

TUCPR06



Fast Interactive Python-based Analysis of Streamed Images

Andrei Sukhanov,

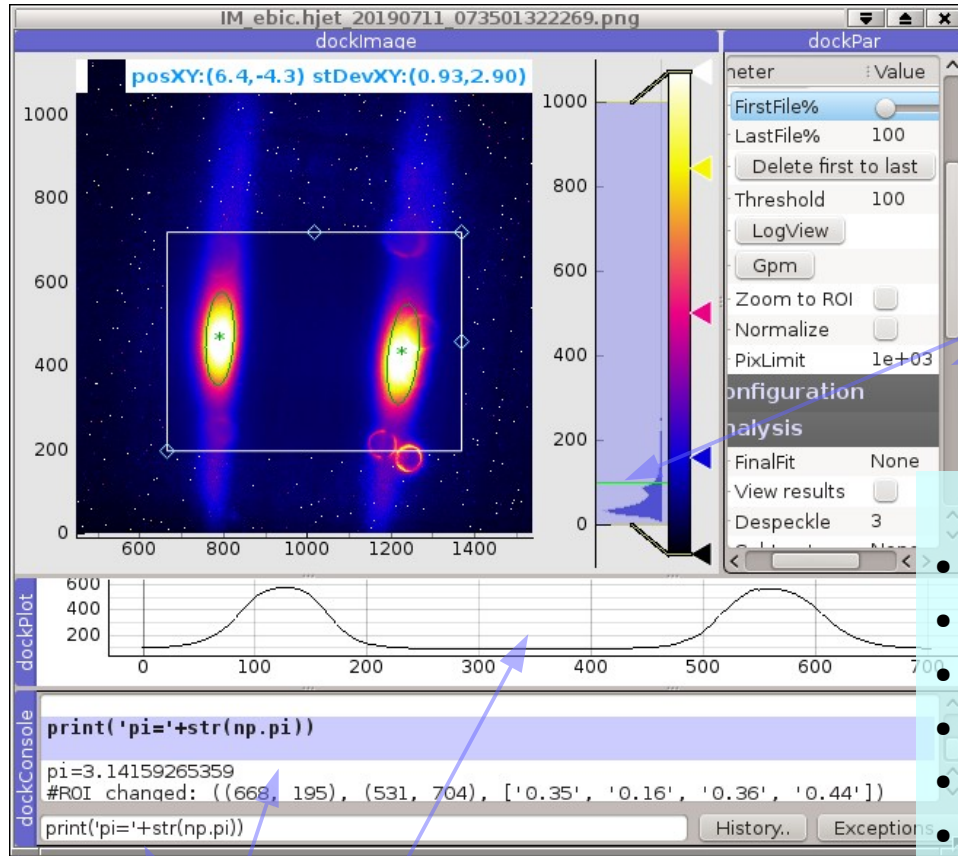
W. Fu, J.P. Jamilkowski, R.H. Olsen
Collider-Accelerator Department,
BNL, USA

Topics

- Motivation.
- Features.
- Built-in image analysis.
- File browsing and cleanup.
- Add-ons.
- Integration into Control Architecture.

- ImageViewer is a general purpose image analysis application, written in pure Python.
- Graphics by pyqtgraph (Qt). (Fast).
- Analysis by scipy.ndimage. (Reliable).
- Image streams (16+ bits/channel) from:
 - Camera servers (RHIC controls, EPICS, liteServer),
 - File systems,
 - Local USB camera,
 - HTTP.
- The default file format is PNG, other formats are supported as well.

Features



Single control window with adjustable panes

- Image pane
 - contrast/coloration control
 - color mapping
 - threshold slider, isocurves
- Control pane

Features

- Arbitrary rotation,
- Region Of Interest (ROI), ROI projection plots.
- ROI Partitioning,
- Reference images, pedestal suppression,
- Multi-frame averaging,
- More in following slides...

