Plugins

http://mipav.cit.nih.gov
MIPAV Team

Matthew McAuliffe
William Gandler
Ruida Cheng
Evan McCreedy
Justin Senseney

Contractors
Nish Pandya, SSAI
Alexandra Bokinsky, Geometric Tools Inc.
Medical Image Processing, Analysis & Visualization
&
Plugins

Justin Senseney

SenseneyJ@mail.nih.gov
Biomedical Imaging Research Services Section (BIRSS)
Imaging Sciences Laboratory
Division of Computational Bioscience
Center for Information Technology
(301) 594-5887
http://mipav.cit.nih.gov
Options exist
How to manipulate
Plugins

- PlugInAlgorithm – Develop new functionality and ability to call functions already in MIPAV.

- PlugInFile – Develop files readers to support unique file formats.

- PlugInGeneric – Plugins that do not require an open image.

- PlugInView – Develop new visualizations of datasets.

- PlugIn – ImageJ plugin.
Plugins

- To build a plugin, three files are typically used:
  - PlugInTest.class
    - Interface between MIPAV and plugin.
  - PlugInDialogTest.class
    - Creates dialog for input.
  - PlugInAlgorithmTest.class
    - The actual algorithm.
Plugins - Location

- Stored for each user:
  - Windows   - C:\Users\{{user_name}}\mipav\plugins
  - Linux/Mac OS   - /home/\{{user_name}\}/mipav/plugins
Plugins - Installing

This interface allows for batch installation of plugins into MIPAV. You may select Java class files or container files. These include *.class, *.jar, *.zip, *.jar files.

Select a plugin directory

Select class files

Selected class files

Install Plugin(s) 
Close
Plugins – Installing

- Installing a plugin (Plugins -> Install Plugin)

Copies files:
- .class
- .jar
- .zip

into user directory.

HINT:
If something goes wrong, just copy the files yourself using locations on Slide 8.
Plugins – Installing (2)
Plugins - Executing

- Executing a plugin named 'NewGeneric1'.
- A dialog box allows selecting options such as 'Do gaussian blur'.
- Two images showing before and after processing with the plugin.
Plugins - Scripting

Execute plugin

`PluginDialogNewGeneric2("input_image_1 ext_image Simage1", "do_gaussian boolean true")`
Script Running Review
Uninstall Plugin

Uninstall Plugin(s)

Select plugins to uninstall

- NewGeneric2

Information

Plugin uninstall results: (1 deleted, 0 failed)
PluginNewGeneric2 was successfully deleted.
Plugin Lab

• Install
  • PlugInCT_MD – already in MIPAV
  • MuscleSegmentation –
    senseneyj/code/bioformats.html
  • ImageJ –
Plugins from command line

- Mipav –p Plugin_Name

- Can be run headless
Break - Review

- Plugins
  - File
  - Generic
  - Algorithm
  - View
- Stored in user folder
- Install, Run, Uninstall
Development environment

- Textpad: http://www.textpad.com/
  - Simple, few linked files
  - Want to learn Java
- Eclipse: http://www.eclipse.org/
  - Many files, libraries
  - Source repository
  - Want to use Java
Download Eclipse

- Download here:
  http://www.eclipse.org/downloads/
“Install” Eclipse
Fix Eclipse

To specify memory, you can add the following arguments to the Eclipse launch command:

```
-startup
plugins/org.eclipse.equinox.launcher_1.1.1.R36x_v20101122_1400.jar
--launcher.library
plugins/org.eclipse.equinox.launcher.win32.win32.x86_64_1.1.2.R36x_v20101222
-product
org.eclipse.epp.package.java.product
--launcher.defaultAction
openFile
--launcher.XXXMaxPermSize
256M
-showsplash
org.eclipse.platform
--launcher.XXXMaxPermSize
256M
--launcher.defaultAction
openFile
-vmargs
-Dosgi.requiredJavaVersion=1.5
-Xms512m
-Xmx2048m
```
Start Eclipse
Java Programs

- Virtual machine = platform independent
- JRE to execute
JRE Setting
Add JRE
Locate JRE
Locate JRE

![Image of Locate JRE dialog box]

**JRE Definition**
Specify attributes for a JRE

- **JRE home:** C:\Program Files\mipav531\jre
- **JRE name:** mipav531
- **Default VM Arguments:**
- **JRE system libraries:**
  - C:\Program Files\mipav531\jre\lib\extj3dcore.jar
  - C:\Program Files\mipav531\jre\lib\extj3dutils.jar
  - C:\Program Files\mipav531\jre\lib\extjhf5.jar
  - C:\Program Files\mipav531\jre\lib\extjhf5obj.jar
  - C:\Program Files\mipav531\jre\lib\extjmf.jar
  - C:\Program Files\mipav531\jre\lib\extjocl.jar
  - C:\Program Files\mipav531\jre\lib\extjogl.all.jar
  - C:\Program Files\mipav531\jre\lib\extlocaledata.jar
  - C:\Program Files\mipav531\jre\lib\extmediaplayer.jar

- **Add External JARs...**
- **Javadoc Location...**
- **Source Attachment...**
- **Remove**
- **Up**
- **Down**
- **Restore Default**
Eclipse project

• Mipav
  • Java project
  • Contains plugins
Make Mipav Project
Make Mipav Project

Specify JRE
Mipav Program

• Execution environment needs to be replicated
Execution environment

- Classpath
  - Libraries - jars
  - Associated class files
- Memory – used by VM
- Launching class file
Jars

- Libraries
  - PDFbox
  - JPEG2000
  - ImageJ
- Help files
- Insight Toolkit wrappers
Add Jars
More Jars
More Jars: Help
More Jars: ITK
Class files

- Object-oriented
- Executable programs
- In Mipav:
  - Read/write images
  - Apply filters
  - Run plugins!
External class folder

Select the correct version of Mipav
Click Finish!

