPIX, GIF, JPEG, PNG, PNM or TIFF image (see [file formats for loading images](#) for more specifications).



 **Note:** When opening an image, a new **default project** is automatically created.

 **Note:** It is also possible to open an image by dragging a file in the workspace.

If the image width or height is larger than 768 pixels and if the image is not a tiled image, a popup asks you to optimize the image. If you press **Yes**, the image is converted to **Tiled TIFF format** (it is strongly recommended for large images).




4) Reopen

Reopens a project  or an image  previously opened.

5) Convert Images to Tiled TIFF

Converts one or several images (BMP, FlashPIX, GIF, JPEG, PNG, PNM and TIFF) to **Tiled TIFF format**.

 **Note:** Use <Ctrl> or <Shift> for selecting more than one image in the **Open** dialog box.

6) Exit

Closes all the opened projects and exits JMicroVision.


Options Menu (Main)



1) Preferences

1.1 Look & Feel

This option allows you to choose a specific design for the JMicroVision interface. The number of look and feel depends on which platform and Java Virtual Machine JMicroVision is running on.


 **Note:** Restart JMicroVision to refresh the interface correctly. GTK, GTK+ and Mac OS X look and feel are not supported.

1.2 Memory

This page only shows information about the memory used by JMicroVision. For changing the memory settings, see [Memory Management](#).

1.3 Localization

The **Locale** selection changes the number layout and the date layout in JMicroVision (see Java documentation for [Locale ID](#)).

 **Note:** Use the number format of the selected **Locale** when entering or editing a number in JMicroVision. For instance, the decimal point is a comma in some countries.

The date layout depends of the **Locale** selection and the **Date** format selection.

Windows Menu (Main)



1) Projects

Maximizes the image window of the project selected in the submenu.

2) Cascade

Stacks the image windows and aligns them from the upper left to the lower right corner within the workspace.

3) Tile Vertically

Arranges the image windows (except iconified windows) so that they are all the same size. The workspace is divided vertically.

4) Tile Horizontally

Arranges the image windows (except iconified windows) so that they are all the same size. The workspace is divided horizontally.

5) Iconify All

Minimizes the image windows at the bottom of the workspace.

6) Close All

Closes all the opened projects and their child windows.

Help Menu (Main)



1) Contents & Index

Opens [the online help](#) window.

2) About JMicroVision...


Opens the about window which displays the JMicroVision version number, the license and system information.

File Menu (Project)




1) Save Project

Saves the current project file with a ".jmv" extension.

-  **Note:** Before overwriting the older file, the structure of the current project is tested and a message appears if it contains errors. For saving anyway, use the command "Save Project as...".

2) Save Project as...



Saves the current project as a new file with a ".jmv" extension.

-  **Note:** If the project contains errors, this command tries to correct them.

3) Export Project as...


3.1 Image (PNG, TIFF)

Saves in a PNG or a TIFF file a snapshot of the project rendering that includes all the vector drawings, the displayed image and the scale. The resolution of the saved image corresponds to the actual pixel size of the main image of the project.

-  **Note:** The **Display** settings allow showing or hiding the different vector drawings, the image and the scale. Only the visible objects will be included in the saved image.
-  **Note:** Only a part of the image can be saved when specific areas have been created with the **Area Editor**. A popup dialog lets you select one of the areas or the whole image.

3.2 Vector Format (SVG)


Saves the project in a SVG file.

-  **Note:** The **Display** settings allow showing or hiding the different vector drawings, the image and the scale. Only the visible objects will be included in the SVG file.

4) Import Image...


4.1 Matching Coordinates

Adds an image that represents the same scene and has coordinates which match to those of the main image of the project.

-  **Note:** The added images must have the same size as the main image of the project.

4.2 Different Coordinates

Opens the **Image Rectification** module.

 **Note:** The added images can have different size and different coordinates than the main image of the project.

5) Image List

Opens the **Image List Editor** window.

6) Close

Closes the project and its child windows.

Project Menu (Project)




1) Lens

Shows or hides the **Lens** on the image.

2) Project Preferences

Opens the **Project Preferences** window, which allows customizing the **Tools**.

 **Note:** The preferences are automatically saved in the **JMicroVision Project** file.

3) Data Viewer

Opens the **Data Viewer** window, which displays data tables produced by the different **Tools**.

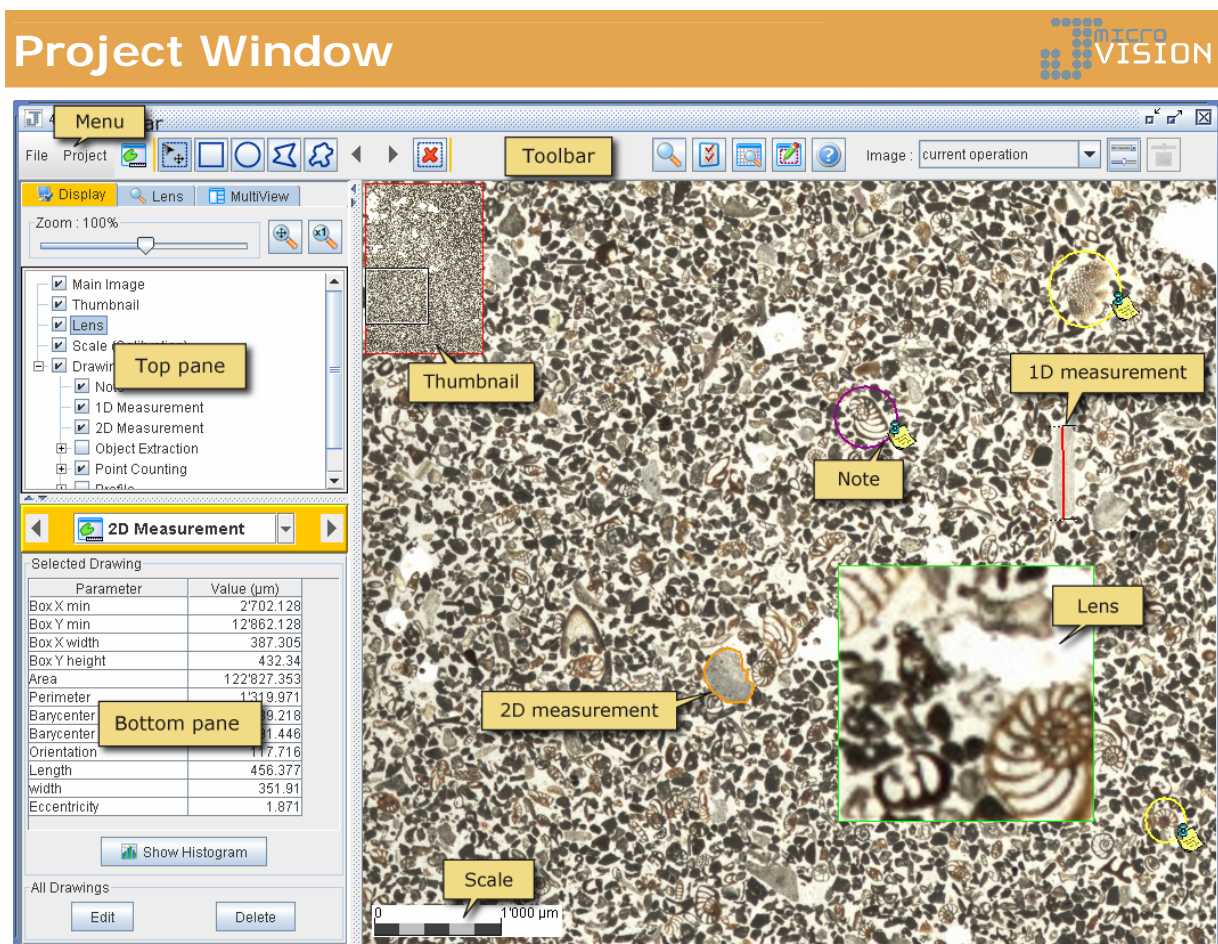
4) Area Editor

Opens the **Area Editor** window, which allows defining areas that can be used later in some **Tools** to process only a part of the image.

5) Image Properties

Opens the Image Properties window, which provides basic information about the image displayed in the main window.

User Interface



1) Menu

- File Menu
- Project Menu
- Toolbar

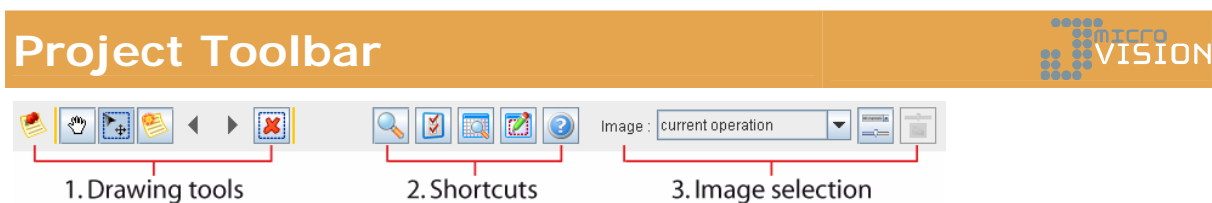
2) Top pane

- **Display** tab: Allows displaying the image, the thumbnail, the lens, the scale and all the vector graphics.
- **Lens** tab: Shows the settings of the lens.
- **Multiview** tab: Allows creating one to eight views that can contain any image of the project.

3) Bottom pane (Tools)

Select one of the following tools in the yellow header:

- **General Description**: Allows managing your standard card.
- **Spatial Calibration**: Allows calibrating the image.
- **Note**: Allows encircling something in the image and setting a name and a comment.
- **1D measurement**: Allows collecting data in one dimension.
- **2D measurement**: Allows collecting data in two dimensions.
- **Object Extraction**: Allows extracting objects from the image with a segmentation method and classifying the extracted objects.
- **Point Counting**: Allows performing a digital point counting on the image.
- **Background**: Allows extracting the background of the image.
- **Profile**: Allows generating profiles (color, grain size, background or extracted objects).
- **Image Factory**: Allows various image processing operations.



1) Drawing tools

Only items related to the selected **Tool** are visible. In the toolbar above, the selected tool is **Note**.

2) Shortcuts


The **Lens** button is a toggle button. The other buttons open the **Preferences** window, the **Data Viewer**, the **Area Editor** and the **online help**.

3) Image selection

3.1 Combo box selection

This option allows to display in the main window the image selected in the list, which includes the images loaded in the project or those produced by the **Image Factory** tool.


By default, there are two images: **current operation** and **source**. When **source** is selected, the image from which the project has been created is displayed. **Current operation** is the default mode that shows an image and the operation applied to it (e.g. when adjusting the **threshold** values).

 **Note:** The preview of an image operation is displayed through the **current operation** image, which is selected automatically.

3.2 Slider selection



The toggle button (the first one) allows switching between the combo box and the slider selection. Each tick below the slider corresponds to one image.

 **Note:** When moving the slider, the name of the current image is displayed in the status bar.

The second button opens the Slide Show Selection dialog, which allows selecting the images (by clicking on its check box) to include in the slider and changing the image position (use drag and drop, first select one or several images and then drag them). When the **Show Transition** button is checked, it enables to merge two images by positioning the slider between two ticks (the percentage of each image is displayed in the status bar).

Image List Editor



The **Image List Editor** allows to manage all the images of the project. The main image of the project and the **imported images** are linked directly to a file and followed by the annotation ": {File}" (except the rectified images followed by the annotation ": {Rectified}", see **Image Rectification**). The other images are image operations produced by the **Image Factory** tool.

 **Note:** To open this dialog: In the **File Menu** of the project window, select **Image List**.


Edit Path: Edit the path of an image linked directly to a file. Changing the path of an image will have an effect on all the image operations that have this image as a source.

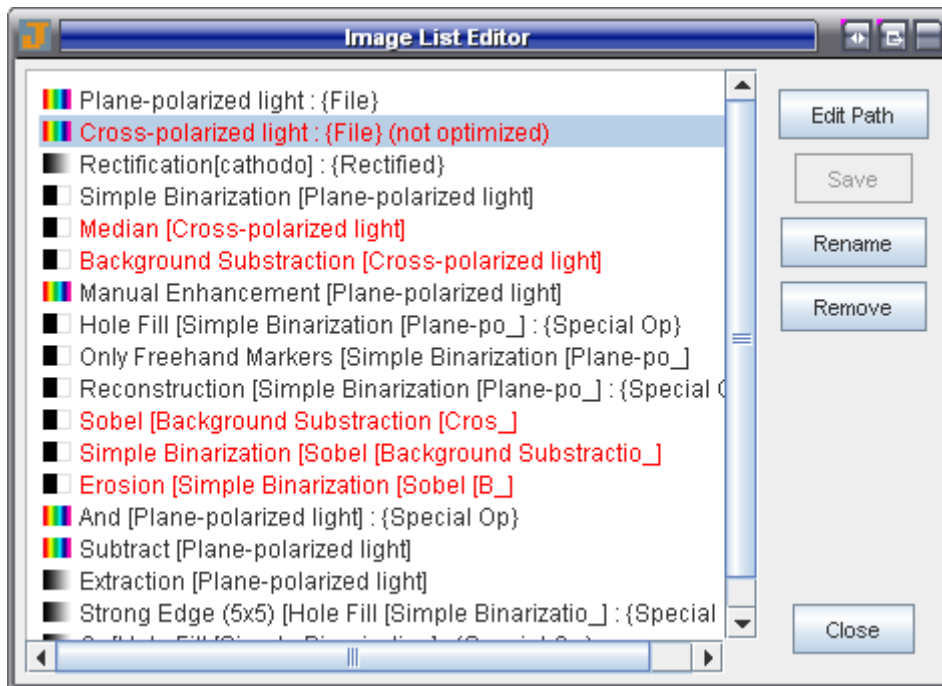
 **Note:** The path of the main image (first on the list) cannot be changed (to change the main image, see **New Project - From another Project**). For the others, the new path must be an image with the same size as the main image of the project, except for the rectified images.

Save: Save the selected image in **Tiled TIFF**, **TIFF** or **PNG** (see **file formats**). To save a part of the image, create an area (see **Area Editor**), then when saving a dialog appear that let you choose the desired area.

Rename: Rename the selected image.

Remove: Remove the selected image from the project.

 **Note:** Removing an image followed by the annotation ": {File}", ": {Rectified}" or ": {Special Op}" will also remove all the image operations linked to this file.




- ✎ **Note:** The images in red are invalid and they do not appear in the project, except in this editor. It can occur when an image file has been moved, renamed or deleted outside JMicroVision.
- ✎ **Note:** The annotation "**(not optimized)**" means that the file is not a [Tiled Tiff format](#).
- ✎ **Note:** For the icon on the left, see [Image Types](#).

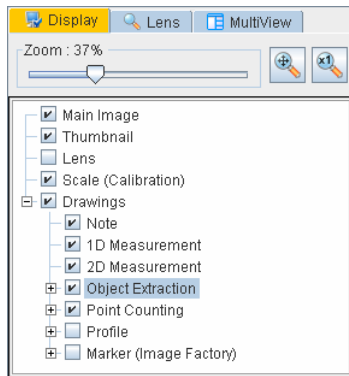
►► **See also:**



[Image Factory](#)

Display




The display options allow to customize the rendering of the images and the vector drawings. The entire image (thumbnail) is visible at the top left corner of the main image. Dragging the rectangle over the thumbnail allows navigating through the image. Another way of moving the image is to select the hand  in the toolbar or to keep pressed the space bar for dragging the image.



