

EdgeFitter Plugin

This plugin for ImageJ provides an automatic method for determining the width of an object in an image. Specifically, it takes a line selection which is drawn over an object by the user, and corrects the line so that its endpoints to lie on the edge of the object.

Although it is possible to perform this task by hand, it usually takes time for the user to accurately determine the boundaries of an object under inspection. When faced with performing this task many times, this expense for accuracy can add up, resulting in the entire process taking an especially long time. Additionally, when manually determining the boundaries of an object, the user may inadvertently introduce bias into the process, which can influence the results of the measurements.

This plugin solves these problems by automatically and consistently performing the boundary determination for an object. All the user needs to do is draw a rough and inexact line across the object of interest, and the plugin will fit the line so that its endpoints line on the edges of the object.

Installing the Plugin

For ImageJ to be able to use the plugin, it must first be installed in the plugins folder within the ImageJ application folder. On Windows machines, the ImageJ folder is usually located in the Program Files folder, which is located in the root of the hard drive. To access it, select the “Run” item from the Windows Start Menu, then when it appears type “C:\Program Files” into the “Open” box, and click OK. On Macintosh machines, the ImageJ folder is usually located within the Applications folder. To access it, double-click on the Hard Drive icon on the Desktop, and then open the Applications folder in the resulting window. Once the plugin is placed in the plugins folder, which is inside the ImageJ application folder, the next time ImageJ starts up, an option to run the EdgeFitter plugin will exist in the Plugins menu.

There are two files that constitute the EdgeFitter plugin: EdgeFitter_.class and EdgeFitter_.java. The .class file contains the actual plugin code that ImageJ runs, and as such, it is required for the plugin to work. The .java file contains the human-readable source code for the plugin, and is therefore optional. Those with a curiosity for seeing how the code works or for making changes to the plugin may be interested in obtaining and installing the .java file along with the .class file, but all that is necessary for the plugin to work is the .class file. These files belong in the aforementioned plugins folder within the ImageJ application folder.

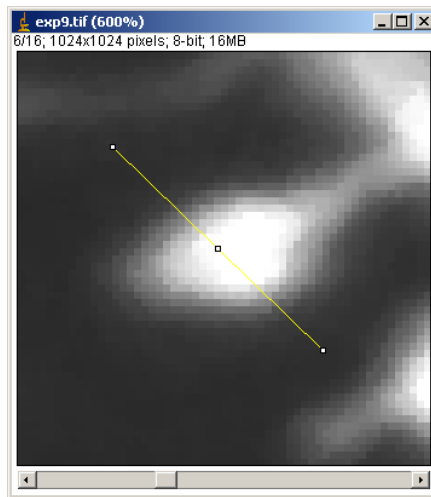
Source Image Requirements

Since ImageJ can understand virtually any kind of image format, this plugin can be used to analyze the image content of virtually any kind of file. Apart from the file format, however, there are some content restrictions that must be satisfied for the plugin to work on an image. First of all, the image must be an 8-bit grayscale image. If it is not already, ImageJ can convert the image to an 8-bit grayscale image by going to the “Image” menu,

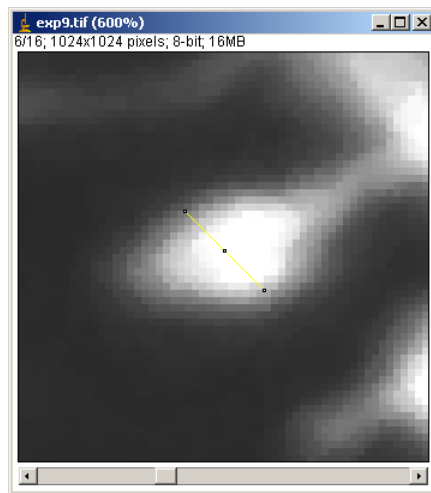
opening the “Type” submenu, and selecting the “8-bit” option. Secondly, the objects in the image must have a higher gray value (i.e. they must be brighter) than the surrounding background. If this is not the case, the image can be inverted to correct this. To invert the image, make sure the image window is selected, then go to the “Edit” menu and select “Invert”.

Using the Plugin

Step 1: Use the line selection tool to draw a rough and inexact line across an object of interest. The line must extend beyond the boundaries of the object of interest, but must not intersect other objects at the same time. Example:



Step 2: Run the plugin from the “Plugins” menu. The gradients along the line will be computed, and edges of the object that intersect the line will be determined. The line will then be fit so that its endpoints lie on these edges. Example:



Step 3: Take a measurement of the length of the fitted line, if desired.

Release Information

v1.0: Initial release of the plugin.