- ROI projection plot
- Message pane
- Interactive Python console

Calibration pixels/mm

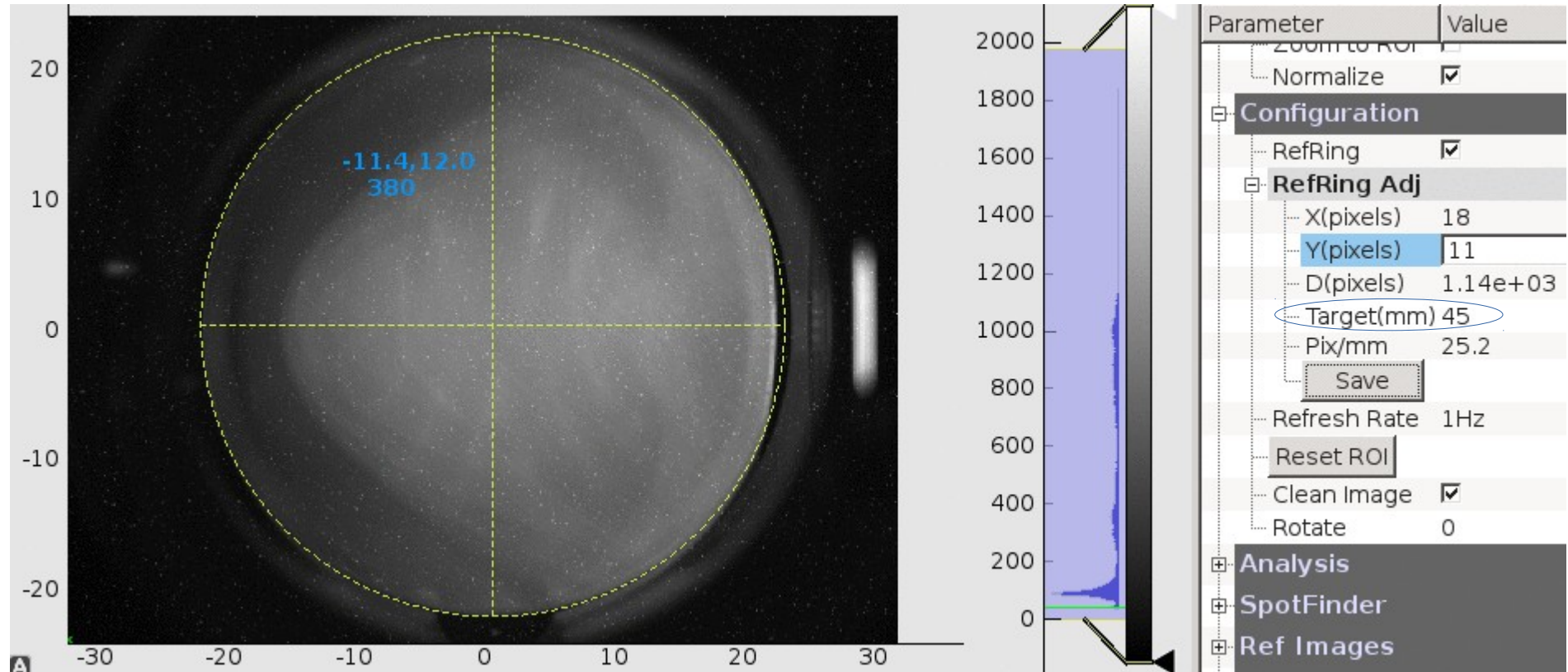
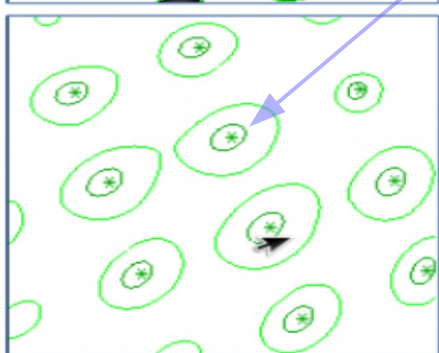
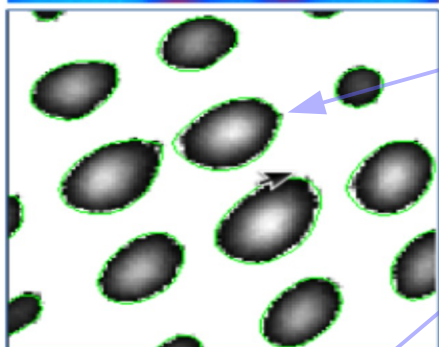
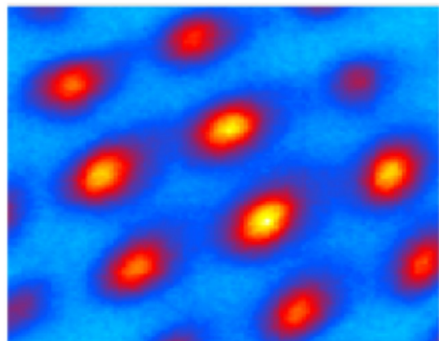


