

---

# **Fiducial Registration Educational Demonstration Documentation**

**Stephen Thompson**

**Mar 08, 2022**



# CONTENTS

<b>1</b>	<b>Citing</b>	<b>3</b>
<b>2</b>	<b>Developing</b>	<b>5</b>
<b>3</b>	<b>Installing</b>	<b>7</b>
<b>4</b>	<b>Licensing and copyright</b>	<b>9</b>
<b>5</b>	<b>Acknowledgements</b>	<b>11</b>
	<b>Python Module Index</b>	<b>19</b>
	<b>Index</b>	<b>21</b>





Author: Stephen Thompson

This is the Fiducial Registration Educational Demonstration (SciKit-SurgeryFRED). This version uses a graphical user interface based on Matplotlib and has been superseded by the browser based version at [SciKit-SurgeryFRED](#).

Fiducial Registration Educational Demonstration (SciKit-SurgeryFRED) is part of the [SciKit-Surgery](#) software project, developed at the [Wellcome EPSRC Centre for Interventional and Surgical Sciences](#), part of [University College London \(UCL\)](#).

Fiducial Registration Educational Demonstration is tested with Python 3.X

Fiducial Registration Educational Demonstration is intended to be used as part of an online tutorial in using fiducial based registration. The tutorial covers the basic theory of fiducial based registration, which is used widely in image guided interventions. The tutorial aims to help the students develop an intuitive understanding of key concepts in fiducial based registration, including Fiducial Localisation Error, Fiducial Registration Error, and Target Registration Error.

```
python sksurgeryfred.py
```

Please explore the project structure, and implement your own functionality.



## **CITING**

If you use SciKit-SurgeryFRED in your research or teaching please cite it. Individual releases can be cited via the Zenodo tag. SciKit-Surgery should be cited as:

Thompson S, Dowrick T, Ahmad M, et al. "SciKit-Surgery: compact libraries for surgical navigation." International Journal of Computer Assisted Radiology and Surgery. 2020 May. DOI: 10.1007/s11548-020-02180-5.





## 2.1 Cloning

You can clone the repository using the following command:

```
git clone https://github.com/SciKit-Surgery/scikit-surgeryfredmatplotlib
```

## 2.2 Running tests

Pytest is used for running unit tests:

```
pip install pytest  
python -m pytest
```

## 2.3 Linting

This code conforms to the PEP8 standard. Pylint can be used to analyse the code:

```
pip install pylint  
pylint --rcfile=tests/pylintrc sksurgeryfredmatplotlib
```



## INSTALLING

You can pip install directly from the repository as follows:

```
pip install git+https://github.com/SciKit-Surgery/scikit-surgeryfredmatplotlib
```

### 3.1 Contributing

Please see the [contributing guidelines](#).

### 3.2 Useful links

- [Source code repository](#)
- [Documentation](#)



## LICENSING AND COPYRIGHT

Copyright 2020 University College London. Fiducial Registration Educational Demonstration is released under the BSD-3 license. Please see the [license file](#) for details.



## ACKNOWLEDGEMENTS

Supported by [Wellcome](#) and [EPSRC](#).

### 5.1 Requirements for Fiducial Registration Educational Demonstration

This is the software requirements file for Fiducial Registration Educational Demonstration, part of the SNAPPY project. The requirements listed below should define what Fiducial Registration Educational Demonstration does. Each requirement can be matched to a unit test that checks whether the requirement is met.

#### 5.1.1 Requirements

ID	Description	Test
0000	Module has a help page	pylint, see tests/pylint.rc and tox.ini
0001	Functions are documented	pylint, see tests/pylint.rc and tox.ini
0002	Package has a version number	No test yet, handled by git.

### 5.2 latest

#### 5.2.1 `sksurgeryfredmatplotlib` package

##### Subpackages

`sksurgeryfredmatplotlib.algorithms` package

##### Submodules

`sksurgeryfredmatplotlib.algorithms.ablation` module

Functions for point based registration using Orthogonal Procrustes.

