MIJ: Making Interoperability Between ImageJ and Matlab Possible

Daniel Sage\(^1\), Dimiter Prodanov\(^2\), Jean-Yves Tinevez\(^3\), Johannes Schindelin\(^4\)

\(^1\)Biomedical Imaging Group, EPFL, Lausanne, Switzerland; \(^2\)Bio-Nano Electronics Department, Imec, Leuven, Belgium; \(^3\)Institut Pasteur, Imagopole / PFID, Paris, France; \(^4\)LOCI, University of Wisconsin-Madison, USA.

Abstract

We present a software module MIJ\(^*\) that allows to combine the powerful numerical computation of Matlab\(^1\) and the image-analysis capabilities of Image\(^2\). Since both ImageJ\(^3\) and Matlab\(^4\) run on multiple operating systems, MIJ is platform-independent. MIJ uses the Java Virtual Machine (JVM) to interact with the Matlab\(^4\) core. MIJ starts Matlab\(^4\) by calling a method from the Matlab\(^4\) launcher. In this way we can execute Matlab\(^4\) methods from the ImageJ\(^3\) console or scripts. In such way, we could bring the richness of ImageJ\(^3\) and of their plugins to the Matlab\(^4\) world.

MIJ contains a collection of static methods to exchange data bidirectionally between Matlab\(^4\) and ImageJ\(^3\). MIJ tries to avoid loss of accuracy by converting the image data to the closest numerical type of Matlab\(^4\). The only drawback of this interoperability model is the current limitation of the heap memory in Matlab\(^4\). Depending on the architecture and of the version of Matlab\(^4\), the heap space can be increased, but it cannot be changed by MIJ.

Since 2011, MIJ is included in the Fiji\(^5\) distribution of ImageJ\(^3\) together with a friendly startup Matlab\(^4\) script called Miji\(^6\). After calling a single function from Matlab\(^4\), all the functionality of all the plugins and libraries included in Fiji can be accessed by Matlab\(^4\) scripts, via Miji\(^6\). Most notably, Fiji\(^5\) 3D Viewer can be used to display three- and four-dimensional data in volume-rendering, iso-surface and orthoslices.

Features

MIJ is an Open-source Java class facilitating interoperability between Matlab\(^4\) and ImageJ\(^3\). Fiji\(^5\):

- Run ImageJ in the JVM of Matlab\(^4\);
- Export/import a 3D Matlab\(^4\) variable as ImageJ\(^3\) image: the image can be displayed or not;
- Export/import a 3D Matlab\(^4\) variable as ImageJ\(^3\) stack: the image stack can be displayed or not;
- ImageJPlus\(^7\) references can be converted to Matlab\(^4\) matrices for further processings using the Java interpreter of Matlab\(^4\);
- Enables to run plugins (setup path);
- Enables to run scripts (setup path);
- Additional information from ImageJ\(^3\) can be transferred to Matlab\(^4\): - histogram; - table of results; - column of results; - region-of-interests.

Application Cases

Using fast Matlab routines and display results on ImageJ

Reading image files using Bio-Formats on Matlab

Image-analysis on ImageJ, statistics on Matlab

Interoperability

Matlab Statistics analysis Optimization method
Fast matrix (image) operation
Large base of educators engineers
Broad scientific community
Interpreted language, prototype
ImageJ Image analysis User interface. Image interaction
Advanced research plugins
Large base of user (biologist, ...)
Biologist and developer community
Open-source, reproducible

How to use

Download

miJ.jar
installed

installation

Copy files into Matlab
(Java dir)
automatic

setup

Add miJ.jar and ij.jar in the
Java class path
Set the path

Run

MIJ

Reference

[8] TrackMate: http://fiji.sc/TrackMate

Technical Issues

The data has to be converted, copied and often transposed which requires a overhead time of computation. On a middle-range machine it takes around 100 ms to transfer 10 Mb.

Memory limitation

Matlab provides a Java Virtual Machine (JVM) which interprets the command of the console. In 2012, the version of the JVM is 1.6. The main drawback is that Matlab has limited the Java heap memory (e.g. 256 Mb on Mac OSX, Matlab R2010_b). This limitation restricts the usage of Java for huge datasets like multidimensional imaging purposes. For some operating systems, Matlab proposes to change the limitation by editing a “hidden” file java.opts: -Xms1000m.