Image Analysis. Computer Vision

- Conversion of color (RGBA) to gray-scale
- Detection of connected objects using thresholding and labelling



* Parameters of fitted ellipses of found objects are calculated using Image Moment Statistics:

Center Of Gravity: $x = M_{10}/M_{00}$,
 $y = M_{01}/M_{00}$,

Half widths (RMS): M_{20} , M_{02} ,

Pearson Correlation Coefficient:

$$\rho = \frac{M_{11} - M_{10} M_{01}}{\sqrt{M_{20} - M_{10}^2} \sqrt{M_{02} - M_{01}^2}}$$

Tilt angle of the ellipse:

$$\theta = \text{atan}\left(\frac{2\rho \cdot M_{20} \cdot M_{02}}{M_{20}^2 - M_{02}^2}\right)$$

Image Analysis, Fitting.

The standard analysis provides fast and robust estimation of the object position.

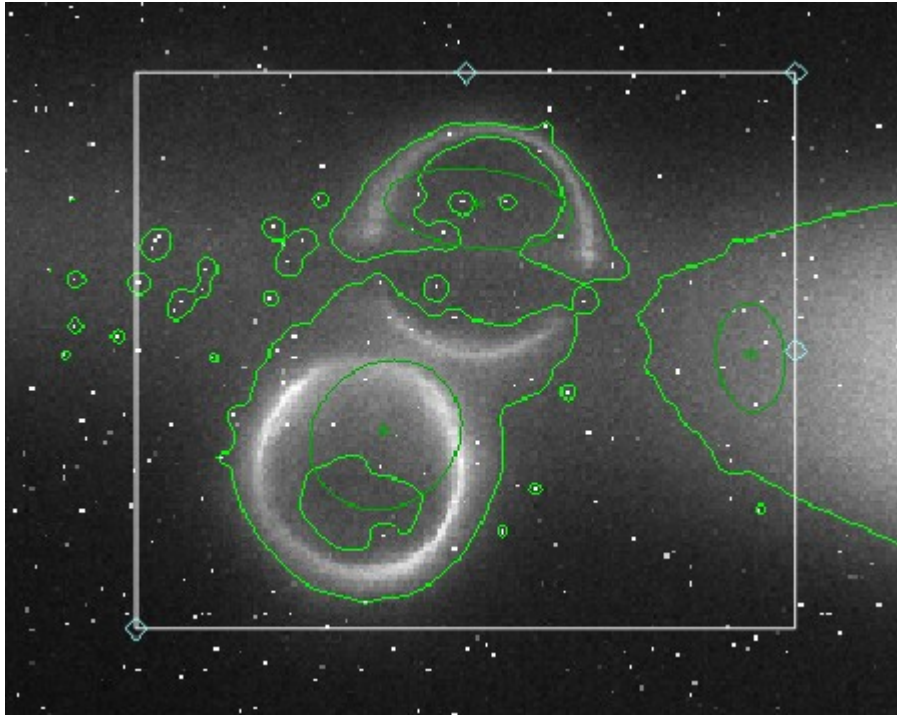
Calculated width of the object may be biased if background level is significant.

Parameter estimation could be improved using:

- 1) Gaussian fit of X and Y projections of the object
(reasonably fast).
- 2) Full 2D gaussian fit (slow).

