PlotInteractionExercise

March 13, 2017

1 simple plot of a 2D image

- using Plot2D

1.1 load data from data/lena.hdf5

```
In [1]: import h5py
        import numpy
        dataPath='data/ascent.h5'
        f=h5py.File(dataPath)
        image=numpy.array(f['data'], dtype='float32')
```

```
In [2]: from silx.io.utils import h5ls
        h5ls(dataPath)
```

```
Out[2]: '<HDF5 dataset "data": shape (512, 512), type "<i8">\n'
```

1.2 plot the lena image

- using silx.gui.plot.Plot2D.addImage()

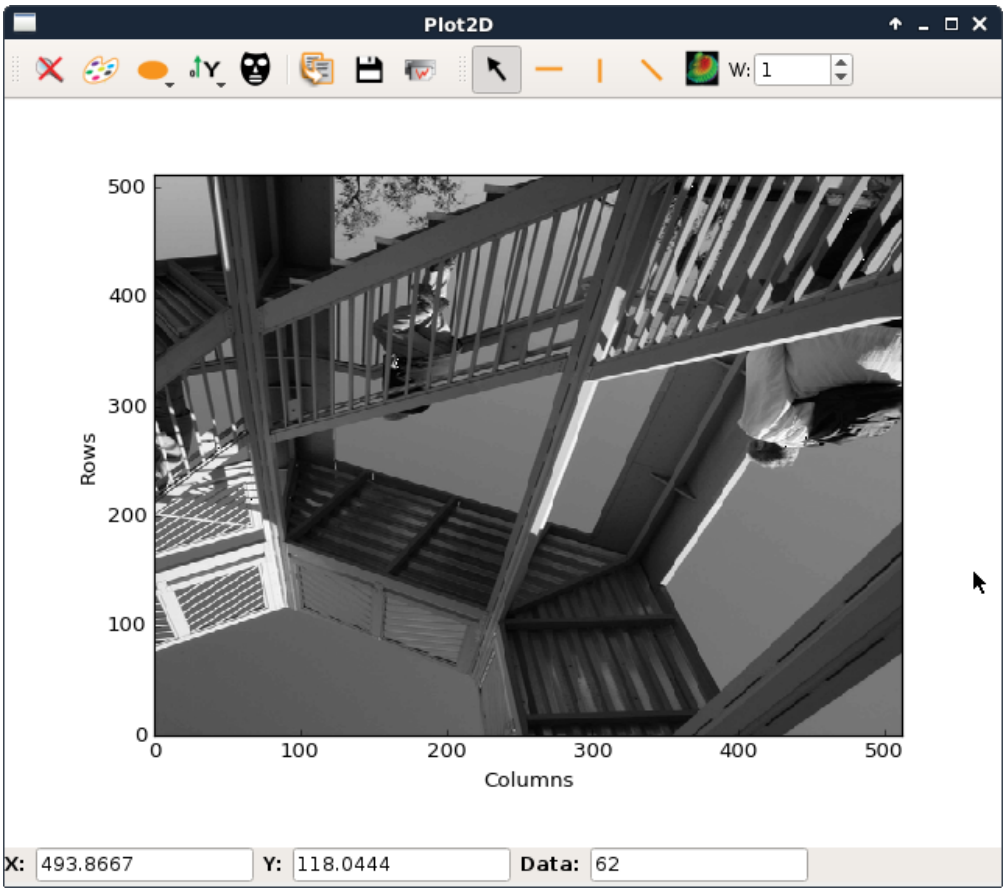
```
In [ ]: ...
```

2 display the pixel intensity distribution

2.1 create the histogramnd

- using silx.math.histogram.Histogramnd
- http://www.silx.org/doc/silx/dev/modules/math/histogram.html

```
In [ ]: from silx.math.histogram import Histogramnd
        histo, w_histo, edges = Histogramnd(image.flatten(),
                                            n_bins=256,
                                            histo_range=[0,256])
```



