Contents

System Requirements

Operating System

This codebase can be setup on Windows or MacOS.

Library

This was developed using Java 1.8.0_231. To re-build this codebase you should have same version of JDK installed. If you do not have it, you may get it from Oracle's website and install it.

Tools

You would need following tools to work with codebase.

1. Eclipse
2. GIT
3. Eclipse-GIT Extension (Optional)
4. Source Tree (Optional)

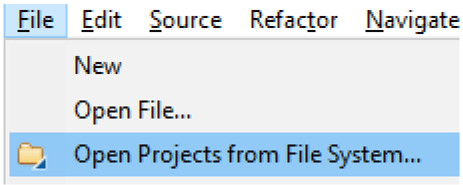
Cloning Repository

You may close the repository from following URL to your system

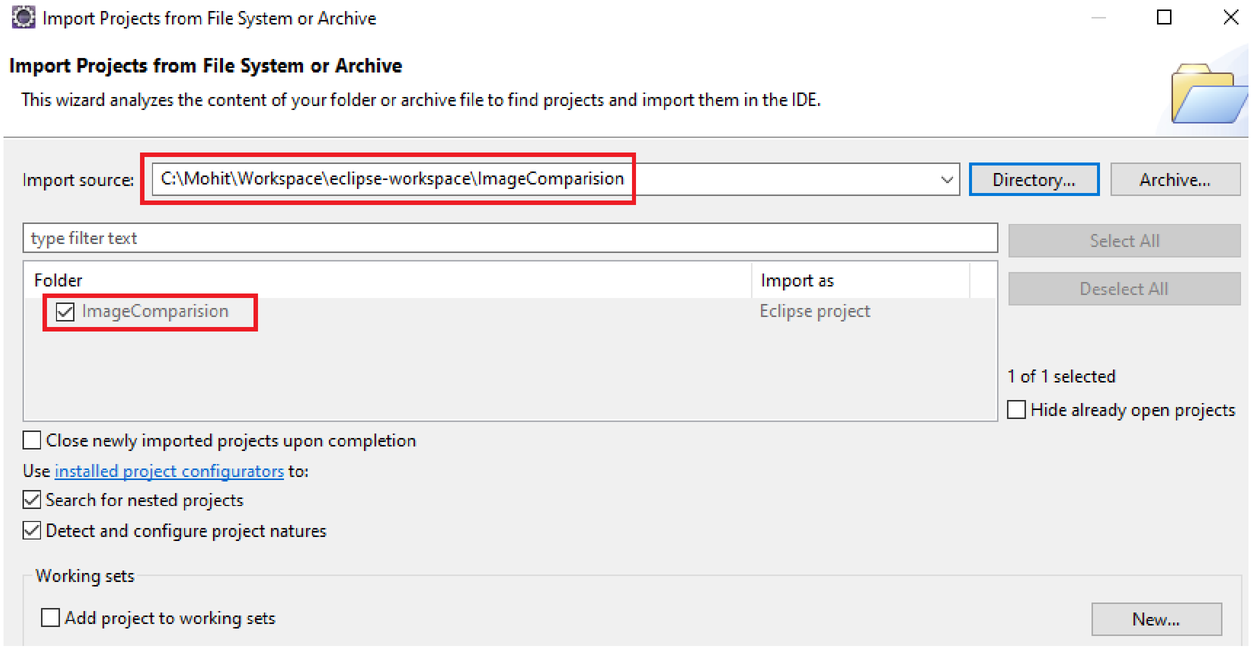
https://github.com/mkthakral/ImageComparisionTool.git

Import to Eclipse

Once you have the codebase on your system, you can import it into eclipse.



Select the codebase path and click "Finish"



Import Projects from File System or Archive

Import Projects from File System or Archive

This wizard analyzes the content of your folder or archive file to find projects and import them in the IDE.