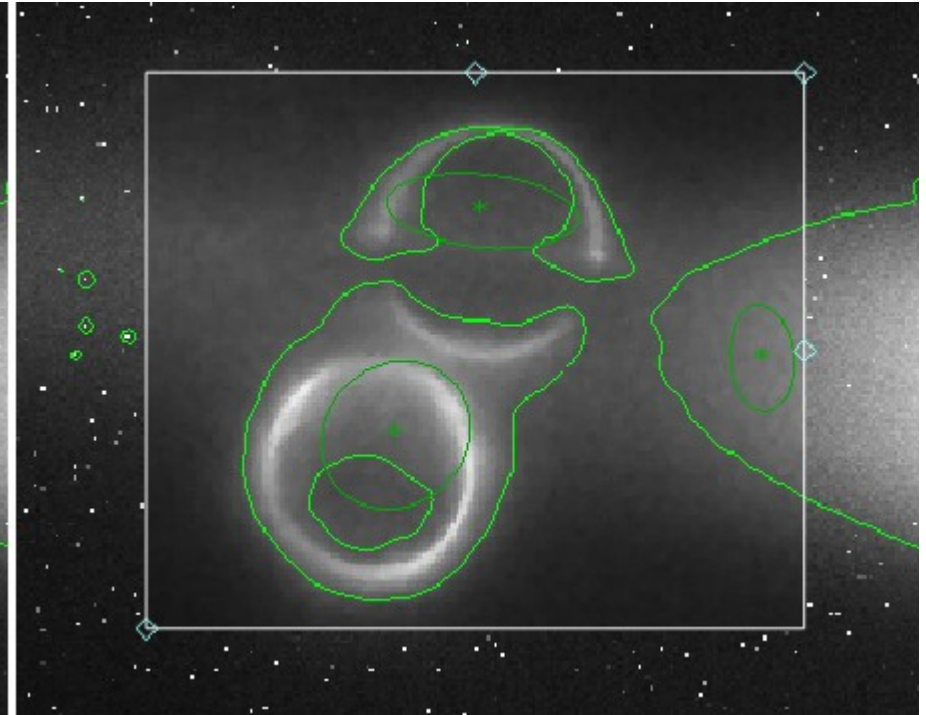
The fitting range is extended $6 * \text{RMS}$ to correct for pedestal.

De-speckling



Original image

Speckles are disconnected from objects.
They do not spoil the standard analysis.



De-speckling with median 3x3 filter

De-speckling is needed when fitting is involved.

File browsing and cleanup

Naming.

Name of the saved file contains timestamp with microsecond precision.

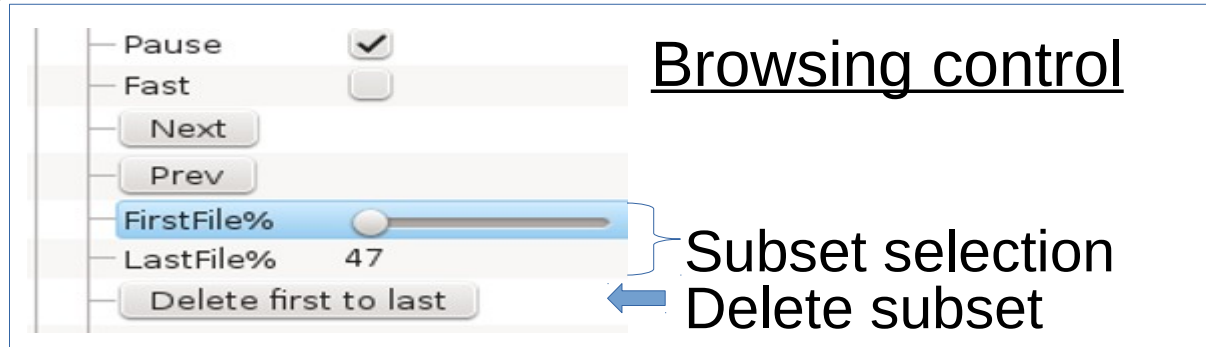
Command line:

`imageViewer.py -bfile /path/IV_camName_20190316_200259951187.png`

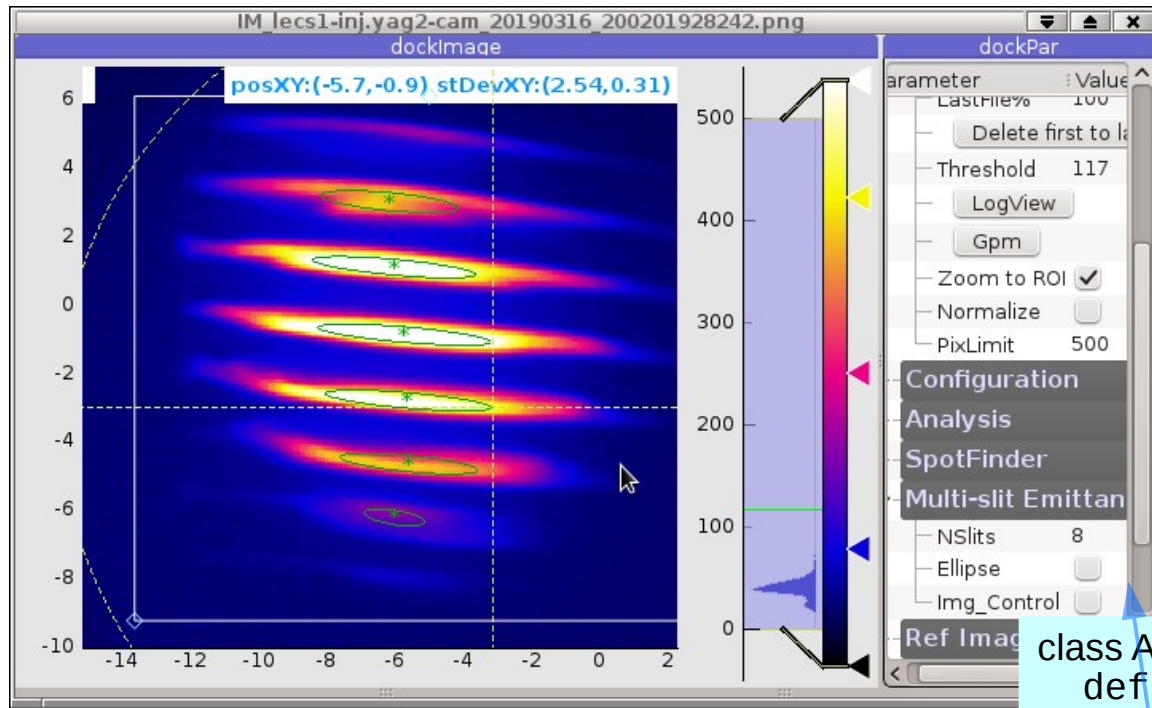
Hundreds of thousands of images could be saved per day.

Linux **ls** command may take several minutes to list a directory.

The **browsing control** speeds up the browsing and cleanup of Image directories.



Add-ons.

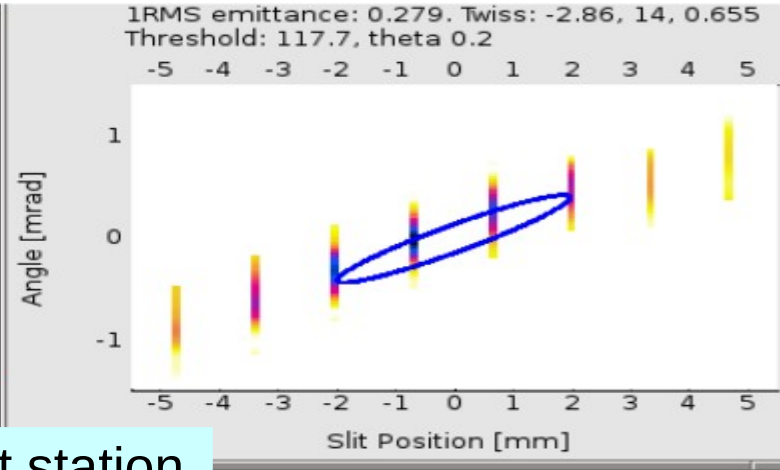
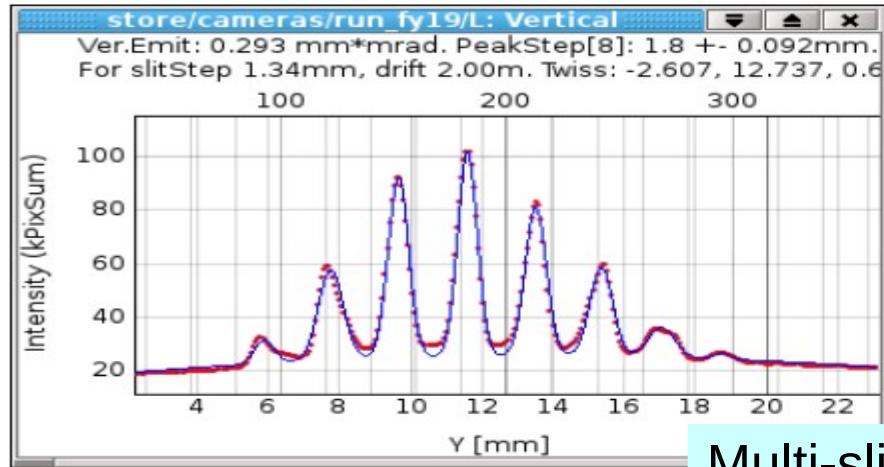