The slider enables to set a custom zoom factor for the main image. When the "resize to best fit" button  is pressed, the whole image is adjusted inside the window. The "actual pixel size" button  displays the image at 100%. For getting an optimal speed of vector graphics and image display, it is recommended to set the zoom at 100%.

The check boxes allow displaying or hiding the image, the thumbnail, the **lens**, the **scale** and all the vector drawings produced by the different **Tools**.

 **Note:** Only the drawings which are checked are loaded in the **Data Viewer** and involved in the **object classification**.

 **Note:** Using the mouse wheel on the main image changes the zoom factor.

1) Display preferences

For best performance, select the nearest neighbor interpolation, set the drawing opacity to 100% and do not check the antialiasing button. In addition, the "fill drawing" option in Object Extraction must be unchecked (only the outlines are displayed) to improve the rendering performance.

Background color: Set the background color of the main window.

Interpolation is a process of generating the value of a pixel based on its neighbors to increase or to decrease the number of pixels in a digital image.


- The **Nearest neighbor** interpolation is the simplest method. Basically, it extends the pixel value.
- The **Bilinear** Interpolation determines the value of a new pixel based on a weighted average of the 4 pixels in the nearest neighborhood of the pixel. The averaging has an anti-aliasing effect and therefore produces relatively smooth edges.
- The **Bicubic** and **bicubic 2** interpolations are more sophisticated (slower rendering) and produce smoother edges than the bilinear interpolation. A new pixel is the result of a bicubic function using 16 pixels in the nearest neighborhood of the pixel.

Handle size (pixel): Set the size of the handles of the selected drawings.

Move the image if drawing hitting a border: Moves automatically the image when drawing or moving vector graphics near a border.


Antialiasing for drawings and text: Draws vector drawings and text (only on the image) with antialiasing.

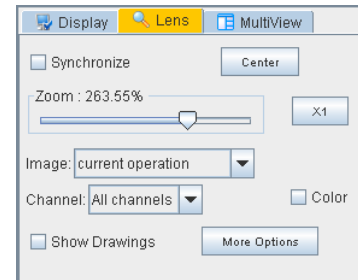
Drawings Opacity: Select the percentage of the drawings opacity (when the value is less than 100, the rendering can be very slow. To speed up the transparency rendering, use the OpenGL acceleration, see [Using OpenGL for Java 2D](#)).


 **Note:** To show the preferences, open the **Project Preferences** window , select **Display** or the submenu **Thumbnail**.


Lens



The lens  can zoom in and out in the range of 8.33% to 1200% by moving the slider, by clicking on the **X1** (100%) button or by selecting a value in the context menu of the lens. It can be used in many situations, for instance to magnify an area, to compare images or to display an area without the vector drawings.



 **Note:** Double clicking on the lens adjusts the zoom factor of the lens to the one of the main image.

 **Note:** Using the mouse wheel on the lens changes the zoom factor.

Synchronize: When this option is activated, the zoom factor of the lens is permanently adjusted to the zoom factor of the main image. Therefore, the user interface for changing the zoom factor of the lens is locked.

Center: Moves the lens to the center of the main image.

Image: Select the image to display in the lens.

Channel: Can display all channels or any channel of a color image (see **Image Types**).



Color: When this box is checked, the **Red**, **Green**, **Blue** or **Hue** channel is colored, otherwise these channels are displayed in grayscale.

Show Drawings: Displays in the lens the visible vector drawings (see **Display**).

1) Lens preferences


Constrain Proportions: If you change the lens width when this box is selected, the height will automatically adjust according to the ratio between the width and the height.

Interpolation is a process of generating a value of a pixel based on its neighbors to increase or to decrease the number of pixels in a digital image (see **Display preferences** for more details about the interpolation types).

 **Note:** To display the preferences, press the **More Options** button or open the **Project Preferences** window  and select **Lens**.

Multiview



Multiview  allows displaying simultaneously several image views with a particular zoom factor or a factor synchronized to the main image.

New Multiview: Select a number of views (one to eight).

Size: Set the width of the views.

Vertical/Horizontal: When this box is checked, the views are aligned vertically on the right of the main window. Otherwise, they are aligned horizontally at the bottom.

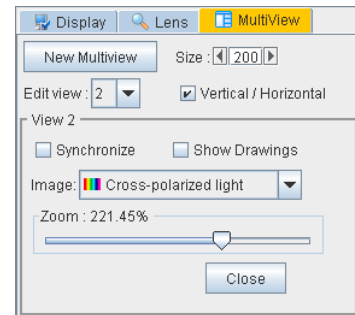
Edit view: Select one view or select "All" for editing all the views at the same time.

Synchronize: When this option is activated, the zoom factor of the view is permanently adjusted to the zoom factor of the main image. Therefore, the user interface for changing the zoom factor is locked.

Show Drawings: Displays on the image the visible vector drawings (see **Display**).

Image: Select the image to be displayed.

Zoom: Move the slider to adjust the zoom factor.



1) Multiview preferences

Choose the colors of the pointer.

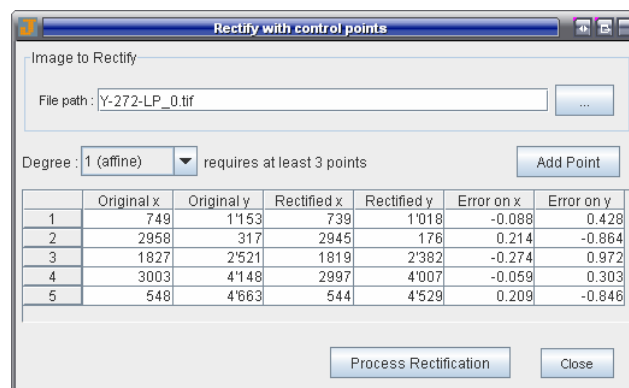
Note: To display the preferences, open the **Project Preferences** window and select **Multiview**.

Image Rectification



The image rectification module using control points for geometric corrections helps to match two images that represent the same scene but have a different size, orientation or deformation. In order to rectify the second image, it is necessary to set corresponding points on both images. Each pair of points must match as well as possible the same place on both images.

Note: To open this dialog: In the **File Menu** of the project window, select **Import Image - Different Coordinates**.



File path: Press the browse button on the right and choose an image to rectify.


Add Point: Click on this button, then move the cross cursor on the reference image (at the top) and click on the desired point. A corresponding point is automatically set according to the image size on the image to rectify (at the bottom).


Note: When selecting a row of the table that corresponds to a pair of points located on both images, the two points are selected and displayed in the center of the main window.

A point can be deleted by pressing the delete key when a point or a row in the table is selected.

Degree: It is possible to make affine, quadratic, quartic or quintic transformations (most of the time the affine transformation is the more suitable) and to have an error associated to each point, when there are more points than the minimum required by the transformation type. The error on x and y axis (distance from a point to the predicted point according to the equation of the transformation) can give a clue as to which points should be moved in order to reduce the error.

When there are enough points to compute the rectification, press **Process Rectification** to add the rectified image to the project.

 **Note:** The rectified image added in the project is overwritten when the button **Process Rectification** is pressed several times. To get more than one rectified image in the same project, just close the rectification module and reopen it to rectify an image.

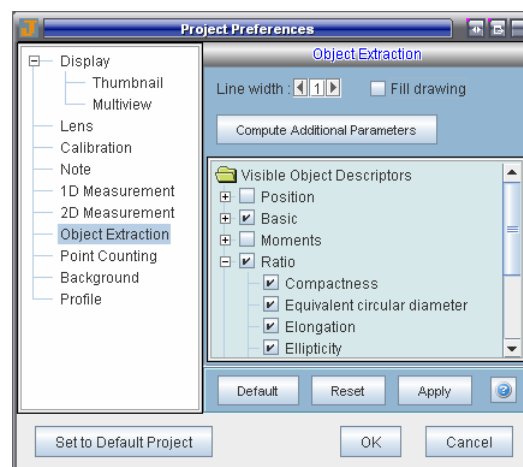
 **Note:** The project windows cannot be closed as long as the image rectification window is open.

Project Preferences



The project and its tools can be customized from the **Project Preferences** window. All the preferences are saved in the project file.

For more details, see preferences of each item: **Display**, **Lens**, **Spatial Calibration**, **Note**, **1D Measurement**, **2D Measurement**, **Object Extraction**, **Point Counting**, **Background** and **Profile**.





1) Set to Default Project


Clicking on **Set to Default Project** sets all the preferences as default preferences for any project created later. The **Default** button will restore the default preferences of each item.

Data Viewer



The **Data Viewer**  can display the data produced by the **Tools** and export them.

 **Note:** Only vector drawings which are visible on the image (see **Display**) are loaded in the Data Viewer. In addition, only the parameters that have been checked in the **preferences** are visible.

 **Note:** For sorting a column, click on the column header. A triangle indicates whether data are sorted ascending or descending.

Data: Select a tool from the list.

Select graphic: When this box is checked and a cell of the table is selected, the corresponding drawing is selected and displayed in the center of the main window.

Mode: Graphics or Classes (see below).

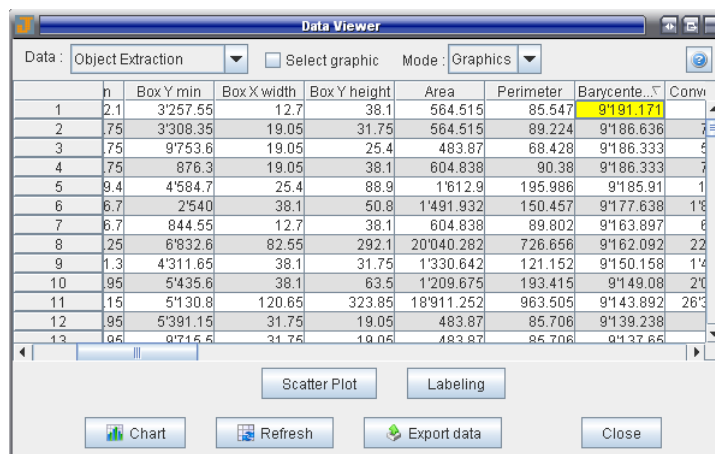
Refresh: Reload the data (e.g. when drawings have been deleted or hidden).

Export Data: **Export data** in a semicolon separator file (.csv), in an ASCII tabulation separator file (.txt) or in an XML version 1.0 file (.xml).

 **Note:** When exporting data, the complete decimal fraction of values is written.

1.1 Graphics Mode

Shows the visible parameters of vector drawings of **Note**, **1D Measurement**, **2D Measurement**, **Object Extraction**, **Point Counting** and **Image Factory** (markers).



	h	Box Y min	Box X width	Box Y height	Area	Perimeter	Barycente...	Conv
1	2.1	3'257.55	12.7	38.1	564.515	85.547	9'181.171	
2	75	3'308.35	19.05	31.75	564.515	89.224	9'186.636	
3	75	9'753.6	19.05	25.4	483.87	68.428	9'186.333	
4	75	876.3	19.05	38.1	604.838	90.38	9'186.333	
5	9.4	4'584.7	25.4	88.9	1'612.9	195.986	9'185.91	1
6	6.7	2'540	38.1	50.8	1'491.932	150.457	9'177.638	18
7	6.7	844.55	12.7	38.1	604.838	89.802	9'163.897	6
8	25	6'832.6	82.55	292.1	20'040.282	726.656	9'162.092	22
9	1.3	4'311.65	38.1	31.75	1'330.642	121.152	9'150.158	14
10	95	5'435.6	38.1	63.5	1'209.675	193.415	9'149.08	20
11	15	5'130.8	120.65	323.85	18'911.252	963.505	9'143.892	263
12	95	5'391.15	31.75	19.05	483.87	85.706	9'139.238	
13	95	9'715.5	31.75	19.05	483.87	85.706	9'137.65	

Chart: Shows the histogram of parameters.

Scatter Plot: Allows comparing values of two parameters.

Labeling: Allows applying a label on drawings (see **Labeling**).

1.2 Classes Mode

Shows the surface percentages and the number of occurrences of the **Object Extraction**, **Point Counting** or **Background** classes.

Class	Whole Image (%)	Whole Image	area 1 (%)	area 1	area 2 (%)
Quartz	32.33%	194	36.25%	91	27.87%
Porosité	17.00%	102	17.13%	43	17.21%
Ciment micritique	12.67%	76	11.95%	30	15.98%
Débris bioclastiques non...	5.33%	32	3.98%	10	5.74%
Ciment microcristallin	4.67%	28	6.77%	17	3.69%
Non défini	3.67%	22	1.20%	3	2.05%
Grains cortiqués	3.17%	19	1.99%	5	2.46%
Débris bryozoaires	2.83%	17	3.59%	9	2.05%
Débris brachiopodes	2.83%	17	2.39%	6	2.87%
Particules micritisées	2.83%	17	0.80%	2	5.74%
Foraminifères	2.67%	16	4.38%	11	1.23%
Coprolithes	2.67%	16	3.19%	8	2.87%
Débris biogènes	2.67%	16	2.39%	6	3.69%

Chart: Shows the pie chart of classes.

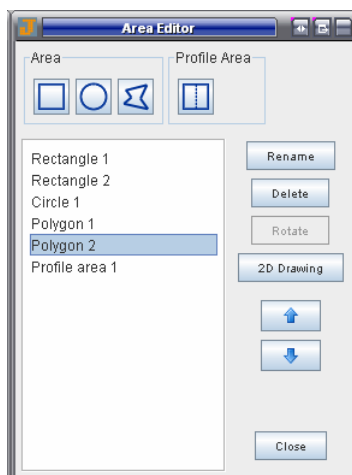
Compute result from Area: Allows computing the amount of classes in a specific area of the image (see **Area Editor**).

Note: Only the whole image area is automatically computed. The other areas must be computed every time data are reloaded.

Area Editor



This Area Editor allows defining specific areas for limiting the processing to a part of the image. The profile areas, which have a rectangular shape crossed by a dotted line that indicates the direction of the profile, are intended only for the **Profile** tool. The other areas, which are either rectangular, elliptic or polygonal, can be used by **Data Viewer**, **Object Extraction**, **Point Counting** or **Image Factory**.



For creating a new area, press one of the shape button and then click and drag the shape on the main image (a cross-cursor is displayed before and during the drawing time).

Note: The selected areas in the list are those visible on the image. For resizing or moving an area, select it and move the matching handle once the cursor has changed.

