

Emwiz scan class for Virtual Pepper-Pot

Class structure

Example object and use

Charge cut

Georgi Georgiev
PITZ Physics Seminar
02.05.2019

Emwiz scan class

Structure

- Object oriented approach
 - One scan – one object
- Properties
 - Key measurement results
 - Internal use variables
- Methods
 - Calculation of results
 - Cascading calls
- Under development
 - Mostly complete (like beta version)
 - Some setup information is hard coded in current version
 - Mean beam momentum
 - Drift

```
% Valid only for single axis (X or Y)
classdef emwiz_scan < handle
    % single slit scan class
    properties
        emwiz_file % (structure) directory + filename(s)
        noisecut % filter function handle
        axis % scan axis (x - horizontal, y - vertical)

        bc_img % beamlet collector (screen) image (for MOI)
        bc_bkg % beamlet collector (screen) background (for MOI)
        bc_sdv % beamlet collector (screen) background deviation (for MOI)
        bc_prop % beamlet collector (screen) properties
        bc_moi % beamlet collector (screen) MOI
        bc_pos % beamlet collector (screen) pixel position (along axis)
        bc_xroi % rectangular region of interest, based on MOI
        bc_yroi % rectangular region of interest, based on MOI

        em_img % EMSY image
        em_bkg % EMSY background
        em_sdv % EMSY background deviation
        em_moi % EMSY MOI
        em_prop % EMSY screen properties
        em_proj % EMSY image projection
        em_pos % EMSY screen pixel position (along axis)

        bl_frame % beamlet image frames
        bl_bkg % beamlet background (frames?)
        bl_sdv % beamlet background deviation
        bl_proj % beamlet image projections
        bl_sop % beamlet Sum Of Pixels

        slit_pos % split positions for each frame

        fit_para % best fit parameter values
        %beam_fraction % 1 - charge cut
        scaling_factor % emittance scaling factor
        charge_cut % charge cut fraction
        drift % length of the drift

        phase_space % the reconstructed phase space
        beam_matrix % 2x2 covariance matrix
        emit_geo % geometrical emittance
        emit_nor % normalized emittance
        emit_sca % scaled normalized emittance
    end
    methods
        function obj = emwiz_scan(directory, scanframes)
```

Contents of an object

Example object

- Processed image data
 - EMSY
 - High screen (full beam)
 - Beamlet frames
- Coordinates
 - Slit positions
 - Pixel positions
- Fit parameters
 - Charge cut
- Beam matrix
- Emittance values
- *Subject to change*

[emwiz_scan](#) with properties:

```
emwiz_file: {1x2 cell}
  noise_cut: @emcalc3
    axis: 'y'
    bc_img: [161x153 double]
    bc_bkg: []
    bc_sdv: []
    bc_prop: [1x1 struct]
    bc_moi: [161x153 logical]
    bc_pos: [1x161 double]
    bc_xroi: [1x153 double]
    bc_yroi: [1x161 double]
    em_img: [74x63 double]
    em_bkg: []
    em_sdv: []
    em_moi: [74x63 logical]
    em_prop: [1x1 struct]
    em_proj: [1x74 double]
    em_pos: [1x74 double]
  bl_frame: [161x153x65 double]
  bl_bkg: []
  bl_sdv: []
  bl_proj: []
  bl_sop: [1x65 double]
  slit_pos: [1x65 double]
  fit_para: [4.2443 0.0204 0.0307]
scaling_factor: 1.9544
charge_cut: 0.6319
drift: 3133
phase_space: []
beam_matrix: [2x2 double]
  emit_geo: 0.0115
  emit_nor: 0.5034
  emit_sca: 0.9839
```