Add-on is a python file, which is specified in a command line argument and is imported during initialization. An add-on can instantiate (import) other add-ons dynamically.

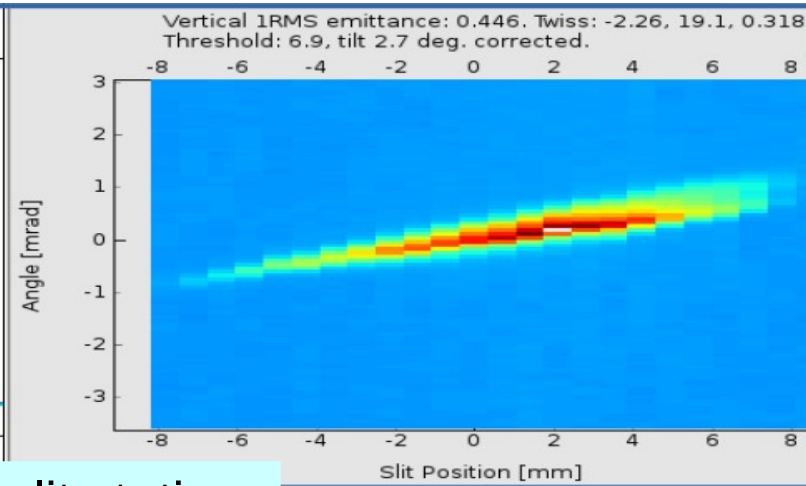
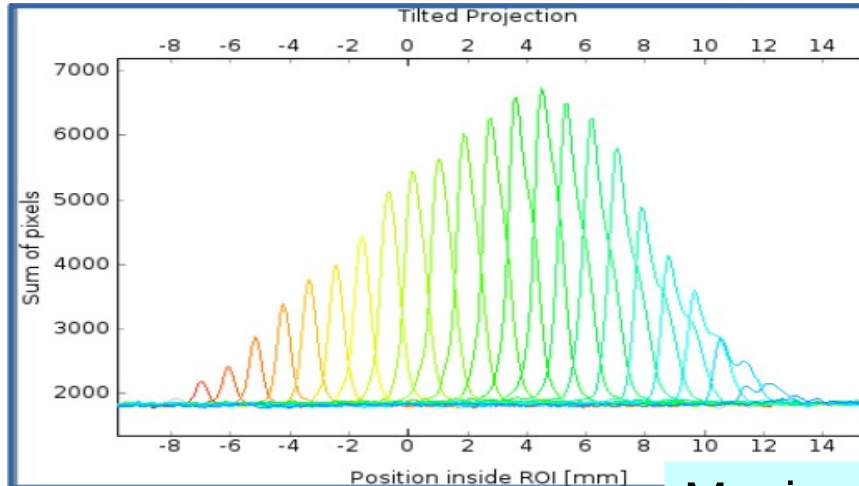
Add-on should be sub-classed from AddonBase class:

```
class AddonBase():
    def __init__(self, imager):
        self.imager = imager
    def entryName(self):
    def controlPane(self):
    def addon_clicked(self, **kwargs):
    def process(self):
    def exit(self):
```