**class** `sksurgeryfredmatplotlib.algorithms.ablation.Ablator`(*margin*)

Bases: `object`

handles the simulated ablation for scikit-surgery fred

**ablate**(*estimated\_target*)  
performs and ablation, returns a score.

**decrease\_margin**()  
Make the margin smaller

**increase\_margin**()  
Make the margin bigger

**setup**(*target, target\_radius*)  
Setup target etc.

### sksurgeryfredmatplotlib.algorithms.add\_fiducial module

Functions to support MedPhys Taught Module workshop on calibration and tracking

**class** sksurgeryfredmatplotlib.algorithms.add\_fiducial.**AddFiducialMarker**(*fig, plotter, pbr, logger, fixed\_fle\_sd, moving\_fle\_sd, max\_fids=None*)

Bases: object

A class to handle mouse press events, adding a fiducial marker.

**reset\_fiducials**(*mean\_fle\_sq*)  
resets the fiducial markers

### sksurgeryfredmatplotlib.algorithms.fit\_contour module

Fit a contour to an image

sksurgeryfredmatplotlib.algorithms.fit\_contour.**find\_outer\_contour**(*image, alpha=0.015, beta=10.0*)

Fits an active contour to the outer most edge in the image :params image: the image to fit to :params alpha: Snake length shape parameter. Higher values makes

snake contract faster (default 0.015)

**Params beta** Snake smoothness shape parameter. Higher values makes snake smoother (default 10.0)

**Returns** the resulting contour and the initialising contour

sksurgeryfredmatplotlib.algorithms.fit\_contour.**to\_gray**(*image*)

converts and image to grayscale if not already done :params image: The image to convert, can be gray or rgb :returns: a grayscale version



## Module contents

### sksurgeryfredmatplotlib.logging package

#### Submodules

#### sksurgeryfredmatplotlib.logging.fred\_logger module

Class to handle sksurgeryfred logging

**class** sksurgeryfredmatplotlib.logging.fred\_logger.**Logger**(*config*)

Bases: object

Implements logging functionality for sksurgeryfred. Configuration is done by passing a dictionary on construction. Subsequent calls to log("message") will write to log file.

**Parameters** *config* –

- a dictionary containing configuration

parameters. If dictionary contains no "logger" entry then an empty logger is created and subsequent calls to log() will have no effect. Otherwise a logger is created according to the entries in the logger config dictionary. ("log file name", "overwrite existing"

**Raises** IOError if the user can't write to the named log file?

**log**(*message*)

If logging, passes message to logger

**log\_result**(*actual\_tre, fre, expected\_tre, expected\_fre, mean\_fle, no\_fids*)

Writes the registration result to log file

**log\_score**(*state\_string, score*)

Writes the registration result to log file

**read\_log**()

reads a log file and returns lists of values

## Module contents

### sksurgeryfredmatplotlib.plotting package

#### Submodules

#### sksurgeryfredmatplotlib.plotting.interactive\_plots module

Functions to support MedPhys Taught Module workshop on calibration and tracking

**class** sksurgeryfredmatplotlib.plotting.interactive\_plots.**PlotRegStatistics**(*plot*)

Bases: object

writes the registration statistics

**set\_visibilities**(*fids\_text, tre\_text, exp\_tre\_text, exp\_fre\_text, fre\_text, score\_text, total\_score\_text, margin\_text, repeats\_text*)

Sets which text boxes will be visible

**update\_fids\_stats**(*no\_fids, mean\_fle*)

Updates the fids stats display

**update\_last\_score**(*last\_score*)

Updates the margin text box

**update\_margin\_stats**(*margin*)

Updates the margin text box

**update\_repeats**(*repeats*)

Updates the total score text box

**update\_stats\_plot**(*tre, exp\_tre, fre, exp\_fre*)

Updates the statistics display

**update\_total\_score**(*total\_score*)

Updates the total score text box

**class** `sksurgeryfredmatplotlib.plotting.interactive_plots.PlotRegistrations`(*fixed\_plot, moving\_plot, stats\_plot*)