Import source: C:\Mohit\Workspace\eclipse-workspace\ImageComparision

Folder	Import as
☑ ImageComparision	Eclipse project

1 of 1 selected

□ Hide already open projects

□ Close newly imported projects upon completion

Use installed project configurators to:

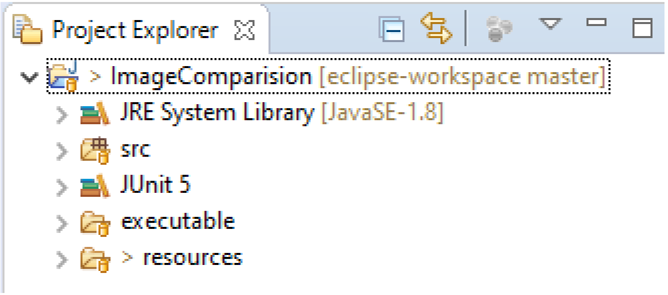
☑ Search for nested projects

☑ Detect and configure project natures

Working sets

□ Add project to working sets

Once import is complete, you should see the project structure like this.



Project Explorer
> ImageComparision [eclipse-workspace master]
JRE System Library [JavaSE-1.8]
src
JUnit 5
executable
> resources

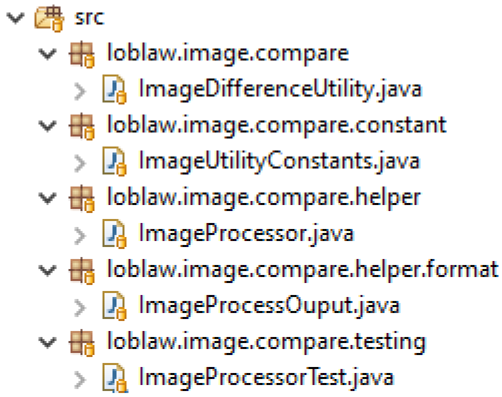
Directory Structure

Directory	Purpose
<Project Home>/src	This has all the code needed to build or run the application. This also has the code for unit testing.
<Project Home>/executable	Here you will find the application’s built ready to use versions.
<Project Home>/executable/Version-N	Here you will find executable JAR i.e. the built application for specific version. It also has “changelog.txt” which lists the changes done in this version.
<Project Home>/resources	This directory contains documentation of the project and resources needed to run unit tests.

Understanding Code

Files

Following is the code file structure.



```
src
  loblaw.image.compare
    ImageDifferenceUtility.java
  loblaw.image.compare.constant
    ImageUtilityConstants.java
  loblaw.image.compare.helper
    ImageProcessor.java
  loblaw.image.compare.helper.format
    ImageProcessOuput.java
  loblaw.image.compare.testing
    ImageProcessorTest.java
```

Please find high level function of each file.

File	Purpose
ImageDifferenceUtility.java	Entry point of the application. Creates a user form i.e. GUI. Reaches out to “ImageDifferenceUtility.java” for processing input.
ImageUtilityConstants.java	Contains constant variables used in project, also contains a bit of configurable items like error messages.
ImageProcessor.java	This is the primary working horse of the application which contains algorithm to compare images, reading input, creating output file.
ImageProcessOuput.java	This is a custom class, a format to return response from “ImageProcessor.java” to “ImageDifferenceUtility.java”
ImageProcessorTest.java	This has multiple unit test cases built using Junit.

Algorithm

This section explains the algorithm that’s used in this utility to compare images.

Here are the steps that algorithm executes:

1. Check if image dimensions of both input images match

2. If the images dimensions match, we get their height and width and go through each pixel
3. From each pixel value we get value of RGB i.e. Red, Green and Blue Colors
4. We add all the difference of colors and finally get the average by dividing it from Total number of pixels on image i.e. image height * image width * 3 (3 for RGB)
5. To get the percentage of difference between two images, we apply this formula (average/255)*100 as there are total 255 pixels

Example:

Input:

Image 1: 752x750

Image 2: 752x750

Total Pixel = 752 x 750 x 3 = 1692000

Total RGB Difference after going through each pixel’s RGB = 39592599

Average Pixel Difference = 39592599/1692000 = 23.399881205673758 = 23.39

Percentage Difference between two images = (23.39/255) * 100 = 9.17%

Modifying Utility

Code Merge

To modify code for any new requirements, you may cut a branch from master branch in GIT repository and make changes. Once you are done with the changes you may get this code merged with master branch.