Add-ons for emittance measurement

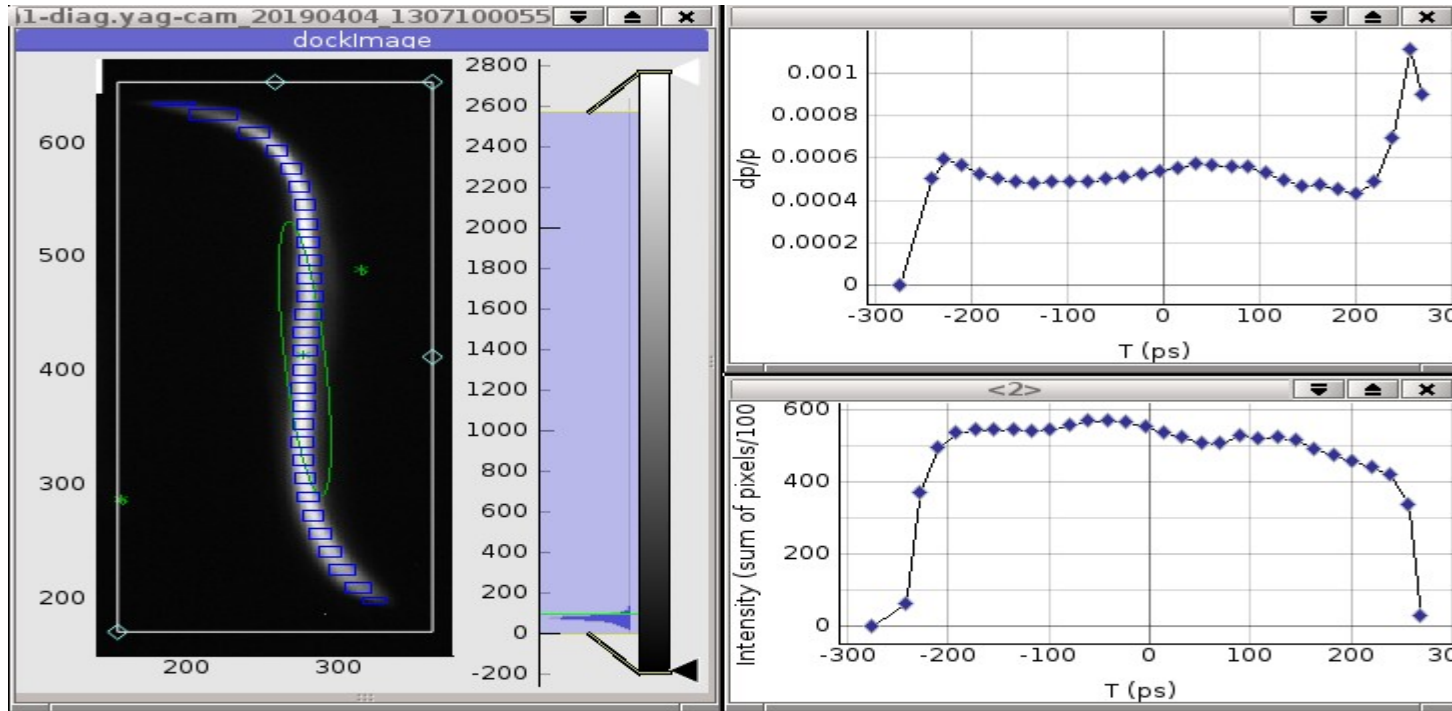


Multi-slit station



Moving slit station

RF-Diagnostic Add-on



ROI is split into N slices, standard analysis performed on each slice.

A momentum spread and delay in a bunch is calculated, based on actual beamline parameters.

Logging of Calculated Parameters

- Currently about **30 cameras** are instrumented in the RHIC complex. Each of these cameras is served by a corresponding **imageMan** program, which is a graphic-less version of the **imageViewer**.
- It continuously publishes results of the image analysis and calculated beam-related parameters to the RHIC logging system.
- The interactively adjusted parameters of an imageViewer can be synchronized with the corresponding imageMan.
- imageViewer initializes from imageMan.
- imageViewer can log local analysis to a JSON file.

Summary

- Pure Python, general purpose image analysis tool.
- Multi-object analysis.
- Easy extendable with add-ons.
- Standard analysis time of a 2.4 Mpixel 12-bit image is 11 ms, de-speckling time of the ROI: 70 ms.
- ~30 cameras at RHIC are served by imageMan's and monitored by imageViewer's. All calculated parameters are logged continuously by RHIC Controls.
- <https://github.com/ASukhanov/Imagin>