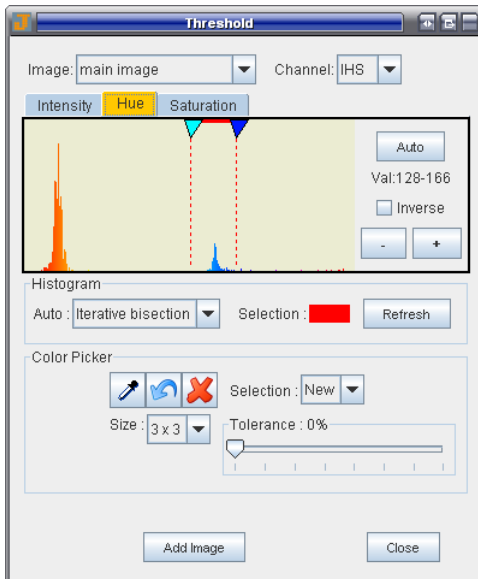
Rotate: Allows to rotate a profile area.

2D Drawing: Creates 2D drawings from the selected areas. Using this option gives information on the area position and the area measurements.

Note: When closing the dialog, all the rectangular areas that have a region outside the image are corrected so that they remain inside the image. For the other area types, a message asks the operator to correct them.

Note: The length and the width of an area must be larger than 4 pixels, otherwise the area is not saved. There is no restriction for the profile areas, the smallest width can be 1 pixel.

Histogram Thresholding



Histogram thresholding is used by the **Object Extraction**, **Background** and **Image Factory** (segmentation) tools to select a range of pixel values that should match to what you want to select in the image.

Thresholding is a technique for converting a grayscale or color image to a binary image based upon two threshold values. The histogram presents the frequency of intensity values of a channel in an image (see **Image Types**). The threshold values (min and max) can be applied by moving the arrows above the histogram or by using an automatic threshold method.

Note: The histogram represents the intensity values of the visible part of the image only. After moving the image, use the **Refresh** button to update the histogram.

Note: The **Image Factory** tool offers some image preprocessing functions (e.g. to reduce noise).

Image: Select one image in the list.

Channel: Select the **RGB** or **IHS** channels for a color image (see **Image Types**)

Inverse: Inverse the selection values.

- +: Shrink or stretch the histogram vertically.


Auto: In the combo box, select an automatic histogram threshold method. Then click on the **Auto** button to find the appropriate max threshold value (min is always equal to 0).


- **Iterative bisection** finds the midpoint in an iterative fashion.
- **Max variance** maximizes the ratio of the between-class variance to the within-class variance.
- **Max entropy** maximizes the entropy.
- **Min error** minimizes the probability of error.
- **Min fuzziness** minimizes the fuzziness.

Refresh: Refresh the histogram according to the visible part of the image.

Selection Color: Change the selection color on the image and between the arrows of the histogram.

Color Picker

This tool offers an alternative method to select a range in the histogram by picking colors in the image. The selection results of the min and max values of the pixels, which are displayed in the dialog at the left of the picker button .

: Click on this button, then move the cross cursor on the image and finally click on the desired area to make a new selection.

: Undo the selection (only the last selection can be restored).

: Clear the current selection.

Selection: When **Add** is selected, the new selection does not replace the previous.

Size: Set the size of the picker.

Tolerance: Increase the min and max picked values.


 **Note:** Adjusting the slider after the selection has no effect.


Image Types



Most of the digital image channels have 256 intensity levels coded on 8 bits per channel. The color image is produced by combining three channels. The grayscale images have only one channel with 256 gray levels per pixel. The binary images are coded on 1 bit and therefore have only 2 levels.

1) Binary image

A binary image has only one channel (**Binary**). Pixel values in binary images are either black or white. The binary images can be created by one of the **segmentation methods**.

 **Note:** The color of the objects (white regions) is referred to as the foreground color. The rest (black) is referred to as the background color.

2) Grayscale image

A grayscale image has only one channel (**Gray levels**). Pixel values in grayscale images are based upon luminance.

The **Real Gray** channel results from the conversion of a color image to a grayscale image (in **lens** options or in **channel** operations). The coefficients used to compute luminance are 0.299 (red), 0.587 (green) and 0.114 (blue). $\text{Pixel} = 0.299 * \text{red} + 0.587 * \text{green} + 0.114 * \text{blue}$.

3) Color image

In **lens** options or in **channel** operations, a color image can be separated into three channels using the **RGB** or **IHS** color space or in one **Real Gray** channel (see above).

3.1 RGB

The RGB model is an additive color model and also one of the most common. It has three primary colors: **red**, **green** and **blue**. These three colors are combined in various proportions to produce all the colors displayed on your image. The ICC (International Color Consortium) profile used by JMicroVision is the sRGB.

3.2 IHS

IHS (or called HSV) is a more intuitive color space than RGB.

- **Intensity** (or luminosity) is the amount of light independently of the color.
- **Hue** describes the visual sensation according to which an area appears to be similar to one of the perceived colors, red, yellow, green and blue, or a combination of two of them.
- **Saturation** is the colorfulness of an area judged in proportion to its brightness. Saturation runs from neutral gray through pastel to saturated colors. A vivid color usually has a high saturation level (not much white).

Coloring Objects



This dialog enables to color all the objects from a look-up table according to the values of the selected descriptor.

Note: To open this window, select the **Object Extraction** tool, then the **Objects** tab and finally click on the **Color** button.

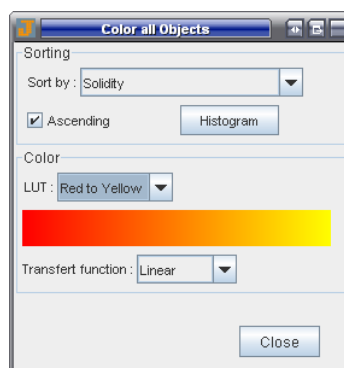
Sort by: Select an **object descriptor**.

Ascending: If checked, the objects with the smallest values are colored by the left side of the look-up table and the biggest by the right side.

Histogram: Open the histogram of the selected descriptor.

LUT (look-up table or palette of colors): Select a LUT.

Transfer function: Select Linear, Logarithmic or Exponential. The transfer function sometimes helps to bring out small differences between objects at the beginning or at the end of the range values.



Note: When closing this window, all the objects retrieve their original color (corresponding to the class which they belongs to).

Labeling



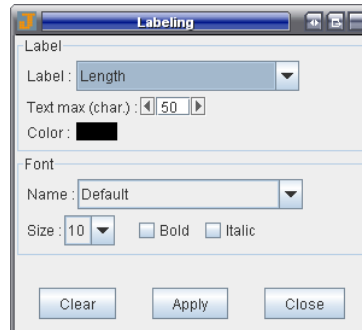
This dialog enables to attach a label on the vector drawings of the selected Tool in the **Data Viewer**.

Label: Select a parameter to display as a label.

Text max. (char.): This is the limit of characters that can be displayed for a text parameter (e.g. Name of **Note**).

Color: Select a font color.

Font: Customize the font size and type.



Note: To open this window: In the **Data Viewer** , click on the **Labeling** button.

Class Editor



The Class Editor allows creating, modifying and merging the classes of **Point Counting**, **Object Extraction** and **Image Factory (markers)**. A class enables to group several objects. The class results can be shown in the **Data Viewer** (see **Class mode**).

Add: Create a new class.

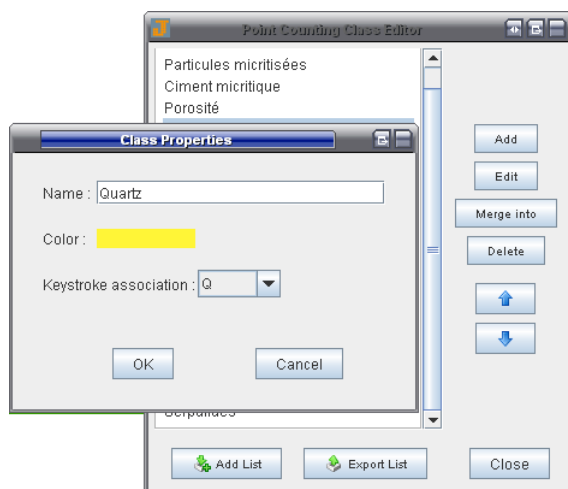
Edit: Open the **Class Properties** window of the selected class.

Merge into: Merge the selected classes into another one.

Delete: Delete the selected classes.

Add list: Load a list of classes (.pcl file) and add it to the current list.

Export list: Save the current list of classes in a ".pcl" file.



Class Properties

Name: Enter the name of the class.

Color: Choose the color of the class.

Keystroke association: Select a key.

Note: This parameter is optional and is used only by the **Point Counting** classes. It enables using keystrokes instead of clicking on the button of classes.

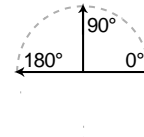
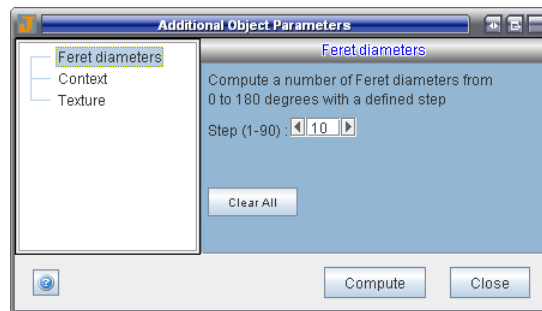
Note: The classes are saved in the **JMicroVision Project file**.

Feret Diameters



Feret diameters are the distances between parallel tangents touching opposite sides of the object. The number of Feret diameters from 0 to 180 degrees is defined by an angle (1-90 degrees). Feret Min is the shortest Feret diameter and Feret Max is the longest Feret diameter.

- Note:** To compute ferret diameters: In the **Project** menu, click on **Preferences**. In the **Project Preferences** window, select **Object Extraction**, click on the button **Compute Additional Parameters** and select **Feret Diameters**.



Select a step from 1 to 90 and press the **Compute** button (all the previous computed Feret diameters are deleted).

Example:

- step = 2 => Feret diameters at 0, 2, 4, 6, ..., 174, 176 and 178 degrees
- step = 10 => Feret diameters at 0, 10, 20, 30, ..., 150, 160 and 170 degrees
- step = 90 => Feret diameters at 0 and 90 degrees

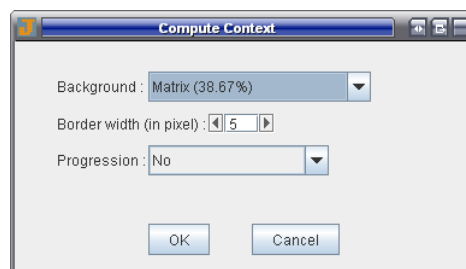
- Note:** To display the Feret diameters in the Data Viewer or in the **Object Extraction** tool (Objects tab): In the **Project Preferences** window, select **Object Extraction** and check **Feret Diameters (Additional)**. The Feret diameters are not saved in the **JMicroVision Project file**.


The **Clear All** button deletes all the Feret diameters of all the objects.

Context



Context is an additional object descriptor which describes the neighboring environment of the object. A **Background** is required to represent the environment.



 **Note:** To compute the context parameter: In the **Project** menu, click on **Preferences**. In the **Project Preferences** window, select **Object Extraction**, click on the **Compute Additional Parameters** button and select **Context**.

In the window **Compute Context**:


- 1) Select a **Background** (if there is no background, make one)
- 2) Choose a border width that represents an external border of the object in pixels.
- 3) Select a progression type (from internal border layer to external border layer)

The context parameter creates an external border for every object and counts the border pixels superimposed with pixels of the selected background. If the progression is:

- No : All the border pixels have the same weight
- Arithmetic ascending : An ascending weight is set from the internal border layer to the external border layer
- Arithmetic descending : An descending weight is set from the internal border layer to the external border layer




In the image on the left (created to represent the context concept), the object is green and the border width is 10 pixels. The internal border layer is red and becomes progressively white to the external border layer. When the progression is arithmetic ascending: the redder pixels get more weight.

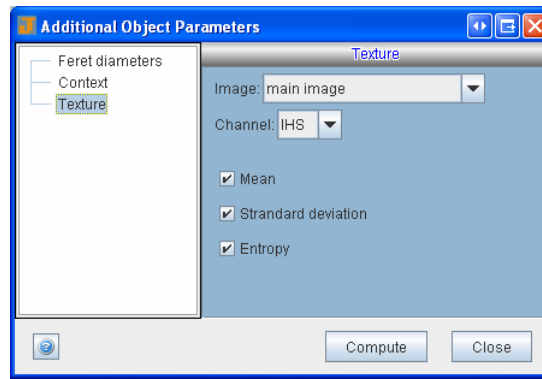
 **Note:** To display the context parameter in the Data Viewer or in the **Object Extraction** tool (Objects tab): In the **Project Preferences** window, select **Object Extraction** and check the **Context (additional)** item. The context parameters are not saved in the **JMicroVision Project file**.

Texture




Mean, **Standard deviation** and **Entropy** are statistical properties of the pixel intensity values. Here, the statistical properties of a channel are computed for each object. The three texture parameters are additional **object descriptors**.

 **Note:** To compute the texture parameters: In the **Project** menu, click on **Preferences**. In the **Project Preferences** window, select **Object Extraction**, click on the button **Compute Additional Parameters** and select **Texture**.





Channel: Select the **RGB** or **IHS** channels for a color image and **Gray Levels** for a grayscale image (see [Image Types](#)).

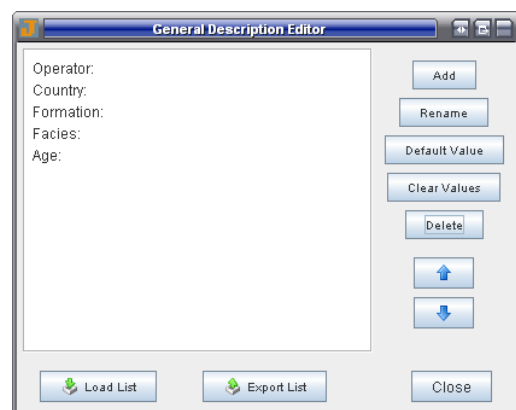
 **Note:** To display the texture parameters in the Data Viewer or in the [Object Extraction](#) tool (Objects tab): In the [Project Preferences](#) window, select **Object Extraction** and check **Texture (Additional)**. The texture parameters are not saved in the [JMicroVision Project file](#).

Tools

General Description



The **General Description**  tool allows creating a specific description card by using the editor . Each item has a property (name) and a value. When creating a list, a default value can be entered. Otherwise the values can be edited directly in the **General Description** table. A list of items can be loaded or exported.



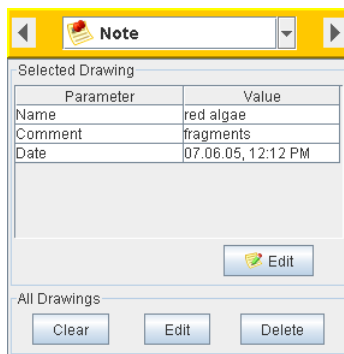
Note



This tool allows attaching notes to relevant objects or properties in the image by encircling them and setting a name and a comment.



In the toolbar above, the first toggle button allows selecting or moving drawings, the second one allows creating new notes. The two arrow buttons allow going from a note to another. The last button will delete the selected drawings.