Example use

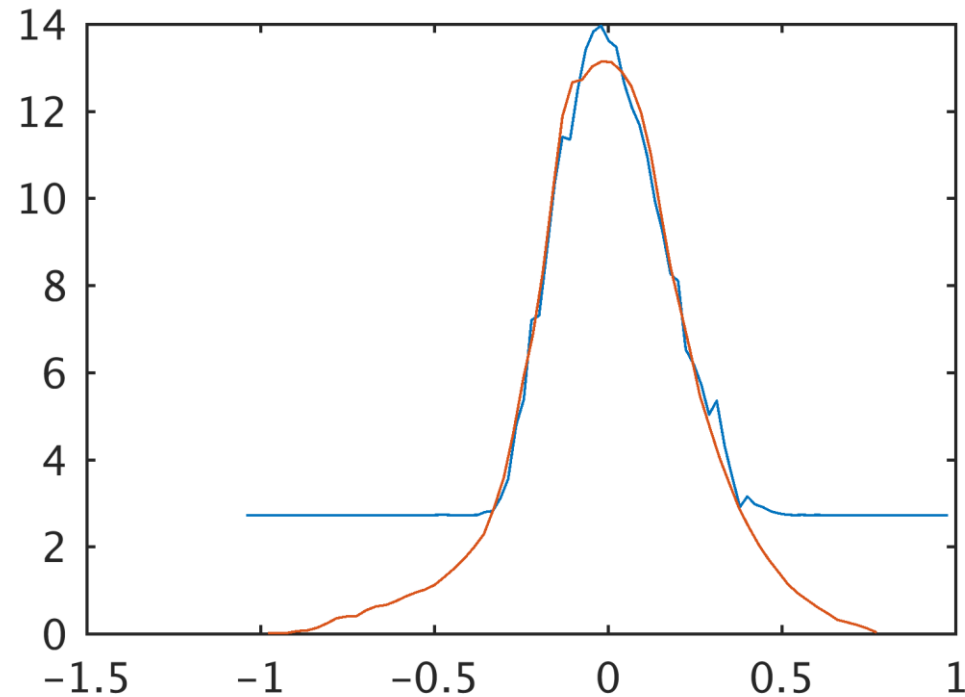
Simplicity

- Initialize
- Load and process images (on demand)
- Call one method
 - Cascading calls to other required methods
 - Lazy paradigm (on demand)

```
>> A = emwiz_scan('/afs/ihf.de/group/pitz/doocs/measure/TransvPhSp/2017/ProjEmittance/201702/17N/382A/EMSY1/', ...  
                 'EMSY1X_382A_2017_02_18_02_43_37__from65_50-to67_70_fast.imc');  
>> A.calc_chargecut()
```

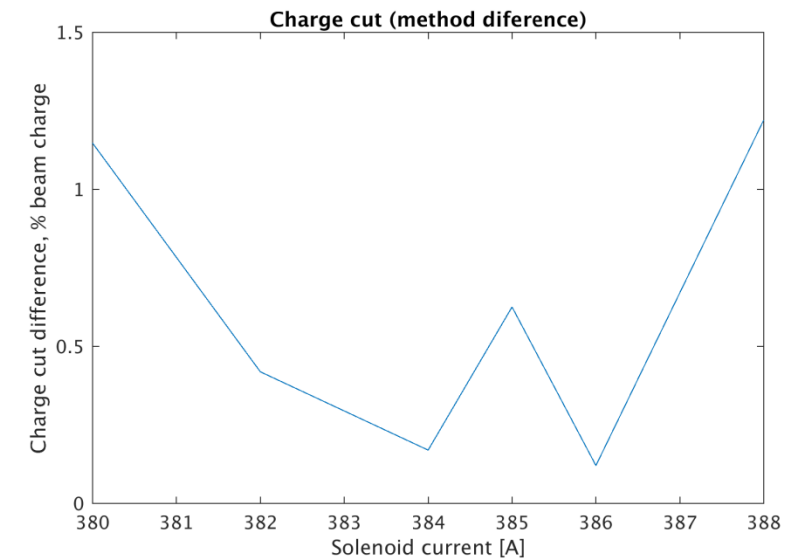
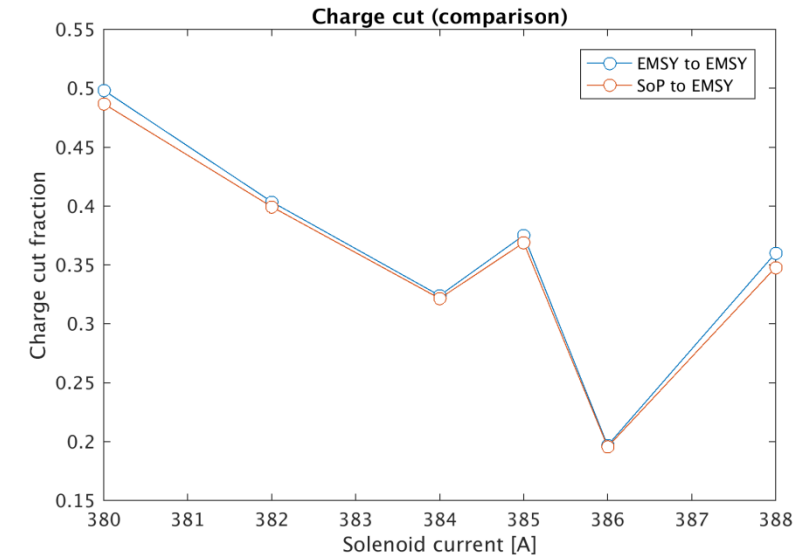
```
Optimization terminated:  
  the current x satisfies the termination criteria using OPTIONS.To1X of 1.000000e-04  
  and F(X) satisfies the convergence criteria using OPTIONS.To1Fun of 1.000000e-04
```

```
>> plot(A.slit_pos+A.fit_para(3),A.bl_sop*A.fit_para(2)+A.fit_para(1))  
>> hold on;  
>> plot(A.em_pos,A.em_proj)  
>> hold off;
```



Charge cut calculation

- Charge cut calculation performed
- Two methods
 - Cut EMSY projection and compare to EMSY
 - Current method
 - Compare normalized SoP to EMSY
- Value deviations
 - Between methods – about 1 %
 - Between script versions – several %
- *Discussion*



THANK YOU!