ascent image

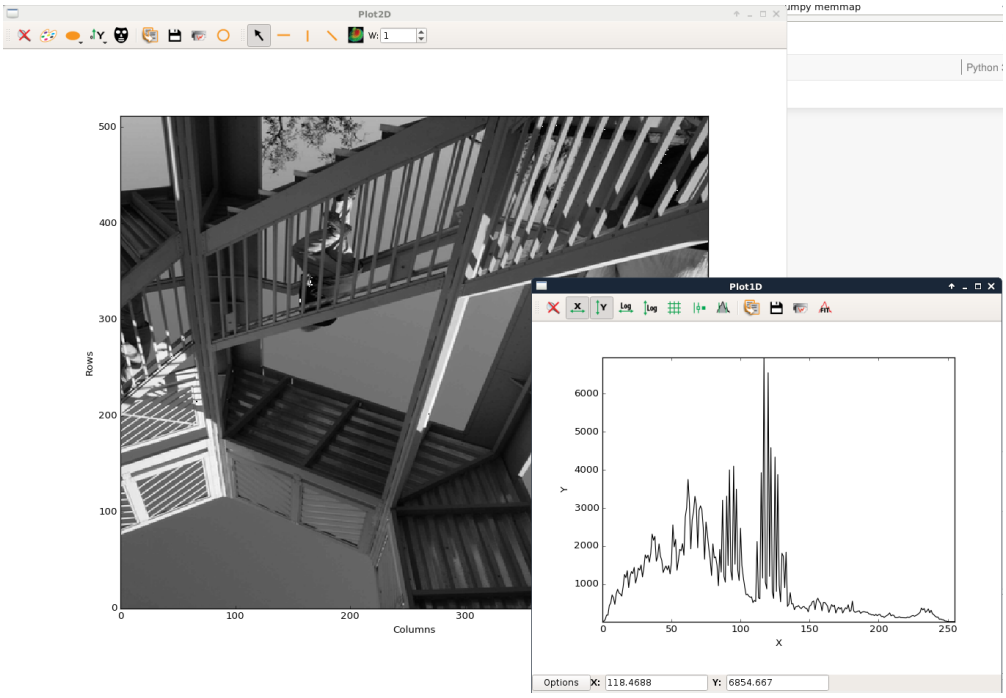
2.2 plot the histogram

- using silx.gui.plot.Plot1d

```
In [ ]: ...
```

3 create a PlotAction which plot the histogram for the current image

- using silx.gui.plot.PlotActions.PlotAction
- doc@ http://www.silx.org/doc/silx/dev/modules/gui/plot/plotactions_examples.html



lena image and pixels intensity

```
In [ ]: from silx.gui.plot.PlotActions import PlotAction
        from silx.math.histogram import Histogramnd
        from silx.gui.plot import Plot1D

        class ComputeHistogramAction(PlotAction):
            """Computes the intensity distribution on the current image

            :param plot: :class:`.PlotWidget` instance on which to operate
            :param parent: See :class:`QAction`
            """
            def __init__(self, plot, parent=None):
                PlotAction.__init__(...)
```

```
    def computeIntensityDistribution(self):
        """Get the active image and compute the image
        intensity distribution"""
        # By inheriting from PlotAction, we get access to attribute
        # self.plot
        # which is a reference to the PlotWindow
        ...
```

3.1 Add this action into the toolBar of the window

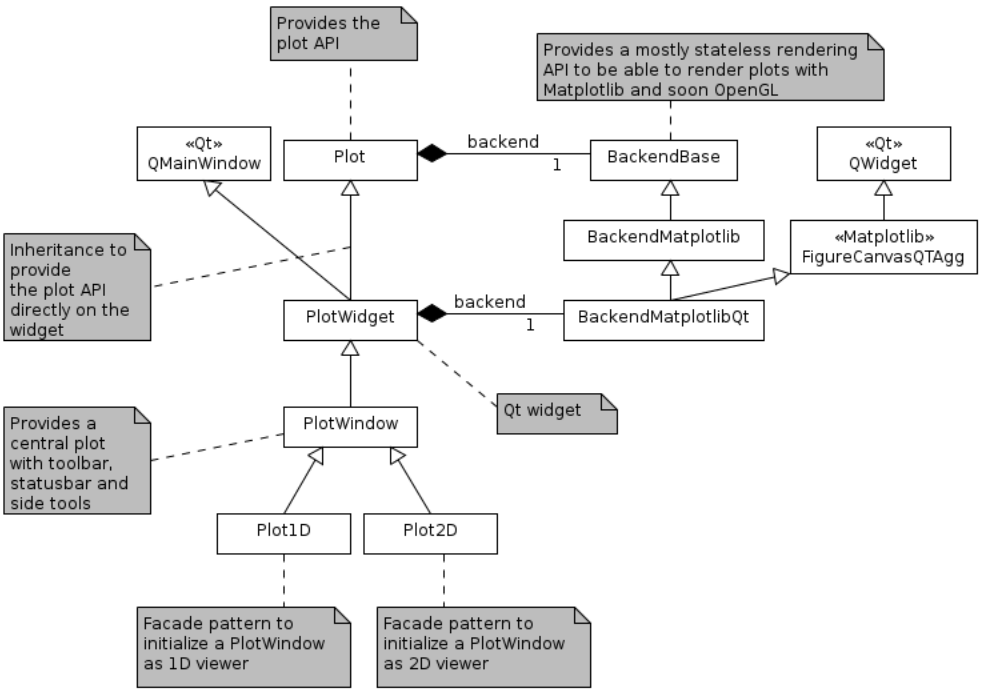
```
In [ ]: myaction=ComputeHistogramAction(plotImage)
        toolBar=plotImage.toolBar()
        ...
```

4 show automatically the histogram when the image change

- using plotImage.sigActiveImageChanged.connect(plotHisto)

```
In [ ]: ...
```

5 For information : the class diagram of the Plot module



plot class diagram

```
In [ ]:
```