The date corresponds to the instant when the note was created or updated (see [date layout preferences](#)). For updating a note, double click on it, or select it and press the **Edit** button. The **Clear** button deletes all the names and the comments from the notes.

1) Note preferences

Only the checked parameters will appear in the **Selected Drawing** panel and in the **Data Viewer**. More details about the position parameters are described in the **Object Descriptors**.

- Note:** To display the preferences, open the **Project Preferences** window and select **Note**.

Spatial Calibration



The spatial calibration tool can convert automatically the pixel values to any other unit of measurement by multiplying the measured values by a conversion factor. Once the spatial calibration is defined, all the position and measurement values in JMicroVision are displayed according to the conversion factor.

When the check box called **Scale** is activated in the tree (see **Display**), a scale is displayed at the left bottom on the image. The scale is automatically adjusted when the zoom factor of the main image is modified.


When there is no calibration, all the **Tools** use the pixel as unit of measurement. Otherwise, the **Output unit** selected in the combo box (European or Anglo-Saxon unit) defines the unit of measurement.

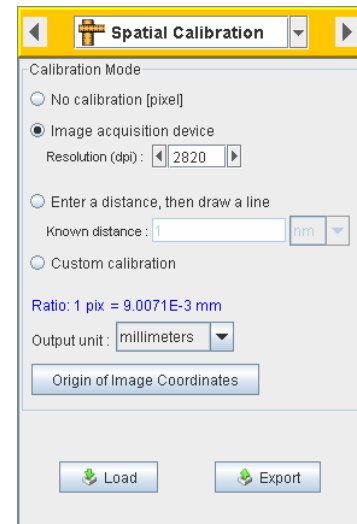
- Note:** The spatial calibration becomes invalid if the image has a vertical resolution different from the horizontal resolution, or if the image has been distorted or rescaled with a different horizontal/vertical ratio.
- Note:** Sometimes you may need to refresh the measurement values to apply the new spatial calibration or when the output unit is modified, by pressing the Refresh button (e.g. **Data Viewer**) or by reselecting the drawing.

In the calibration tool, there are four modes:


- No calibration
- Image acquisition device
- Known distance
- Custom calibration

The **Load** and **Export** buttons allow loading and saving a spatial calibration, so that it can be reused in another project.


 **Note:** The conversion factor uses more decimals than the ones displayed in the interface (in blue) to convert measurements into real values.



The **Shift Image Origin** button opens a dialog which allows to shift the origin of the image. The first panel "Default Image Coordinates" defines a position without any shift. The second panel "New Coordinates" redefines the position entered in the first panel.

 **Note:** The origin can be located by default at the top or at the bottom left corner of the image (depends the Y axis direction, see [Spatial Calibration preferences](#)).

To shift the origin, choose a position which has a known reference from the combo box or enter X and Y values in the first panel (the position can be located in the image by pressing the **Show Position** button). Then, in the second panel, enter X and Y values either in pixels or with a specific unit. When the **OK** button is pressed, the origin of the default image coordinates is shifted. The origin shift is the X and Y distance of the positions entered in the first and the second panel.


 **Note:** If the new coordinates are defined with a specific unit (not pixels), the final X and Y values will be round in pixels (discrete values).


1.1 No calibration

This is the default mode. The conversion factor is set to 1, so the program shows the measurements in pixels without any conversion.

1.2 Image acquisition device

This mode can be used for images acquired by a device like a scanner, by entering the acquisition resolution in dpi.


 **Note:** Some image acquisition devices write the calibration value in the header of the image file. When the image is opened in JMicroVision, the calibration is set automatically (works only with TIFF format).

 **Note:** The calibration is biased when the image has been rescaled or if there is an optical system between the device and the specimen (e.g. that magnifies the specimen).

1.3 Known distance

Some measurement devices may include a scale in the image, thus providing a reference point for calibrating. Images with an object that has a known distance can also be used.

This mode allows setting a conversion value by drawing a line of known distance on an image. Follow these steps:

- Enter a known distance
- Set the unit of the known distance
- Draw a line with the line in the toolbar  (afterwards the line can be moved or rescaled)


1.4 Custom calibration



This mode allows entering a length that corresponds to the length of one pixel.

2) Spatial Calibration preferences

Max length of the scale: defines the maximal length (in pixels) of the scale displayed on the main image. The scale length changes when the zoom factor value or the output unit is changed.

Checking the box **Keep all the position values in pixels** will allow to maintain the position parameters in pixels (see [position object descriptors](#)).

Vertical axis (Y) direction: defines the direction of the Y axis. The default direction is down. In this case, the origin is located at the top left corner of the image . When the direction is up, the origin is at the bottom left corner.

-  **Note:** To display the preferences, open the **Project Preferences** window , select **Calibration**.

1D Measurement



This tool allows creating lines to measure an element in the image or to build a profile of the pixel intensities of the current image. If there is no **spatial calibration**, the segment length value is displayed in pixels.



In the toolbar above, the first toggle button (the one after the hand) allows selecting or moving drawings, the three next ones allows creating new lines: segment, polyline and freehand line. The two arrow buttons allow going from one segment to another. The last button will delete the selected drawings.


Selected Drawing	
Parameter	Value (mm)
First point X	4.693
First point Y	6.773
Last point X	6.683
Last point Y	4.873
Line length	2.994
Orientation	48.241
Orientation significance	0.951
Azimuth	41.759
Azimuth significance	0.919
Barycenter x	5.683
Barycenter y	5.889
Color (RGB)	255:0:0


Show Profile

Selected Drawing

Edit Delete

The **Show Profile** button opens the profile of the pixel intensities of the current image. The **min** and **max** values (yellow horizontal lines on the plot) limit the range of the pixel intensities shown in the **Statistics** and **Data** tables. In the **Statistics** table, the **Threshold ratio** gives the ratio between all the line pixels and the number of the pixels which have intensities in the limited range. The selected table can be saved in **CSV format**.



 **Note:** When moving or resizing a line, or when the image is changed, the profile is automatically actualized.

 **Note:** When moving along the profile plot, a pointer shows the corresponding position on the image and **pixel information** appears in the status bar.

1) 1D Measurement Descriptors

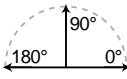
1.1 Position:

- **First point X:** X coordinate position (image) of the first point of the line.
- **First point Y:** Y coordinate position (image) of the first point of the line.
- **Last point X:** X coordinate position (image) of the last point of the line.
- **Last point Y:** Y coordinate position (image) of the last point of the line.
- **Barycenter x:** X coordinate position (image) of the center of gravity of the line.
- **Barycenter y:** Y coordinate position (image) of the center of gravity of the line.

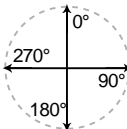
 **Note:** The image position (0,0) is located by default in the upper left corner . However, the origin and the direction of the y axis can be modified in the **Spatial Calibration** tool.

1.2 Basic:

- **Line length:** Length of the line.
- **Orientation:** Angle between the horizontal axis and the line (0-180 degrees, counterclockwise).





- **Orientation significance:** Define the significance (0 to 1) of the orientation according to the line shape. A straight line has the value 1. Conversely, a tortuous line will have a lower value.
- **Azimuth:** Horizontal angular distance from the northern point (at the top). The measurement is made clockwise through 360° by placing the center of the circle on the first point of the line.



- **Azimuth significance:** The ratio between the line length and the Euclidian distance between the first and the last point defines the significance (0 to 1) of the azimuth. A straight line has the value 1.
- **Color (RGB):** Color of the line, defined by the red, green and blue components.

2) 1D Measurement preferences

Only the checked parameters will appear in the **Selected Drawing** panel and in the **Data Viewer**.

-  **Note:** To display the preferences, open the **Project Preferences** window  and select **1D measurement**.

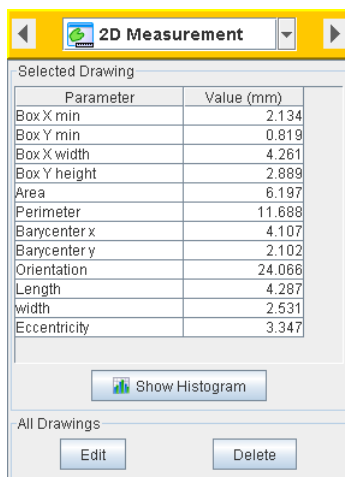
2D Measurement



This tool allows creating shapes to obtain measurements of them. If there is no **spatial calibration**, the parameter values are displayed in pixels (except the orientation, which is expressed in degrees, and the eccentricity, which has no unit). For getting more 2D parameters, the 2D drawings can be **converted into 2D objects**.




In the toolbar above, the first toggle button allows selecting or moving drawings, the other ones allows creating 2D shapes. The two arrow buttons allow going from one shape to another. The last button will delete the selected drawings.





The **Show Histogram** button opens the histogram of the image part, delimited by the selected 2D drawing.

The **min** and **max** values limit the range of pixel intensities, except for the **All data** table that shows all intensities for all channels. In the **Statistics** table, the **Threshold ratio** gives the ratio between all the shape pixels and the number of pixels which have intensities in the limited range. The selected table can be saved in **CSV format**.

-  **Note:** When moving or resizing a 2D drawing, or when the image is changed, the histogram is automatically actualized.

1) 2D Measurement preferences

Only the checked parameters will appear in the **Selected Drawing** panel and in the **Data Viewer**. More details about the parameters are described in the **Object Descriptors**.

-  **Note:** To display the preferences, open the **Project Preferences** window  and select **2D measurement**.

Object Extraction



In JMicroVision components can be extracted as objects (e.g. particles) with the **Object Extraction** tool or as backgrounds (e.g. matrix) with the **Background** tool. The main differences are that the **Object Extraction** tool transforms objects into vector drawings and computes more than forty **descriptors** on each object. However, it is recommended to use the **Background** tool when only the entire surface of a component needs to be calculated and it is not necessary to partition the background in multiple regions.

The **Image Factory** tool offers some image preprocessing functions and different segmentation methods for more advanced use (see tutorials on the website).

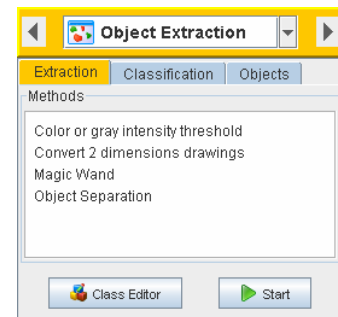
- Note:** **Point counting** is an alternative method to quantify image components, especially when the image is too complex to use the **Object Extraction** or **Background** tools.

1) Extraction

The methods in the list enable creating objects (vector graphics) in different ways. Once created, they can be classified (see **Classification**) and their descriptors can be displayed in the **Data Viewer**.

Start: Open the dialog of the selected method.

Class Editor: Make your own list of classes.



1.1 Color or gray intensity threshold

The segmentation by threshold is the method commonly used to extract objects. The user selects a range of values in a histogram. This range must match as well as possible the pixel intensities of the objects to extract (see **Histogram Thresholding**). In addition, other parameters can be applied:

- **Area:** Define an area for the extraction (see **Area Editor**). The **Whole Image** is the default value.
- **Surface filter:** Remove the objects smaller or bigger than the surface range (Min and Max). It is not possible to define a surface less than 10 pixels because it is meaningless to compute shape descriptors with very few pixels.
- **Border filter:** Remove the objects touching borders.
- **Fill object:** Remove the holes inside the objects.
- **Threshold:** see **Histogram Thresholding**.
- **Assign a class:** Assign a class to the objects (see **Class Editor**).

When the **Extraction config** item is selected, a summary of all the parameters values is displayed. This configuration can be saved with the **Save as Template** button. Afterwards, the template will appear in the combo box (even after closing JMicroVision) and when it is selected, it reapplies the saved configuration.

- Note:** **Templates** can be used by the **Object Extraction** or the **Background** tool, but they can only be created or modified by the **Color or gray intensity threshold** dialog.

1.2 Convert 2 dimensions drawings


This method allows converting **2D drawings** into objects. When the segmentation methods are not able to extract objects correctly, you can manually create an outline around objects with **2D Measurement** tool and then convert them to obtain more descriptors.

1.3 Magic Wand

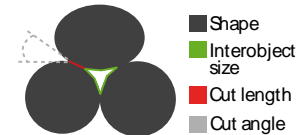
See **Magic Wand**.

1.4 Object Separation

This experimental module is similar to **Color or gray intensity threshold**, but has extra functions to separate connected objects.


 **Note:** The separation process requires to handle the whole image in memory, thus it can be applied on very high-definition images only if the computer has enough memory and the setting of JMicroVision has been adapted (see **Memory Management**).

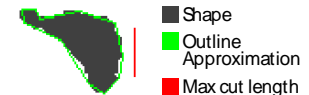
Separation: Choose between **Variable** (undefined shapes) or **Custom Shape**, if the objects to separate have common shape characteristics. The shape variance is defined by a set of 2D Measurement drawings or a class of objects, which are specified when selecting **Custom Shape**.




- **Max cut length (pixels):** Is the max segment length for separating objects.
- **Max cut angle (degrees):** Is the max angle (between two connected objects) for separating objects.
- **Min interobject size (pixels):** Is the smallest size of holes, which are between objects, not inside objects. The default value is 10 pixels. The holes with a smaller size than the entered value are not considered as a possible position for a separation. Change this option only when the holes between objects are bigger than the holes inside objects.
- **Outline approximation (pixels):** Represents the approximation value of the shape outline. Increasing the value will reduce the accuracy of the shape (only when computing, not in the final object shape), but will accelerate the processing of the separation.

Shape Size: Define the **Min size** and the **Max size** of the objects.


 **Note:** When the **Variable Shape** is chosen in **Separation**, the displayed shape is an ellipse that actually represents undefined shapes. For **Custom Shape**, the displayed shape is a vector drawing, which represents the mean of shape parameters of the drawings.



Shape Tolerance: Define the shape and the eccentricity (elongation) tolerance of the objects to separate.

 **Note:** For **Variable Shape**, the sliders are locked to 100% (because the shape is not defined). For **Custom Shape**, the sliders initial values are proportional to the shape and eccentricity variance of the set of vector drawings.

The other commands are identical to **Color or gray intensity threshold**.

 **Note: Templates** can be saved and reapplied later (in **Separation config**).

2) Classification

Allows classifying the objects, see **Classification**.

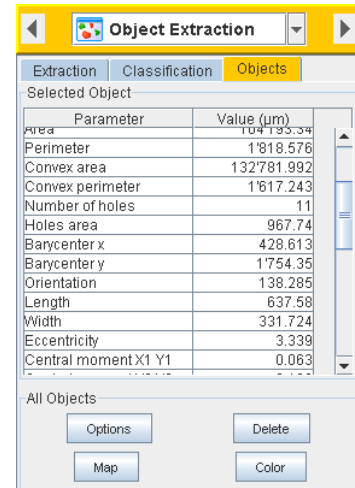
3) Objects

Color: Color all the objects according to a parameter, see **Coloring Objects**.

Map: Color points (points represent the center of gravity of the objects) according to a parameter.