Bases: `object`

Plots the results of registrations

**initialise\_new\_reg**(*img, target\_point, outline*)

resets the registration

**plot\_fiducials**(*fixed\_points, moving\_points, no\_fids, mean\_fle*)

Updates plot with fiducial data

**plot\_registration\_result**(*actual\_tre, expected\_tre, fre, expected\_fre, transformed\_target\_2d*)

Plots the results of a registration

### **sksurgeryfredmatplotlib.plotting.plotting module**

Plotting functions for scikit-surgeryFRED

`sksurgeryfredmatplotlib.plotting.plotting.plot_results`(*logfile*)

Plots the results of multiple runs, from the log file.

### **Module contents**

#### **sksurgeryfredmatplotlib.ui package**

#### **Submodules**

#### **sksurgeryfredmatplotlib.ui.sksurgeryfred module**

User interfaces for sksurgeryFRED

`sksurgeryfredmatplotlib.ui.sksurgeryfred.run_demo`(*image*)

Run FRED

### sksurgeryfredmatplotlib.ui.sksurgeryfred\_command\_line module

Command line processing

`sksurgeryfredmatplotlib.ui.sksurgeryfred_command_line.main(args=None)`  
Entry point for Fiducial Registration Educational Demonstration application

### sksurgeryfredmatplotlib.ui.sksurgeryfred\_game module

User interfaces for sksurgeryFRED

`sksurgeryfredmatplotlib.ui.sksurgeryfred_game.run_demo(image)`  
Run FRED game

### sksurgeryfredmatplotlib.ui.sksurgeryfred\_game\_command\_line module

Command line processing

`sksurgeryfredmatplotlib.ui.sksurgeryfred_game_command_line.main(args=None)`  
Entry point for Fiducial Registration Educational Demonstration application

### sksurgeryfredmatplotlib.ui.sksurgeryfred\_plotter module

User interfaces for sksurgeryFRED

`sksurgeryfredmatplotlib.ui.sksurgeryfred_plotter.run_plotter(logfile)`  
Run FRED Plotter

### sksurgeryfredmatplotlib.ui.sksurgeryfred\_plotter\_command\_line module

Command line processing

`sksurgeryfredmatplotlib.ui.sksurgeryfred_plotter_command_line.main(args=None)`  
Entry point for Fiducial Registration Educational Demonstration application

## Module contents

Fiducial Registration Educational Demonstration

### sksurgeryfredmatplotlib.widgets package

#### Submodules

### sksurgeryfredmatplotlib.widgets.fred\_common module

The main widget for the interactive registration part of scikit-surgeryFRED

**class** `sksurgeryfredmatplotlib.widgets.fred_common.FredCommon(image_file_name, headless=False)`  
Bases: `object`  
an interactive window for doing live registration

**init\_reg()**  
sets up the registration

### sksurgeryfredmatplotlib.widgets.interactive\_registration module

The main widget for the interactive registration part of scikit-surgeryFRED

**class** sksurgeryfredmatplotlib.widgets.interactive\_registration.**InteractiveRegistration**(*image\_file\_name*,  
*headless=False*)

Bases: *sksurgeryfredmatplotlib.widgets.fred\_common.FredCommon*

an interactive window for doing live registration

**initialise\_registration()**  
sets up the registration

**keypress\_event**(*event*)  
handle a key press event

### sksurgeryfredmatplotlib.widgets.registration\_game module

The main widget for the interactive registration part of scikit-surgeryFRED

**class** sksurgeryfredmatplotlib.widgets.registration\_game.**RegistrationGame**(*image\_file\_name*,  
*headless=False*)