Note class folder and JRE selection
MipavMain
Run Configuration
New Configuration

Create, manage, and run configurations

Configure launch settings from this dialog:

- Press the 'New' button to create a configuration of the selected type.
- Press the 'Duplicate' button to copy the selected configuration.
- Press the 'Delete' button to remove the selected configuration.
- Press the 'Filter' button to configure filtering options.
- Edit or view an existing configuration by selecting it.

Configure launch perspective settings from the Perspectives preference page.
Select Main Type
Java Memory

-Xms (min), -Xmx (max)
Memory

Create, manage, and run configurations
Run a Java application

Name: mipav

Program arguments:

VM arguments:
-Xms512m -Xmx5120m

Working directory:
Default: $\{workspace_loc:mipav\}

Filter matched 5 of 5 items

Apply  Revert  Run  Close
Run Mipav

- MipavMain is executed
  - Classpath is set
  - Memory allocated
  - Libraries are referenced
Mipav Launched

Version number

VM Memory
JVM Info

JRE in mipav

Plugin folder in workspace
Java Compiler

- Source code:
  - *.java files
  - Text-editor readable

- Platform independent byte-code:
  - *.class files
Get NewGeneric2

  - PlugInNewGeneric2.java
  - PlugInDialogNewGeneric2.java
  - PlugInAlgorithmNewGeneric2.java
Import NewGeneric2

- In Eclipse: Press File->Import
Compile NewGeneric2

- This is done for you, incrementally in Eclipse
  - Errors and warnings update with each change
Install NewGeneric2

- Only install PlugInNewGeneric2.class

Why did this work?
Run NewGeneric2

- Open an image, try it out

- Install Plugin
- Uninstall Plugin
- Compile and run...

- A titled border
  - Blank text
  - Do gaussian blur

- Memory: 321M / 9583M
Look at NewGeneric2

- Open Dialog, try to change the title “Generic Plugin”
Change NewGeneric2

Location:

```java
/**
 * Used in turning your plugin into a script
 */
protected void storeParamsFromGUI() throws ParserException {
    scriptParameters.storeInputImage(image);
    scriptParameters.getParams().put(ParameterFactory.
    //end storeParamsFromGUI()

private void init() {
    setForeground(Color.black);
    setTitle("Justin's Greatest Plugin");
    try {
        setIconImage(MipavUtil.getIconImage("divin
```
Run NewGeneric2

• See change:
NewGeneric Algorithm

- Look at algorithm construction, execution

```java
fireProgressStateChanged("Message 2D: "+srcImage.getImageName());

if(dcoGaussian) {
    final float[] sigmas = {1.0f, 1.0f};

    AlgorithmGaussianBlur gaussianBlurAlgo = new AlgorithmGaussianBlur(destImage, srcImage, gaussianBlurAlgo.setRunningInSeparateThread(false);
    linkProgressToAlgorithm(gaussianBlurAlgo);
    gaussianBlurAlgo.runAlgorithm();
}

for(int i=1; i<100; i++) {
    fireProgressStateChanged(i);
}
```
CTRL-click on run algorithm, now try hovering:
Plugins - MIPAV API

Download from:
http://mipav.cit.nih.gov/documentation/api.zip
Plugins - MIPAV API

Attach here:

- Java Source Attachment
- Javadoc Location
- Native Library

Javadoc Location

Specify the location (URL) of the documentation generated by Javadoc. The Javadoc location will contain a file called 'package-list'.

- Javadoc URL (e.g. http://www.sample-url.org/doc/ or file://c:\myworkspace/mypack/topic/doc)

Javadoc location path: [Browse...]

- Javadoc in archive
  - External file
  - Workspace file

Archive path: C:\Users\senseneji\Downloads\api.zip [Browse...]
Path within archive: [Browse...]

[Validate...] [Apply] [Cancel]
Plugins - MIPAV API

View here:
Modify Plugin Lab

- Edit PlugInNewGeneric2:
  - Change text in plugin dialog.
  - Print out when algorithm has completed.
  - Increase kernel size.
  - Change 3D to 2.5D processing.
  - Add check box to allow user to select either 3D or 2.5D processing.
Plugins - Documentation

- Plugin development documentation

- This presentation is posted at:
    - Will also be posted here:
      http://mipav.cit.nih.gov/documentation/presentations/

- Also see this wiki page:
  http://mipav.cit.nih.gov/pubwiki/index.php/Writing_MIPAV_plugins_without_MIPAV_source
Summary

• Plugins
  - Unique functionality using MIPAV API
• Develop in Eclipse
• Use MIPAV API