Modify: Modify the object shape by one of these operations:

- **Smooth (mean):** Smooth the contour line of the shape and the holes by a mean filter. The filter size is $2 \cdot \text{Disc radius} + 1$ (in pixels).
- **Smooth (Gaussian):** Smooth the contour line of the shape and the holes by a Gaussian filter. The filter size is $2 \cdot \text{Disc radius} + 1$ (in pixels). A Gaussian filter preserves more the contour shape than the mean filter.
- **Polygonal approximation:** Transform the outline of the shape by a polyline. If the shape contains holes, they are removed. The **Disc radius** represents the approximation value of the shape outline. Increasing this value will reduce the accuracy of the shape.
- **Convex hull:** Transform the outline of the shape by a convex outline and if the shape contains holes, they are filled.





4) Object Extraction preferences

Compute Additional Parameters: see **Object Descriptors**.

Fill drawing: When the button is unchecked, only the outline of objects is displayed.

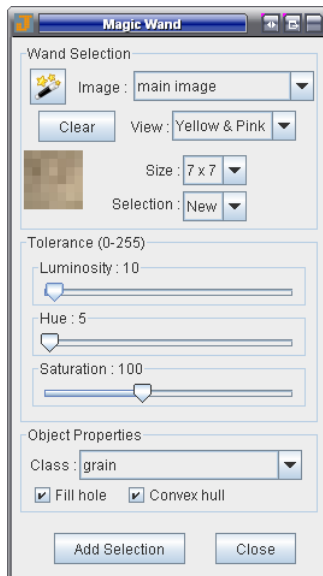
Only the checked parameters will appear in the **Selected Object** panel and in the **Data Viewer**.

 **Note:** To display the preferences, open the **Project Preferences** window , select **Object Extraction**.

►► **See also:**


[Object Descriptors](#), [Feret Diameters](#), [Context](#), [Texture](#)

Magic Wand



The Magic Wand tool is designed to select an object based on its color similarity. It starts selecting when you click at a spot in the image, and expands the selection to the contiguous pixels whose color is similar to that of the starting pixels. You can control the threshold of similarity by adjusting the tolerance values.

The **Add Selection** button converts the current selection into an **Object Extraction** drawing.

 **Note:** The growing process requires to handle the whole image in memory, thus it can be applied on very high-definition images only if the computer has enough memory and the setting of JMicroVision has been adapted, see [Memory Management](#).

1.1 Wand Selection


: Click on this button, move the cross cursor on the image and click on the desired starting area.

Image: Select an image to apply the magic wand.

Clear: Clear the current selection.


View: Choose the selection rendering.

Size: Set the size of the initial area.

Selection: The **New** mode applies a new selection each time. The **Add** mode adds the new selection to the previous.

1.2 Tolerance

Increase the threshold of color similarity. For color images, the tolerance is based on IHS values. For gray scale and binary images, the tolerance is based on the intensity values (see [Image Types](#)).

 **Note:** It is possible to adjust the tolerance values before or after the selection.

1.3 Object Properties

Class: Assign a class to the current object (see [Class Editor](#)).

Fill hole: Remove the holes inside the object.

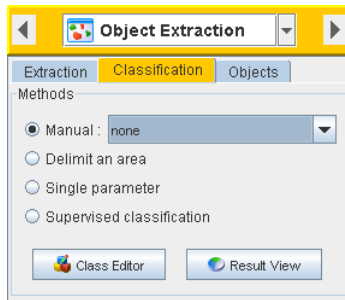
Convex hull: Transform the outline into a convex shape and remove the holes inside the object.

Classification



Four methods can be used to classify the objects created by the **Object Extraction** tool. Furthermore, the easiest way to change the class of one object is to double-click on it or to select **Properties** in the contextual menu to display a dialog which permits to change the class.

Note: Only the checked classes, which are visible on the image (see **Display**), are used by the classification methods. If you do not want to involve a specific class during a classification, just uncheck it.



Class Editor: Make your own list of classes.

Result View: Open the Data Viewer directly in the **Classes Mode** of the Object Extraction tool.

1) Manual

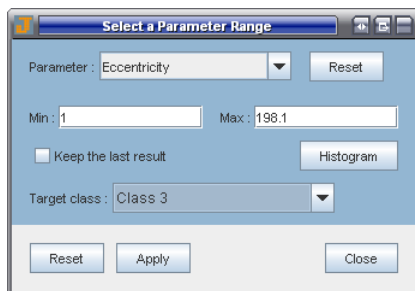
Select a class in the combo box and then click on a object to assign the selected class.

2) Delimit an area

Draw one or more polygonal areas and then click on the **Apply** button to assign the selected class to all the objects that are inside the polygons or touching them.

Note: Press the selection toggle button in the toolbar to stop drawing polygons.

3) Single parameter



Select a parameter, define a range by entering the min and max values, and then press the **Apply** button to assign the selected class to all objects included in this range.

Note: Displaying the values distribution of the **Histogram** may help to choose the range.

Parameter: Select one of the **object descriptors**. The **Min** and **Max** text fields values are the extrema of the selected parameter.

Reset (on top): Set again the extrema (Min and Max) of the selected parameter.

Keep the last result: When this box is unchecked, the previous classification is canceled when a new classification is applied.

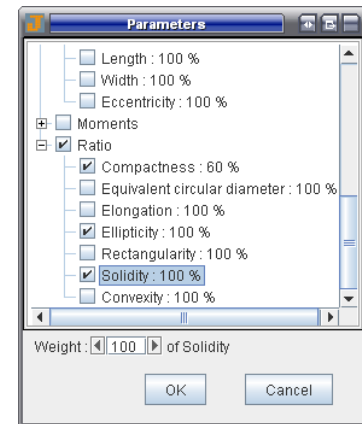
Histogram: Show the histogram of the selected parameter.

4) Supervised classification

The supervised classification uses the nearest neighbors classifier. It involves a set of training objects. New samples are classified basically by calculating the smallest Euclidian distances (parameters space) to the nearest training objects (the most represented class is chosen).

4.1 Performing a classification


- 1) Create the training set. Firstly, add the desired classes (do not use existing classes that contain objects) with the **Class Editor**. Then, assign for each class some training objects by using the **Manual** classification (see above).
- 2) Select the relevant parameters for the classification in the **New Learning Session** tab.
- 3) Click on **Apply** and then check the class boxes that include objects to classify. The other class boxes are the training classes defined in the first point.

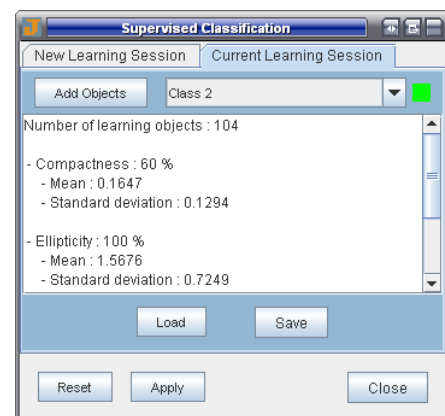


Weight: Set a weight (1 to 100%) to the parameters involved in the classification.

4.2 Load and Save Learning

Once the objects have been classified (see **Performing a classification**), the statistics of parameters of each class are displayed in the **Current Learning Session** tab. Learning can be saved, and then reloaded and applied to objects in another project.

-  **Note:** Learning containing additional parameters (see **Object Descriptors**) cannot be saved (does not appear in the **Current Learning Session** tab), because these descriptors have variable parameters. Involving them could produce erroneous results.




Add objects: Add new training objects to the selected class.

Object Descriptors





All the descriptors are automatically computed when the objects are extracted, except the additional descriptors: **Feret Diameters**, **Context** and **Texture**.

-  **Note:** The additional descriptors values are not saved in the **JMicoVision Project file**.

1) Position:

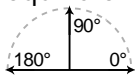
- **First point X:** X coordinate position (image) of the upper left pixel of the object.
- **First point Y:** Y coordinate position (image) of the upper left pixel of the object.

- **Box X min:** X coordinate position (image) of the upper left point of the bounding box*.
 - **Box Y min:** Y coordinate position (image) of the upper left point of the bounding box*.
 - **Box X width:** Width of the bounding box*.
 - **Box Y height:** Height of the bounding box*.
 - **Barycenter x:** X coordinate position (image) of the center of gravity of the object.
 - **Barycenter y:** Y coordinate position (image) of the center of gravity of the object.
- * The bounding box is the smallest rectangle that completely encompasses the whole object.

 **Note:** The image position (0,0) is located by default in the upper left corner . However, the origin and the direction of the y axis can be modified in the **Spatial Calibration** tool.

2) Basic:

- **Area:** Number of pixels in the object (minus any holes).
- **Perimeter:** Length of the outline of the object (the outline of holes are not considered).
- **Convex area:** Area of the convex hull of the object.
- **Convex perimeter:** Perimeter of the convex outline of the object.
- **Number of holes:** Number of holes in the object.
- **Holes area:** Holes area of the object.
- **Orientation:** Angle between the horizontal axis and the major axis of the ellipse equivalent to the object (0-180 degrees, counterclockwise).



- **Length:** Caliper length along the **orientation** axis of the object.
- **Width:** Caliper length along the **orientation** axis + 90° of the object.
- **Eccentricity:** Ratio between the major and the minor axis of the ellipse equivalent to the object (first and second degree moments).

3) Moments:

All the moments describe the shape of the object.

Three central moments (normalized second-order central moments $X_p Y_q$ ($p+q=2$) of the object).

- **Central moment X1 Y1**
- **Central moment X2 Y0**
- **Central moment X0 Y2**

Seven invariant moments (derived from normalized second and third order central moments) are invariant to translation, rotation and scaling of the object.

- **Invariant moment 1**

- **Invariant moment 2**
- **Invariant moment 3**
- **Invariant moment 4**
- **Invariant moment 5**
- **Invariant moment 6**
- **Invariant moment 7**

Four invariant moments (derived from second and third order central moments) are invariant to translation, rotation, scaling and perspective projection of the object.

- **Invariant affine moment 1**
- **Invariant affine moment 2**
- **Invariant affine moment 3**
- **Invariant affine moment 4**

4) Ratio:

- **Compactness:** Ratio of the **area** of the object to the area of a circle with the same perimeter.
- **Equivalent circular diameter:** Diameter of a circle with the same area as that of the object.
- **Elongation:** Ratio of the **length** to the **width**.
- **Ellipticity:** Ratio of the area of an ellipse (formed with **length** and **width** as axes) to the **area** of the object.
- **Rectangularity:** Ratio of the area of a rectangle (formed with **length** and **width** as sides) to the **area** of the object.
- **Solidity:** Ratio of the **area** of the object to the **convex area**.
- **Convexity:** Ratio of the **convex perimeter** to the **perimeter** of the object.

5) Feret (Additional):

Feret diameters are the distances between the parallel tangents touching opposite sides of the object. The number of Feret diameters from 0 to 180 degrees is defined by an angle (1-90 degrees).

6) Context (Additional):

Context is an additional object descriptor which describes the neighboring environment of the object. A **Background** is required to represent the environment.

7) Texture (Additional):

There are three **texture parameters**, **Mean**, **Standard deviation** and **Entropy**.

►► **See also:**

[Object Extraction](#), [Feret Diameters](#), [Context](#), [Texture](#)

Point Counting



Point counting is a technique to measure the amount of each component present in an image. It consists in laying a grid over the image and counting what is under each grid intersection point.

Point counting is an alternative method to quantify image components stochastically, especially when the contrast between the components is poor or when the image is too complex for using the **Object Extraction** or **Background** tool.

1.1 Grid types

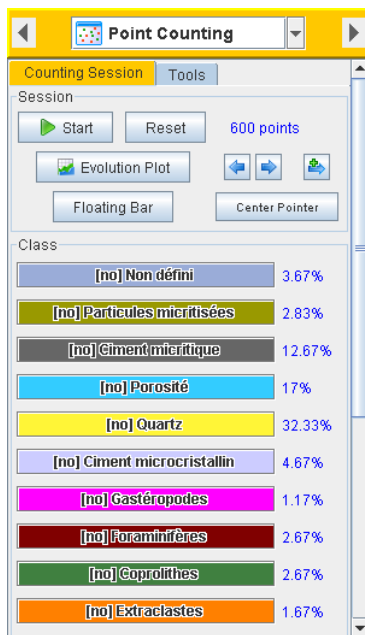
- **No grid (manual)**: This mode allows counting objects in an image manually. Each time you click on an object, it is tagged using a colored point of the selected class in the **Manual** combo box (at the top of the **Class** panel).

Note: To delete a point or modify its class, select **None** in the **Manual** combo box, select the desired point and use the delete command from the contextual menu (right-click).

- **Random grid**: The intersection points are in a uniform random grid, which means that the points are selected randomly. This is the default grid.
- **Recursive grid**: The intersection points are located in a rectangular grid, which is subdivided recursively.


1.2 Starting a counting

- Make your own list of classes with the **Class Editor**.
- Press the **Start** button.
- In the **Point Counting Properties** window:
 - **Grid type**: Select a grid (see Grid types above).
 - **Counting limitation**: If necessary, select a limitation of the counting to a background or to a specified object class (see **Background** or **Object Extraction**).
For showing an alert message after a certain number of points, check the box and enter a number.
 - **Area**: Select the area of the counting (see **Area Editor**).
 - Finally, click **OK** to start the counting.
- For displaying several images simultaneously with a particular zoom factor for each image, create different views (see **Multiview**).
- Press the button of the class (or the related keystroke) that matches what is in the center of the pointer. Repeat this operation until the counting is statistically representative (see **Stopping the counting** below).
- Display the results with the **Data Viewer**.




Reset: Reset the current session.

Floating Bar: Open the **Class** panel in an independent window. Closing the window will replace the panel in the initial state.

 **Note:** Having the list next to the pointer is more comfortable, because you can focus your eyes on the center of the screen during the counting.

Center Pointer: Replace the pointer in the center of the screen.

The two first **Arrow Buttons** allow going from a point to another chronologically. The third one  allows getting a new position (it can also be used for bypassing a position).

Refresh: Refresh the percentages (e.g. when points have been deleted).


 **Note:** Classes can be added, removed or merged even while counting.

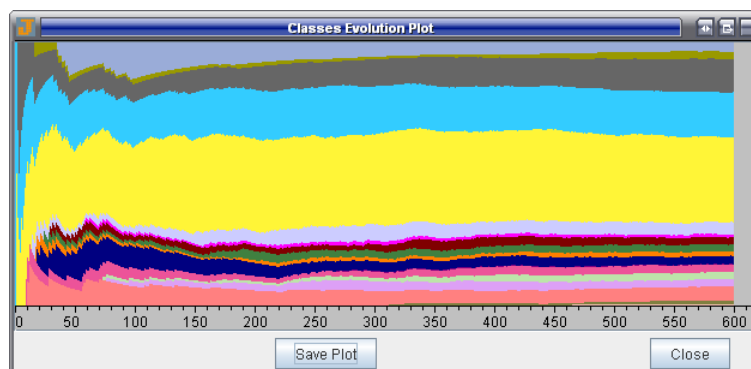
1.3 Modifying the class of a point

Navigate with the **Arrow Buttons** and then click on the desired class button. It is also possible to change the class of a point by double-clicking on the point or by selecting **Properties** in the contextual menu.