Bases: *sksurgeryfredmatplotlib.widgets.fred\_common.FredCommon*

an interactive window for doing live registration

**initialise\_registration()**  
sets up the registration

**keypress\_event**(*event*)  
handle a key press event

**class** sksurgeryfredmatplotlib.widgets.registration\_game.**VisibilitySettings**(*buffer\_size*)

Bases: object

randomly selects from list of visibilities, has five states FLE and no fids Expected FRE Expected TRE Actual FRE

**get\_vis\_state()**  
returns a random visibility state

### Module contents

### Module contents

FiducialRegistrationEducationalDemonstration

### 5.2.2 sksurgeryfredmatplotlib\_game module

### 5.2.3 sksurgeryfredmatplotlib\_plotter module

## 5.3 First notebook

You can write up experiments in notebooks, and they can be generated into Sphinx docs using `tox -e docs`, and for example set up to run on readthedocs.

See [this](#) and [this](#) examples.

### 5.3.1 NOTE:

Getting jupyter to run your code in this package relies on 3 things:

- You must ensure you start jupyter within the tox environment.

```
# If not already done.
source .tox/py36/bin/activate

# Then launch jupyter
jupyter notebook
```

- Then when you navigate to and run this notebook, select the right kernel (named after your project) from the kernel menu item, in the web browser.
- Add project folder to system path, as below.

```
[1]: # Jupyter notebook sets the cwd to the folder containing the notebook.
# So, you want to add the root of the project to the sys path, so modules load correctly.
import sys
sys.path.append("../..")
```

- modindex
- genindex
- search



## PYTHON MODULE INDEX

### S

- `skssurgeryfredmatplotlib`, 16
- `skssurgeryfredmatplotlib.algorithms`, 13
- `skssurgeryfredmatplotlib.algorithms.ablation`,  
11
- `skssurgeryfredmatplotlib.algorithms.add_fiducial`,  
12
- `skssurgeryfredmatplotlib.algorithms.fit_contour`,  
12
- `skssurgeryfredmatplotlib.logging`, 13
- `skssurgeryfredmatplotlib.logging.fred_logger`,  
13
- `skssurgeryfredmatplotlib.plotting`, 14
- `skssurgeryfredmatplotlib.plotting.interactive_plots`,  
13
- `skssurgeryfredmatplotlib.plotting.plotting`, 14
- `skssurgeryfredmatplotlib.ui`, 15
- `skssurgeryfredmatplotlib.ui.skssurgeryfred`, 14
- `skssurgeryfredmatplotlib.ui.skssurgeryfred_command_line`,  
15
- `skssurgeryfredmatplotlib.ui.skssurgeryfred_game`,  
15
- `skssurgeryfredmatplotlib.ui.skssurgeryfred_game_command_line`,  
15
- `skssurgeryfredmatplotlib.ui.skssurgeryfred_plotter`,  
15
- `skssurgeryfredmatplotlib.ui.skssurgeryfred_plotter_command_line`,  
15
- `skssurgeryfredmatplotlib.widgets`, 16
- `skssurgeryfredmatplotlib.widgets.fred_common`,  
15
- `skssurgeryfredmatplotlib.widgets.interactive_registration`,  
16
- `skssurgeryfredmatplotlib.widgets.registration_game`,  
16
- `skssurgeryfredmatplotlib_game`, 17
- `skssurgeryfredmatplotlib_plotter`, 17





## INDEX

### A

`ablate()` (*sksurgeryfredmat-plotlib.algorithms.ablation.Ablator method*), 11

`Ablator` (class in *sksurgeryfredmat-plotlib.algorithms.ablation*), 11

`AddFiducialMarker` (class in *sksurgeryfredmat-plotlib.algorithms.add\_fiducial*), 12

### D

`decrease_margin()` (*sksurgeryfredmat-plotlib.algorithms.ablation.Ablator method*), 12

### F

`find_outer_contour()` (in module *sksurgeryfredmat-plotlib.algorithms.fit\_contour*), 12

`FredCommon` (class in *sksurgeryfredmat-plotlib.widgets.fred\_common*), 15

### G

`get_vis_state()` (*sksurgeryfredmat-plotlib.widgets.registration\_game.VisibilitySettings method*), 16