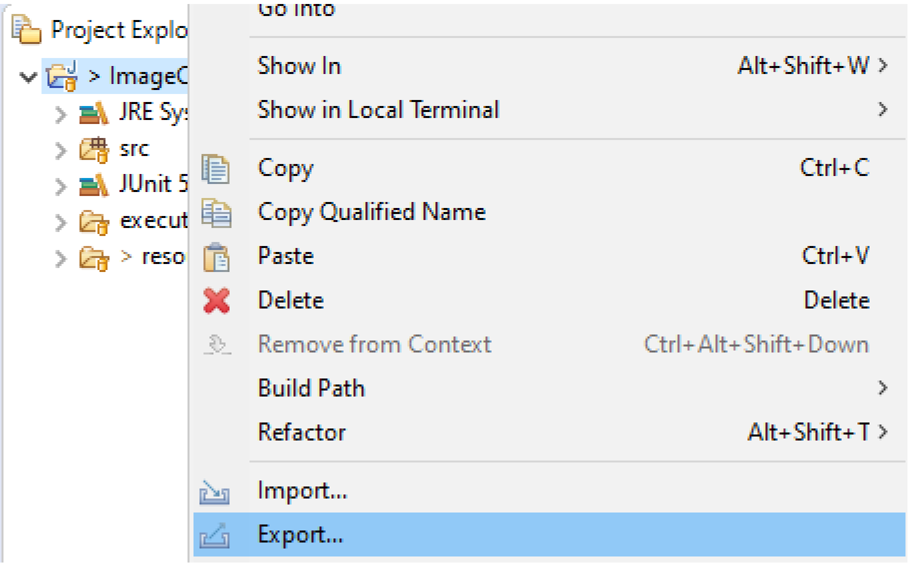
Directory Structure

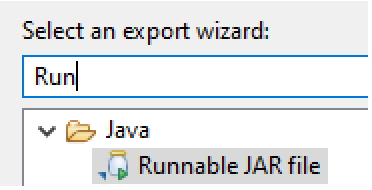
You should create also create following manually.

Create	Purpose
<Project Home>/executable/Version-N	Create this directory structure to host a new version of executable file.
<Project Home>/executable/Version-N/changelog.txt	Create this file and mention what changes are done in this version of application.

Building Executable

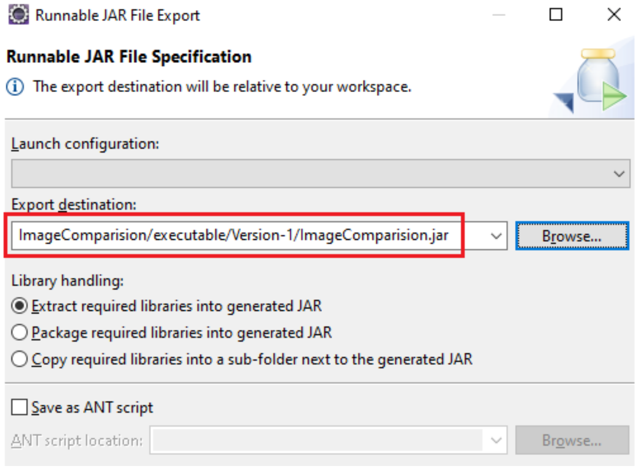
When you are modify the code, you should create a new version of executable JAR. For this




Select an export wizard:

Run

Select the directory created above and mention file name.


Runnable JAR File Export

Runnable JAR File Specification

The export destination will be relative to your workspace.

Launch configuration:

Export destination: ImageComparision/executable/Version-1/ImageComparision.jar

Library handling:

● Extract required libraries into generated JAR

○ Package required libraries into generated JAR

○ Copy required libraries into a sub-folder next to the generated JAR

□ Save as ANT script

ANT script location:

This shall create the JAR file at the selected directory.

Check in this directory into GIT.

Test Cases

We have written several unit test cases when you modify the code. Do not forget to review/add/modify test cases according to changes done.

Version

Update version information at `ImageUtilityConstants.java -> VERSION`