1.4 Stopping the counting



There are two ways of stopping the counting:

- The first one is simply by fixing a limit of points in the **Point Counting Properties**. An alert message appears when reaching the limit.
- The second one is by observing a stochastic criterion in the **Evolution Plot**  (see image below). When the percentages are getting stable in the plot, the counting can be stopped.



2) Point Counting preferences

Default grid type and **default point limit** can be changed in the **Point Counting Properties** when starting a new counting.

 **Note:** To display the preferences, open the **Project Preferences** window  and select **Point Counting**.


Background




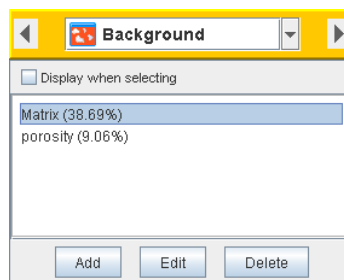
This tool allows calculating the entire surface of a component in the image by **thresholding the histogram**.

In JMicroVision, components can be extracted as objects (e.g. particles) with the **Object Extraction** tool or as backgrounds (e.g. matrix) with the **Background** tool. The background uses less memory, can be computed faster and can be modified. For these reasons, it is recommended to extract as background when only the entire surface of a component needs to be calculated and it is not necessary to partition the background in multiple regions.

The **Image Factory** tool offers some image preprocessing functions and different segmentation methods for more advanced use (see tutorials on the website).

 **Note:** To show the real surface (if calibrated), open the **Data Viewer**.

 **Note:** For computing the background surface in a limited area (by default, the background surface percentage of the whole image is calculated), see **compute result from area**.



1) Background preferences

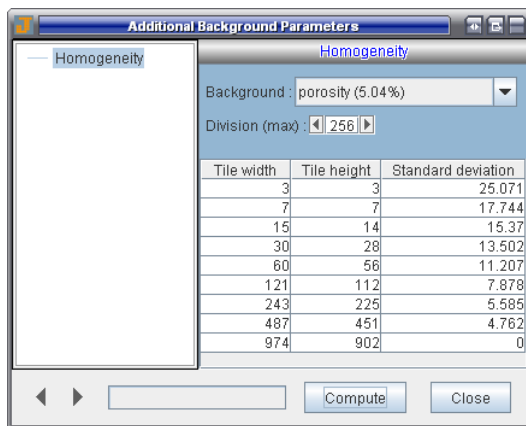
To compute additional parameters, see [the background descriptors](#).

Background Descriptors



1) Homogeneity

This parameter divides the image recursively and computes the **Background** surface for each piece. The **Standard deviation** shows the surface variation between the pieces and the whole image. A constant standard deviation means that the amount of the background surface is the same in different scales.



Background: Select a background from the list.

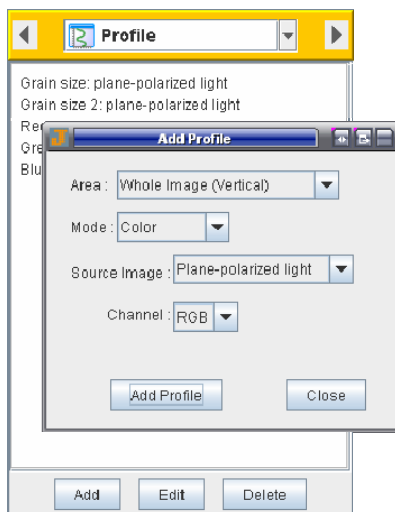
Division (max): The value entered is the number of divisions according to the width and the height of the image.

Tile width and **Tile height** are the size (in pixels) of pieces of the image after the division.

Standard deviation represents the variation between the mean background surface of all pieces and the background surface of the whole image.

Note: To compute the homogeneity parameter, open the **Project Preferences** window , select **Background**, click on the button **Compute Additional Parameters** and select **Homogeneity**. The homogeneity values are not saved in the **JMicroVision Project file**.

Profile



There are four modes for computing profiles: **Color**, **Grain Size**, **Background** or **Object**. The profiles are created from a rectangular area according to a direction in order to show the variation of the grain size, the variation of the pixel color or the variation of a component previously extracted with the **Background** or **Object Extraction** tool.

Note: The views produced by **Multiview** are not visible when using the **Profile** tool.

1.1 Add a profile

Area: Select an area representing the desired region of computing (for adding new profile areas, see **Area Editor**). The central dotted line of the area indicates the direction of the profile.

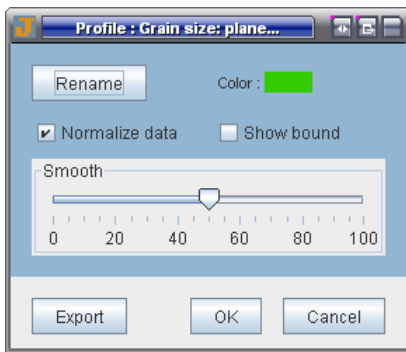
Mode:

- **Color:** This profile shows the variation of the **Channel** pixel intensities (see **Image Types**) of the selected **Source Image**.
- **Grain Size:** This profile shows the variation of the grain size of the selected **Source Image**. This mode does not require any preprocessing or manipulation. However, some conditions must be satisfied to obtain valid results: the contrast between the grains and the background must be sufficient, their color must be

uniform enough and the space between the grains must be constant.
At least 50 pixels of width are required to compute the profile.

- **Background:** This profile shows the variation in the amount of the selected **Background** (see **Background**).
- **Object:** This profile shows the variation in the amount of the selected **Object** (see **Object Extraction**).

1.2 Edit the selected profile



Rename: Edit the profile name.

Color: Choose the color profile.

Normalize Data: Checking this box normalizes the values between 0 and 1, therefore accentuates the variation.

Show Bound: Show the outline of the profile area (dotted line).

Smooth: This slider allows to smooth the profile.

Export: Export in a text file the profile values. In the file, the first line defines the step length from a value to another and the second line is the unit of measure of the step.

2) Profile preferences

Select the **line width** of profiles.


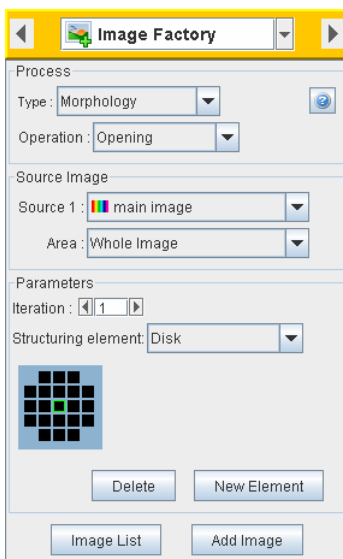




- **Note:** To display the preferences, open the **Project Preferences** window  and select **Profile**.

Image Factory



The Image Factory  allows performing image processing operations. For creating a new image, just select an operation **Type**, an **Operation**, a **Source** image, an **Area** (the operation will be limited to an area, see **Area Editor**), modify if necessary the parameters specific to the selected operation, and finally, for adding the resulting image to the project, click on the **Add Image** button.

- **Note:** When an operation is selected, the resulting image is immediately visible in the main view (the selected image in the toolbar is **current operation**). For getting an optimal speed of image display, it is recommend to set the zoom at 100% (or click on ).

-  **Note:** All images added to the project are saved in the **JMicroVision Project file**. Only the operation on an image is saved instead of the bitmap file, what considerably reduces the file size.
-  **Note:** For renaming, removing or saving an image, see **Image List Editor**.

For getting more information about specific operation types:

- **Segmentation**
- **Arithmetic**
- **Logic**
- **Classic Filtering**
- **Nonlinear Filtering**
- **Gradient Filtering**
- **Binary**
- **Morphology**
- **Pseudo Color**
- **Channel**
- **Enhancement**
- **Rasterize Drawings**
- **Plugins**

1.1 Markers

The markers are vector drawings. They can be useful for retouching an image (c.f. separate objects in a binary image), for creating masks (see **Logic**), for binary **Reconstruction**, for measuring manually objects (c.f. fibers), or for **watershed segmentation** from markers. Once the markers are drawn, they can be rasterized (**Rasterize Drawings** - Freehand Markers) or directly used by **watershed segmentation**.



In the marker toolbar above, the first toggle button allows selecting or moving markers, the second one allows drawing freehand lines (the line thickness is defined in the center of the toolbar). The third one allows creating 2D shapes. The fourth one will delete the selected markers. In the right part of the toolbar, select a class to assign to the markers that will draw. The last button opens the marker class editor (see **Class Editor**). The number of classes is limited to 254.


-  **Note:** For changing the class of a marker, double-click on it or select **Properties** in the contextual menu to display a dialog which permits to change the class.

Image Operations

Segmentation



The goal of image segmentation is to detect and extract the regions which compose an image. Segmentation is a critical step in image analysis and also one of the most difficult tasks with nontrivial images. For these reasons, different methods are provided in JMicoVision.

1.1 Simple Binarization

This operation produces a binary image (see [Image Types](#)). In automatic [histogram thresholding](#) the min threshold is always equal to 0 and the following methods give the max threshold value:

- **Iterative bisection:** For each band an initial threshold is defined to be the midpoint of the range of data represented by the histogram. The mean value is calculated for each sub-histogram and a new threshold is defined as the arithmetic mean of the two sub-histogram means. This process is repeated until the threshold value no longer changes.
- **Max variance:** Calculates the threshold which maximizes the ratio of the between-class variance to the within-class variance for each band.
- **Max entropy:** The entropy of a range of gray levels is defined to be the negation of the sum of products of the probability and the logarithm thereof over all gray levels in the range. The maximum entropy threshold is defined to be the value which maximizes the sum of the entropy of the two ranges which are above and below the threshold, respectively. This computation is effected for each band.
- **Min error:** For each band the histogram is modeled as the sum of two Gaussian distributions and the threshold which minimizes the misclassification error is computed. If the underlying histogram is unimodal the mean value of each band will be returned as the threshold. The bimodality of the histogram for that band will be identically zero.
- **Min fuzziness:** Calculates the threshold which minimizes the fuzziness.

Note: The color of the object (white) is referred to as the foreground color in the resulting image. The rest (black) is referred to as the background color.

1.2 Color or gray intensity threshold

See [Histogram Thresholding](#).

1.3 k-means clustering

See [k-means clustering](#).

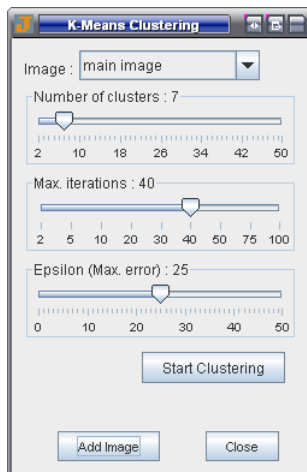
1.4 Watershed

See [Watershed](#).

▶▶ See also:

[Image Factory](#)

k-means clustering



The k-means clustering basically groups pixels according to their color similarity. The number of groups corresponds to the number of the defined clusters. The grouping is done iteratively by minimizing the sum of squares of distances between the pixel intensity and the corresponding cluster centroid. For grayscale images, the process groups pixels according to the gray level similarity.

Note: For aborting the process, click on the progress bar.

Image: Select an image.

Number of clusters: Set the final number of clusters.

Max. iterations: Define the maximum number of iterations to stop the clustering.

Epsilon (Max. error): Define the tolerable error. When the difference between the pixels less the cluster centroids in the last and the previous iteration is less than the error, the clustering is stopped.

▶▶ See also:

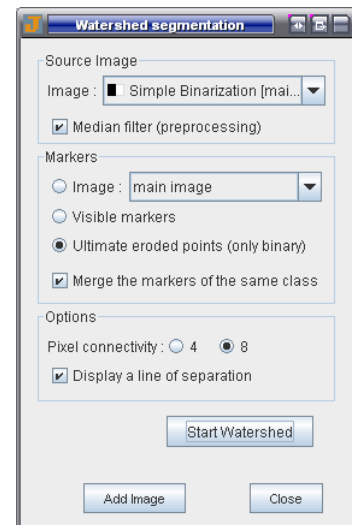
[Image Factory](#)

Watershed



Image segmentation is the process of isolating objects in the image from the background. Separating touching objects in an image is really a difficult task. The watershed transform is often applied to this problem. Another method available in JMicroVision is **Object Separation**.

The watershed from markers can be intuitively described as the flooding simulation, where the image is treated as a surface (light pixels are high and dark pixels are low). This surface is submerged slowly in water. From each of the holes, the flooding water is painted with the color of the marker. At any time that different colored water coming from different holes meet, a dam is built to avoid their mixing. This mechanism continues progressively until the brightest pixel is reached. The final dams are the watershed lines and the colored water is the segmented regions.



Note: To obtain better result, the source image is automatically transformed into Sobel edge image (see **gradient filtering**). The gradient tends to be high at the borders of the objects and low inside the objects.

This implementation always uses markers to control the watershed segmentation. The marker-controlled method leads to segmentation results in which one object (segmented region) corresponds to only one marker.

Note: The watershed process requires to handle the whole image in memory, thus it can be applied on very high-definition images only if the computer has enough memory and the setting of JMicroVision has been adapted, see **Memory Management**.

1.1 Source Image

Image: Select an image as input for watershed.

Median filter (pre-processing): Checking this box applies to the input image a median filter (size: 7x7), which reduces noise while it preserves the position of borders.

1.2 Markers

Each marker results in one region. When the markers selection is an image, each region characterized by a color or a gray level is considered as a marker, except black region that represents the background. There are different ways to obtain an image of markers. The most common one is the **ultimate eroded points** image. However, a combination of morphological operations can produce an image of markers.

- **Image:** Select an image where each non-black region (pixels in 8 **pixel connectivity**) represents a marker.

Note: It is recommended to use grayscale image as markers, because during the process, the color image is converted to a grayscale image. Therefore, different colors can be transformed into the same gray level.

- **Visible markers:** Use the drawn **markers** that are visible.


- **Ultimate eroded points (only binary)**: This mode requires a binary image as source and will produce the watershed lines image. The ultimate eroded points operation can be divided in three steps. Firstly, the **Euclidian Distance Map** image is computed. Secondly, the latter image is successively eroded until it remains only the local maximum of brightness (the local center of white regions). And thirdly, the image is smoothed by a Gaussian filter (5x5). Finally, to obtain the object separation, a **logical And** operation must be performed between the watershed lines image and the source image. This separation method is suitable for convex and relatively round particles.

Merge the markers of the same class: This option has an effect only if the **Display a line of separation** button is checked. It removes the boundary lines of regions, which belongs to the same class. Do not check this button if all the markers have the same color.


 **Note**: When **Ultimate eroded points** is selected, this option has no effect.

1.3 Options

Pixel connectivity: Define the **pixel connectivity** when the algorithm floods the regions.

 **Note**: When 8 is selected, the black boundary line is larger (pixels in 4 pixel connectivity). This enables to extract each region as an object (in **Object Extraction**), if 4 is selected, the connected regions will be merged into one object.

Display a line of separation: Display the boundary line of segmented regions.

