ImageJ plugin VA_gtmBrush "Ground-Truth-Marking Brush"

Reference link: https://github.com/jbstphr/ImageJ_GroundTrust_Brush

1. The introduction

The ImageJ-plugin creates/edits coupled *ground_mark* as"binary" file ".**\xxx_gt.bmp**" for an image-file "**.\xxx.*ext***." (where"***ext***" could be "***bmp***" etc).

1.1. The plugin's files

The source:
`VA_gtmBrush.java`

Compiled java-code:
```
VA_gtmBrush$Options$1.class
VA_gtmBrush$Options$2.class
VA_gtmBrush$Options$3.class
VA_gtmBrush$Options$4.class
VA_gtmBrush$Options$5.class
VA_gtmBrush$Options.class
VA_gtmBrush.class
```

To install the compiled Java-code

- Create new ImageJ plugins' sub-directory like `C:\Program Files\ImageJ\Plugins\MyPlugMenu` (or use some of existing ones).
- Copy the files `*.class` there
- Start ImageJ; the menu "Plugins" now has sub-menu-item "*MyPlugMenu*" and command-item "*VA_gtmBrush*"
- Click command "*VA_gtmBrush*"; the glyph of running plugin should appear in the toolbar.

To build from source (if *Java development tools* are installed and visible for ImageJ):

- Copy `VA_gtmBrush.java` to some ImageJ plugins' sub-directory.
- In running ImageJ click "*Plugins|Compile and Run*" and browse for `VA_gtmBrush.java`; success compilation generates the files `*.class` and the glyph of running plugin appears in the toolbar.

1.2. The data-files

An image-file `Dir\Name.ext` (any 2D-format supported by ImageJ; e.g. "`Name.bmp`") might have *coupled overlay* `Dir\GT\Name.bmp` (where BMP has one gray-scale-channel with possible pixel values 0 or 255).

1.3. The functionality

The plugin operates with *coupled overlays* (it does not touch original images) using one of two possible (at given moment) *Regions of interest*:

- *ImageRoi* (2D arrays of pixels) with opacity=1..10.
- *ShapeRoi* (boundary curves with opacity=0 for internal regions).

ImageRoi is default modus operandi (actually a BMP is an ImageRoi). Conversion *ImageRoi* =>*ShapeRoi* uses semi-standard ImageJ-command "*Set selection*" for binary images, which requires powerful PC and might have different implementations.

However *ShapeRoi* (after initial conversion) could be is easier for human-assisted operations and has simpler JAVA-code for maintain edit.

2. User interface

Mouse click/drag marks area of current brush in *coupled overlay* (or un-marks one when key ALT is pressed).

2.1. Popup menu

The dialog appears by right mouse-click only when there is no open image and there is no "Option control dialog". Preset the *opacity*:

0 - *ShapeRoi*.
1..10 - *ImageRoi*.

Initialization profile (object *Prefs*) records only positive values (so initial modus operandi is *ImageRoi*).

2.2. Options control dialog

The dialog appears by double mouse-click (and by right mouse-click when there is/are open images).

[Reload] – load *coupled overlay*.
[Save] – save *coupled overlay*.
[Delete] – delete *coupled overlay* (it is disabled when there is no respective file).
[<<] (or [>>]) – open previous image (or next one respectively) and its *coupled overlay*.
Note: standard imageJ-command "Open Next" (key shift+ctrl+O ; also *Open previous* by alt+shift+ctrl+O) doesn't open overlay. However that maintains the *ShapeRoi* and automatically clears *ImageRoi*.

[Auto-save] – trigger [Save] by [<<] or [>>] if user *drags* mouse at least once (mouse-click does not trigger auto-save to avoid occasional empty *coupled overlays*).
[Width] – brush-width in pixels (1..100).
[Opacity] – 1..10 (visible only for *ImageRoi*).

2.3. *Sample operational flow*

- Start ImageJ.
- Click menu “Plugins|MyPlugMenu|VA_gtmBrush (the glyph of running plugin appears in the toolbar).
- Click (on the glyph) right mouse button and preset “opacity” (refer 2.1).
- Double click (on the glyph) left mouse button and the Options control dialog (refer 2.2) appears.
- Click menu “File|Open” (*or press ctrl+O*) and browse for image-file.
- Edit *coupled overlays* `.\GT\*.bmp` as required.

3. Source code

ImageRoi is based on source code of standard ImageJ brush-plugin (with changed objects' ownership to make *ShapeRoi also possible*). It is active when *VA_gtmBrush ::SHAPE_RO I=false*.

ImageRoi is a *ROI* of an *Overlay* attached to *ImagePlus*. Event *imageUpdated(ImagePlus)* destroys the overlay.

ShapeRoi is based on several publications in ImageJ-web-conferences for usage of standard ROI-selections. It is active when *VA_gtmBrush ::SHAPE_RO I=true*.

ShapeRoi is instance of *ImageProcessor* 's *Roi* in *ImagePlus*. Event *imageUpdated(ImagePlus)* does not touch that standard *ROI*.