 **Note**: This option must be activated to distinguish the boundary of region, especially when all the markers have an identical color or with binary images.


» **See also:**

[Image Factory](#)

Arithmetic



The resulting value for each pixel is an arithmetical operation between the input value of the **source 1** and the **source 2**.

 **Note**: Overflowing pixel values resulting of an arithmetic operation (e.g. $225 + 85 = 255$) are set to the maximum allowed value (255). Alternatively the pixel values are wrapped around from zero.

Rules for determining the type of the resulting image when the two source images have different types (see [Image Types](#)):

- **grayscale** + **color** (only the first band is involved: red channel) = **grayscale**
- **grayscale** + **binary** (converts into gray levels) = **grayscale**
- **color** + **binary** (converts into color) = **color**

1.1 Add

$$R(i,j) = SRC1(i,j) + SRC2(i,j)$$

1.2 Subtract

$$R(i,j) = SRC1(i,j) - SRC2(i,j)$$

1.3 Multiply

$$R(i,j) = SRC1(i,j) * SRC2(i,j)$$

1.4 Divide

$$R(i,j) = SRC1(i,j) / SRC2(i,j)$$

Logic



The resulting value for each pixel is a logical operation between the input value of the **source 1** and the **source 2**, except the **Not** operation that requires only one source image.

The truth-table below describes the logical operations only for binary images. Logical operations can also be carried out on images with pixel values between 0 and 255 (grayscale and color images). In this extension, the logical operations are carried out on binary representations of these values. For instance, suppose we wish to "**And**" the values 83 and 71 together. 83 is 1010011 in binary, and 71 is 1000111. After the logical **And** operation, we obtain 1000011 in binary or 67 in decimal.

Rules for determining the type of the resulting image when the two source images have different types (see [Image Types](#)):

- **grayscale** + **color** (only the first band is involved: red channel) = **grayscale**
- **grayscale** + **binary** (converts into gray levels) = **grayscale**
- **color** + **binary** (converts into color) = **color**

Note: A binary image can be used as a mask when a logical **And** is performed between a binary and a color or grayscale image.

1.1 Or

Logical **Or** truth-table:

src1	src2	Result
0	0	0
0	1	1
1	0	1
1	1	1

1.2 And

Logical **And** truth-table:

src 1	src 2	Result
1	1	1
1	0	0
0	1	0
0	0	0

1.3 Not (Invert)

The logical **Not** or the invert is an operator which takes the **source 1** image as input and produces its photographic negative.

Logical **Not** truth-table (only for binary images):

src 1	Result
1	0
0	1

For all image types, the resulting value for each pixel is the input value subtracted from 255. $R(i,j) = 255 - SRC1(i,j)$

1.4 Xor


Logical **Xor** truth-table:

src 1	src 2	Result
0	0	0
0	1	1
1	0	1
1	1	0

Classic Filtering



Classic or linear filtering is a mathematical operation called convolution. Convolution involves the multiplication of a group of pixels in the input image with an array of pixels in a convolution kernel (or mask). The output value produced in a spatial convolution operation is a weighted average of each input pixel and its neighboring pixels in the convolution kernel.

 **Note:** Filtering a binary image will produce a grayscale image.

Linear filtering groups a large number of smoothing, sharpening, noise reduction and edge filters. Here are the preestablished kernels:

1.1 Mean (3x3)

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

1.2 Blur (3x3)

0	1/8	0
1/8	4/8	1/8
0	1/8	0

1.3 Blur more (3x3)

1/14	2/14	1/14
2/14	2/14	2/14
1/14	2/14	1/14

1.4 Sharpen (3x3)

0	-1/4	0
-1/4	8/4	-1/4
0	-1/4	0

1.5 Sharpen more (3x3)

-1/4	-1/4	-1/4
-1/4	12/4	-1/4
-1/4	-1/4	-1/4

1.6 Defocus (3x3)

1	1	1
1	-7	1
1	1	1

1.7 Laplacian 1 (Edge) (3x3)

0	-1	0
-1	4	-1
0	-1	0

Laplacian 2 (Edge) (3x3)

-1	-1	-1
-1	8	-1
-1	-1	-1

1.8 Strong Edge (5x5)

-2	-2	-2	-2	-2
-2	-3	-3	-3	-2
-2	-3	53	-3	-2
-2	-3	-3	-3	-2
-2	-2	-2	-2	-2

1.9 Outline (5x5)


1	1	1	1	1
1	0	0	0	1
1	0	-16	0	1
1	0	0	0	1
1	1	1	1	1

1.10 Emboss (3x3)

-5	0	0
0	1	0
0	0	5

1.11 Gaussian

Gaussian filters are used to smooth edges (high contrast) and reduce noise. The kernel has a shape of a Gaussian hump.

 **Note:** To get kernels larger than (9x9), press the **New Filter** button, choose a kernel size and then click on **Gaussian kernel**.

1.12 Custom Filter

To open the Custom Filter dialog, press the **New Filter** button. For removing a custom filter, select the desired filter and press the **Delete** button.

Kernel size: Define the size of the kernel.

Kernel origin: X and Y position of the kernel origin (the position of the first cell is 0,0).

Divisor: Divide all elements by this number.

Apply Filter: Apply the filter to the current image.

Add Filter: Add the current filter to the filters list (The filter is set permanently to JMicroVision).

Nonlinear Filtering



Nonlinear filtering is based on a logical decomposition of a filter into a set of relatively simple operators.

Filter size: Define the size of the kernel.

Filter shape:

- Square

x	x	x
x	x	x
x	x	x

- Separable square is similar to square except for **Median** filter, where the separable median is defined as the median of the medians of each row. For example, if the pixel values in a 3x3 window are equal to:

1	2	3
5	6	7
4	8	9

Then the overall (non-separable) median value is 5, while the separable median is equal to the median of the three row medians: median(1, 2, 3) = 2, median(5, 6, 7) = 6, and median(4, 8, 9) = 8, yielding an overall median of 6.

- Plus

	x	
x	x	x
	x	

- X

x		x
	x	
x		x

1.1 Median

A median filter is based upon moving a window over an image (as in a convolution) and computing the output pixel as the median value of the pixel intensities within the input window. The effect is that noise is removed while the position of edges is preserved. The median filter is very efficient for the removal of impulse noise (also known as binary noise).

1.2 Min

This filter is similar to the median filter, except that it computes the output pixel as the minimum value.

1.3 Max

This filter is similar to the median filter, except that it computes the output pixel as the maximum value.

1.4 Kuwahara

The kuwahara filter is an edge preserving blur filter as the median filter. It works by calculating the mean and variance for four subquadrants, and chooses the mean value for the region with the smallest variance.

Gradient Filtering



The gradient filtering operation is an edge detector which computes the magnitude of the image gradient vector in two orthogonal directions. The operation may be defined as:

$$G = \sqrt{G_x^2 + G_y^2}$$

where G_x and G_y are the horizontal and vertical gradient images generated from the source image by correlating it with the supplied orthogonal (horizontal and vertical) gradient masks (K_x and K_y).

1.1 Sobel

$$K_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad K_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

1.2 Roberts

$$K_x = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad K_y = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

1.3 Prewitt

$$K_x = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \quad K_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

1.4 Freichen

$$K_x = \begin{bmatrix} 1 & 0 & -1 \\ 1.414 & 0 & -1.414 \\ 1 & 0 & -1 \end{bmatrix} \quad K_y = \begin{bmatrix} -1 & -1.414 & -1 \\ 0 & 0 & 0 \\ 1 & 1.414 & 1 \end{bmatrix}$$

►► See also:

[Image Factory, Classic Filtering](#)

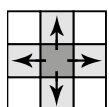
Binary



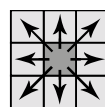
Binary operations can be performed exclusively on binary images. In binary images, white regions represent the foreground (pixels in 8 pixel connectivity) and black regions the background (pixels in 4 pixel connectivity).

Pixel connectivity:

Connectivity defines which pixels are connected to other pixels to form a region (e.g. object, hole).



4-connected: only along the horizontal and the vertical direction.



8-connected: in all directions (including diagonal direction).

1.1 Hole Fill


Fill the holes of objects (white regions). The background and the holes are characterized by black pixels in 4 pixel connectivity. Holes are defined as black regions, which are not connected to an image border.

1.2 Border Kill

Remove all the objects (white regions) that are connected to an image border.

1.3 Euclidian Distance Map

Each white pixel is converted to a gray value that represents the distance from the nearest black pixel. Therefore, the resulting image is a grayscale image.

 **Note:** When the white regions are small, the resulting image seems to be completely black. Use the **Automatic levels** operation to stretch the gray levels distribution.

1.4 Reconstruction

Binary reconstruction allows selecting some white regions using a second binary image. The second image (**source 2**) contains markers (white regions). The reconstruction selects all the white regions in the **source 1** image that are connected at least to one marker or a part of it (to create binary image of markers, see **Markers**).

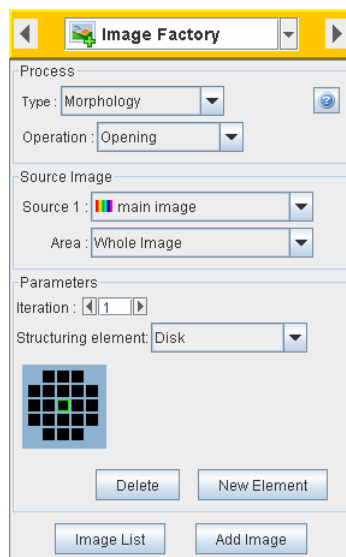
1.5 Gray Levels Conversion

Converts a binary image to a grayscale image (see **Image Types**).

1.6 Color Conversion

Converts a binary image to a color image (RGB) (see **Image Types**).

Morphology



Like spatial filters, morphological operations also work with pixel neighborhood. However, kernel coefficients do not multiply the neighborhood values. In mathematical morphology, the kernel is a structuring element. The two most basic operations are erosion and dilation also known as Minkowski subtraction and addition.

The operators are particularly useful for edge detection, noise removal and image enhancement.

Prestablished kernels: Basic kernels are available in the structuring element list (Square, Disk, Disk 2, Cross, Ring, Horizontal line, Vertical line).

However, it is possible to add new custom structuring elements to the list by pressing the **New Element** button. For removing a custom element, select the desired element and press the **Delete** button.

The Custom Structuring Element dialog allows to create common elements with two values (0 and 1) and grayscale structuring elements (integers from 0 to 255). Here zeros represent blank values which do not belong to the structuring element.

Kernel Size: Define the size of the kernel.

Kernel origin: X and Y position of the kernel origin (the position of the first cell is 0,0).

 **Note:** The small green rectangle indicates the origin.

Apply Element: Apply the structuring element to the current image.

Add Element: Add the current element to the structuring element list (The element is set permanently to JMicroVision).

1.1 Erosion

This operation erodes bright regions on the image.

1.2 Dilation

This operation dilates bright regions on the image.

1.3 Opening

This operation (Erosion followed by Dilation) separates narrowly connected bright regions and removes small ones.

1.4 Closing

This operation (Dilation followed by Erosion) fills holes in bright regions and connects bright regions that are close.

1.5 Internal Gradient

This operation is an edge detector that extracts the internal contours of bright regions. The eroded image is subtracted from the original one.

1.6 External Gradient

This operation is an edge detector that extracts the external contours of bright regions. The original image is subtracted from the dilated one.

1.7 Black TopHat

This operation detects small dark regions. The closed image is subtracted from the original one.

1.8 White TopHat


This operation detects small bright regions. The opened image is subtracted from the original one.

Pseudo color




Pseudo Color operation replaces the standard RGB lookup table (LUT, palette that defines the actual color of the pixel according to its value) by one in the list (Red, Green, Blue,

Red to yellow, Fire, Ice, Hot, Flag, IHS, Spectrum, Rainbow 1, Rainbow 2, Rainbow 3, Rainbow 4, Random). Pseudo colors helps to bring out features or details in an image.

 **Note:** Use the **Refresh** button to get new values of the **Random** LUT.

Offset: Shift the lookup table values.

 **Note:** **Rainbow 3** and **IHS** LUT have continuous values between the last and the first values. Therefore, they are more suitable for a shift of values.

Channel



Channel operations allow extracting or merging the channels (bands) of color images.

1.1 Extraction

When the input image is a color image, this function allows extracting one channel (using the **RGB** or **IHS** color space) or transforming a color image into a **Real Gray** image (see **Color image**).

1.2 Merging

Allows creating a color image in the **RGB** or **IHS** space by selecting 3 channels of any image.

RGB:

Channel 1 ⇒ Red

Channel 2 ⇒ Green

Channel 3 ⇒ Blue

IHS:

Channel 1 ⇒ Intensity

Channel 2 ⇒ Hue

Channel 3 ⇒ Saturation

Enhancement



Image enhancement adjusts pixel values to improve the perception of information in images for the human visual system. Enhancement techniques are not very useful as pre-processing operations for segmentation, except background subtraction (that is not directly an enhancement technique, but more a correction operation), which attempts to make uniform the illumination of the background to facilitate objects segmentation.

1.1 Manual Enhancement

Contrast: When increasing contrast, pixels that are darker than 127 (pixel's value) are made darker and pixels that are lighter than 127 are made lighter. This works fine when the values in the histogram is distributed around the middle; otherwise, contrast will not have the desired effect.

Luminosity: Increasing or decreasing the luminosity shifts the histogram.

Gamma: Moving the slider to the right (greater than one) will increase the number of dark values and darken the image; moving to the left (less than one) will increase the number of light values, lightening the image. The mapping function is an exponential curve.

1.2 Automatic Levels

Automatic levels (histogram normalization) stretch an image's pixel values to cover the entire pixel value range (0-255).

1.3 Equalized Levels

This operation attempts to flatten the image histogram. Histogram equalization employs a non-linear mapping which re-assigns the intensity values of pixels in the input image so that the output image contains a uniform distribution of intensities.

1.4 Background Subtraction


This operation attempts to make uniform the background illumination when, for instance, an image has been acquired by a microscope with a nonuniform illumination. There are two ways to correct this issue:

- 1) Take a picture of the scene without specimen (only the background). Then, take another picture with the specimen.
Select the image with the specimen in **Source 1** and the one without the specimen in **Source 2**.
Finally, adjust the **Luminosity offset** slider to correct the luminosity of the resulting image.
- 2) When you do not have a picture with only the background, press the **Create Artificial Background** button.
Set at least 6 points that match to the background of the image (using the **Add Point** button). The values of these points are used by a polynomial fit function to create an artificial background.
Finally, press **Add image** in the Artificial Background dialog. Close the dialog and then follow the instructions at the point 1.

Rasterize Drawings



This operation allows creating a new image by rasterizing vector drawings.

 **Note:** To show the resulting image without the superposition of the vector drawings, use the lens or the multiview tool and uncheck **Show Drawings**.

Incorporate the source 1 image: When the button is checked, the selected image is included in the background; otherwise the background color is the one selected in the [display preferences](#).

Output type: Select an image type (see [Image Types](#)).

Refresh: Actualize the image. Use this button after showing or hiding drawings.


1.1 Visible Drawings

This mode makes an image with all the visible vector drawings (see **Display**).

 **Note:** Adding the image to the project requires to save it on the disk.

1.2 Freehand Markers

This mode makes an image with all the visible markers (see **Image Factory**) and does not require to save the resulting image on the disk. For this reason, it is recommended to use this mode when it is possible.

 **Note:** Deleting or adding markers will not affect the image created previously from markers.

Plugins



This feature allows creating your own image operations or analysis. The plugins are placed in the "plugins" folder located in the installed directory of JMicroVision. For more information, see the documentation about plugins on the website.

Glossary

A

Algorithm: A set of well-defined rules or procedures for solving a problem or providing an output from a specific set of inputs.

B

Background: The part of a scene behind the object to be imaged. In JMicroVision, this is an alternative to the Object Extraction tool. It is more suitable to use background for highly connected objects (e.g. matrix, cement). The background descriptors are applied to the whole background, not to individual regions.

Bicubic Interpolation: see Interpolation.

Bilinear Interpolation: see Interpolation.

Binary Image: A black and white image represented as a single bit containing either zeros or ones, in which objects appear as silhouettes. The result of thresholding.

Bit: An acronym for a BInary digiT. It is the smallest unit of information which can be represented. A bit may be in one of two states, on or off, represented by a zero or a one.

Blob: A single, connected region in a binary or grayscale image.

BMP (Bitmap): The standard Windows image format on DOS and Windows compatible computers. The BMP format supports RGB (red, green, blue), indexed-colors, grayscale, and Bitmap color modes. BMP files have the .bmp extension. Lossless compression can be specified, but some programs use only uncompressed files.

Bounding Box: The four coordinates which define a box around the object parallel to the major and minor axis.

Brightness: The total amount of light or incident illumination on a scene or object per unit area. Also called intensity.

Byte: Eight bits of digital information. A byte has values from 0 to 255, and is the unit most common to represent the gray scale value of one pixel.

C

Calibration: The act of relating X and Y pixel spacing to known or predetermined pixels per unit length (i.e. inch, mm) factor. Often involves adjusting the image position in setup.

CIE: An acronym for a chromaticity coordinate system developed by the "Commission Internationale de l'Eclairage", the international commission on illumination. In the CIE system, a plot of ratios (x, y and z) of the three standard primary colors (tristimulus values) to their sum. The most common diagram is the 2 dimensional CIE (x,y).

Classification: An assignment of image objects to one of two or more possible groups. Decisions are made by evaluating features either 1) structurally, on the basis of relationships or, 2) statistically. For example, 1) a penny is round, has a certain diameter (+/- a tolerance) and has a histogram of a mean value, or 2) statistically, the object is measured a number of times, then the average and the standard deviation are recorded. After training, the features are weighted based on significance in object identification. For multiple features, absolute values are used.

Color: A visual object attribute which may be described by a "coordinate system" such as hue, saturation and intensity (HSI), CIE or LAB. Wavelengths in the visible part of the electromagnetic spectrum to which retinal rods respond.

Color Space: A two or three dimensional space used to represent color coordinates. RGB, HSI, LAB and CIE are all representations of color spaces.

Contrast: The difference of light intensity between two adjacent regions in the image of an object. Often expressed as the difference between the lightest and darkest portion of an image. Contrast between a flaw or feature and its background is the goal of illumination.

Convex Hull: The region enclosed by the rubber band is called the convex hull of the object. The rubber band bridges over indentations around the periphery of the feature.

Convex Perimeter: The convex perimeter is the length of the convex hull or bounding polygon, and bridges over indentations around the periphery of the feature.

Convolution: Superimposing an m x n operator (usually a 3x3 or 5x5 mask) over an area of the image, multiplying the points together, summing the results to replace the original pixel with the new value. This operation is often performed on the entire image to enhance edges, features, remove noise and other filtering operations.

D

Digital Camera: The newest generation of video cameras transforms visual information into pixels and translates each pixel's level of light into a number in the camera.

Digital Image: A video image converted into pixels. The numeric value of each pixel's value can be stored in a computer memory for subsequent processing and analysis.

F

Fast Fourier Transform: This transform produces a new image which represents the frequency domain content of the spatial or time domain image information. Data is represented as a series of sinusoidal waves.

- Feature Vectors:** A set of features of an object (such as area, number of holes, etc) that can be used for its identification or inspection.
- Features:** Simple image data attributes such as pixel amplitudes, edge point locations and textural descriptors, center of mass, number of holes in an object with distinctive characteristics defined by boundaries or regions.
- Filter:** A device or process that selectively transmits frequencies. In optics, the material either reflects or absorbs certain wavelengths of light, while passing others. Filters can be used to sharpen or smooth image details, or to extract edges from objects.
- Fluorescence:** The emission of light or other electromagnetic radiation at longer wavelengths by matter as a result of absorption of a shorter wavelength. The emission lasts only as long as the stimulating irradiation is present.
- FPX (Flashpix):** An image format in 8-bit grayscale or 24-bit color, uncompressed or with lossy compression. Provides multiple resolutions of each image.

G

- Gamma:** The numeric value for the degree of contrast in a video signal. The exponent in the power law relating output to input signal magnitude. Non-linear camera tube.
- GIF (Graphic Interchange Format):** Image format that supports only 255 colors, so requires lossy quantization for full-color photos. GIF files are best used with large areas of solid colors, such as logos, and simple illustrations with flat colors. GIF files are cross-platform and will work on all computer types. Uses lossless, patented LZW compression. This patent expired in the United States in 2003, but is still effective in Europe, Japan and Canada.
- Gray Level:** A quantized measurement of image irradiance (brightness), or other pixel property typically in the range between pure white and black.
- Gray Scale:** Variations of values from white, through shades of gray, to black in a digitized image with black assigned the value of zero and white the value of one.
- Grayscale Image:** An image consisting of an array of pixels which can have more than two values. Typically, up to 256 levels (8 bits) are used for each pixel.

H

- Histogram:** A graphical representation of the frequency of occurrence of each intensity or range of intensities (gray levels) of pixels in an image. The height represents the number of observations occurring in each interval.
- Histogram Analysis:** Determination of the presence or absence of a feature or flaw based on the histogram values in a certain gray scale region.
- Histogram Equalization:** Modification of the histogram to evenly distribute a narrow range of image gray scale values across the entire available range.
- HLS (Hue-Lightness-Saturation):** see HSI
- Hough Transform:** A global parallel method for locating both curved and straight lines. All points on the curve map into a single location in the transform space.
- HSI (Hue-Saturation-Intensity):** In this color representation system, hue is the color as described by wavelength, for instance the distinction between red and yellow. Saturation is the amount of the color that is present, for instance the distinction between red and pink. The third axis (called lightness, intensity or value) is the amount of light, the distinction between a dark red and light red or between dark grey and light grey. A mathematical conversion from RGB: $H = [\text{PI}/2 - \text{arc tan}$

$$(2 * R - G - B) / 1.732 * (G - B) + \pi; G < B] / 2 * \pi; I = (R + G + B) / 3; S = 1 - [\min(R, G, B) / I]$$

HSI Conversion: A mathematical conversion from the RGB color space to hue, saturation and intensity values.

HSV (Hue-Saturation-Value): See HSI.

Hue: One of the three properties of HSI color perception. A color attribute used to express the amount of red, green, blue or yellow a certain color possesses. White, gray and black do not exhibit any hue. See HSI.

I

IHS (Intensity-Hue-Saturation): see HSI

Image Analysis: Evaluation of an image based on its features.

Image Enhancement: Image processing operations which improve the visibility of image details and features.

Image Processing: Digital manipulation of an image to aid feature visibility, make measurements or alter image contents.

Image Resolution: The number of rows and columns of pixels in an image.

Intensity: See HSI.

Interlaced Scanning: A scanning process in which all odd lines then all even lines are alternately scanned. Adjacent lines belong to different fields.

Interpolation: A mathematical method of regenerating missing or needed information, such as nearest neighbor, bilinear and bicubic interpolation. The bicubic fit is more isotropic than the bilinear method. Interpolation always has the effect of smoothing the image and removing some high frequency information.

J

Java Virtual Machine: A Software that serves as interpreter between Java bytecode and a specific operating system (allows Java applications to run on any platform without changing the code).

JPEG (Joint Picture Experts Group): An image compression standard, lossy compression, uses the discrete cosine transform. The quality can vary greatly depending on the compression settings. JPEG can have millions of colors and are often used for photographs and very complex images. JPEG files are cross-platform and will work on all computer types. JPEG images have the .jpg, .jpe, or .jpeg extension.

L

Look-Up Table (LUT): Software definition: palette of color or pseudo-color values (usually 256 per channel). Hardware definition: High speed digital memory used to transform image input values to outputs for thresholding, windowing and other mappings such as pseudo-color.

Luminance: The black-and-white part of a video signal. Also called Luma.

M

Median Filter: A method of image smoothing (convolution) which replaces each pixel value with the median grayscale value of its immediate neighbors.

Micron: One millionth of a meter, also called a micrometer.

N

Nearest Neighbor Interpolation: see Interpolation.

O

Object Descriptors: see Object Features

Object Features: Any characteristic that is descriptive of an image or region, and useful for distinguishing one from another. A feature may be any measurable item such as length, size, number of holes, surface texture amount or center of mass.

P

Pattern Recognition: A process which identifies an object based on analysis of its features.

Pixel: An acronym for "picture element." The smallest distinguishable and resolvable area in an image. The discrete location of an individual photo-sensor in a solid state camera.

Platform: A platform is a version of interface software meant for a specific computer. Examples of such software are DOS, Windows, Windows95, AS400, Solaris, Linux, and Macintosh platforms. there are many more.

PNG (Portable Network Graphics): PNG image format was developed as an alternative to GIF. PNG files support 24-bit images and produce background transparency without jagged edges. Some older versions of Web browsers may not support PNG images. Like GIF and JPEG files, PNG files are cross-platform and compressed.

Polarized Light: Light which has had the vibrations of the electric or magnetic field vector typically restricted to a single direction, in a plane perpendicular to its direction of travel. It is rated by a type of filter which absorbs one of the two perpendicular light rays. Crossing polarizers theoretically blocks all light transmission.

R

RAM (Random Access Memory): Computer memory that allows data to be read or written at a particular location without having to pass sequentially through preceding locations.

Raster: A series of scan lines that make up a television or computer display. All of the scan lines that make up a frame of video form a raster. Most images are stored in a raster format, as a series of lines.

Region: Area of an image. Also called a region of interest.

Resolution: A basic measurement of how much information is on the screen. Higher resolution means much more details in the image.

RGB (Red, Green, Blue): The three primary color system used for video color representation.

S

Saturation: The amount of color present. One of the three properties of color perception along with hue and intensity. See HSI.

Scaling: A method of altering the size of an image in the x and/or y dimensions, which changes the effective resolution of the image.

Segmentation: The process of dividing a scene into a number of individual objects or contiguous regions, differentiating them from each other and from the image background.

T

Thresholding: The process of converting gray scale image into a binary image. If the pixel's value is above the threshold, it is converted to white. If below the threshold, the pixel value is converted to black.

TIFF (Tagged-Image File Format): TIFF is a very common image format and is used to exchange files between applications and computer platforms. TIFF format is supported by virtually all paint programs, image editing and page layout applications. The TIFF format supports CMYK, RGB and grayscale files. TIFF files have the .tif or tiff extension. Lossy and lossless compression available, but many programs only support a subset of available options.

Tiling: Tiling is supported only by Tiff, flashpix et jpeg2000 format. Tiled images are made up of tiles. Each tile can be processed and stored in memory separately. Tiles may be stored in a centrally-maintained cache for performance. Tiling also facilitates the use of multiple threads for computation. Previously allocated tiles may also be re-used to save memory.

W

WBMP (Wireless Application Protocol Bitmap Format): The Wireless Bitmap image format is mainly used in the Wireless Application Protocol (WAP).

X

XYZ: CIE standard color space, used for comparing colors and in psychophysical experiments. XYZ are generally obtained from video cameras via a single linear transformation.

Index

1D Measurement.....	44	Freehand Markers.....	74
2D Measurement.....	46	Freichen.....	69
And.....	65	Gamma.....	73
Area Editor.....	32	Gaussian.....	66
Arithmetic.....	64	General Description.....	41
Automatic Levels.....	73	Gradient Filtering.....	69
Background.....	57	Help.....	9
Background Descriptors.....	57	Help Menu.....	20
Background Subtraction.....	73	Histogram Thresholding.....	33
Binary.....	70	Hole Fill.....	70
Black TopHat.....	71	Hue.....	34, 73
Blur.....	66	IHS.....	34, 73
Border Kill.....	70	Image Channels.....	34
Bug Reports.....	7	Image Factory.....	59
Calibration.....	42	Image List Editor.....	25
Channel.....	73	Image Rectification.....	29
Class Editor.....	36	Import Image.....	21, 29
Classic Filtering.....	66	Intensity.....	34, 73
Classification.....	51	Internal Gradient.....	71
Closing.....	71	Invert.....	65
Coloring Objects.....	35	Iterative Bisection.....	61
Context.....	37	Keyboard shortcuts.....	11
Contrast.....	73	K-Means Clustering.....	62
Convert To Tile TIFF.....	18	Kuwahara.....	68
Custom Filter.....	66	Labeling.....	35
Data Viewer.....	30	Laplacian.....	66
Different Coordinates.....	29	Lens.....	28
Dilation.....	71	Logic.....	65
Display.....	26, 28	Look and Feel.....	19
Emboss.....	66	Luminosity.....	73
Enhancement.....	73	Magic Wand.....	50
Equalized Levels.....	73	Main window.....	17
Erosion.....	71	Markers.....	59, 63
Euclidian Distance Map.....	70	Max.....	68
External Gradient.....	71	Max Entropy.....	61
Feret Diameters.....	37	Max Variance.....	61
Filter.....	66, 68, 69	Mean.....	66

Median	68	Rasterizing	74
Min	68	Real Gray	34, 73
Min Error.....	61	Reconstruction	70
Min Fuzziness	61	RGB	34, 73
Morphology	71	Roberts.....	69
Multiview	28	Saturation.....	34, 73
Nonlinear Filtering	68	Saving Project	21
Not	65	Segmentation	61
Note.....	41	Sharpen.....	66
Object Descriptors.....	52	Shift Image Origin	42
Object Extraction	47	Simple Binarization.....	61
Opening.....	71	Slide Show	24
Optimizing	10	Sobel.....	69
Options Menu	19	Spatial Calibration	42
Or.....	65	Spectrum	72
Outline	66	Strong Edge	66
Pixel connectivity	70	Support	8
Plugins	75	Supported File Types	12
Point Counting	55	System Requirements	9
Prewitt.....	69	Texture.....	38
Program History	4	Toolbar	24
Project Menu	22	Vertical Profile	58
Project Preferences.....	30	Visible Drawings.....	74
Project Window	23	Wand	50
Pseudo color	72	Watershed.....	63
Rainbow.....	72	White TopHat.....	71
Random LUT.....	72	Windows Menu.....	20
Rasterize Drawings.....	74	Xor.....	65