### I

`increase_margin()` (*sksurgeryfredmat-plotlib.algorithms.ablation.Ablator method*), 12

`init_reg()` (*sksurgeryfredmat-plotlib.widgets.fred\_common.FredCommon method*), 15

`initialise_new_reg()` (*sksurgeryfredmat-plotlib.plotting.interactive\_plots.PlotRegistrations method*), 14

`initialise_registration()` (*sksurgeryfredmat-plotlib.widgets.interactive\_registration.InteractiveRegistration method*), 16

`initialise_registration()` (*sksurgeryfredmat-plotlib.widgets.registration\_game.RegistrationGame method*), 16

`InteractiveRegistration` (class in *sksurgeryfredmat-plotlib.widgets.interactive\_registration*), 16

### K

`keypress_event()` (*sksurgeryfredmat-plotlib.widgets.interactive\_registration.InteractiveRegistration method*), 16

`keypress_event()` (*sksurgeryfredmat-plotlib.widgets.registration\_game.RegistrationGame method*), 16

### L

`log()` (*sksurgeryfredmat-plotlib.logging.fred\_logger.Logger method*), 13

`log_result()` (*sksurgeryfredmat-plotlib.logging.fred\_logger.Logger method*), 13

`log_score()` (*sksurgeryfredmat-plotlib.logging.fred\_logger.Logger method*), 13

`Logger` (class in *sksurgeryfredmat-plotlib.logging.fred\_logger*), 13

### M

`main()` (in module *sksurgeryfredmat-plotlib.ui.sksurgeryfred\_command\_line*), 15

`main()` (in module *sksurgeryfredmat-plotlib.ui.sksurgeryfred\_game\_command\_line*), 15

`main()` (in module *sksurgeryfredmat-plotlib.ui.sksurgeryfred\_plotter\_command\_line*), 15

module

`sksurgeryfredmatplotlib`, 16

`sksurgeryfredmatplotlib.algorithms`, 13

`sksurgeryfredmatplotlib.algorithms.ablation`, 11

`sksurgeryfredmatplotlib.algorithms.add_fiducial`, 12

sksurgeryfredmatplotlib.algorithms.fit\_contour, 12  
 sksurgeryfredmatplotlib.algorithms.add\_fiducial, 12  
 sksurgeryfredmatplotlib.logging, 13  
 sksurgeryfredmatplotlib.logging.fred\_logger, 13  
 sksurgeryfredmatplotlib.plotting, 14  
 sksurgeryfredmatplotlib.plotting.interactive\_plots, 13  
 sksurgeryfredmatplotlib.plotting.plotting, 14  
 sksurgeryfredmatplotlib.ui, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred, 14  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_command\_line, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_game, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_game\_command\_line, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_plotter, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_plotter\_command\_line, 15  
 sksurgeryfredmatplotlib.widgets, 16  
 sksurgeryfredmatplotlib.widgets.fred\_common, 15  
 sksurgeryfredmatplotlib.widgets.interactive\_registration, 16  
 sksurgeryfredmatplotlib.widgets.registration\_game, 16  
 sksurgeryfredmatplotlib\_game, 17  
 sksurgeryfredmatplotlib\_plotter, 17

**S**

set\_visibilities(), 13  
 setup(), 12  
 sksurgeryfredmatplotlib, 16  
 sksurgeryfredmatplotlib.algorithms, 13  
 sksurgeryfredmatplotlib.algorithms.ablation, 11  
 sksurgeryfredmatplotlib.algorithms.add\_fiducial, 12  
 sksurgeryfredmatplotlib.algorithms.fit\_contour, 12  
 sksurgeryfredmatplotlib.logging, 13  
 sksurgeryfredmatplotlib.logging.fred\_logger, 13  
 sksurgeryfredmatplotlib.plotting, 14  
 sksurgeryfredmatplotlib.plotting.interactive\_plots, 13  
 sksurgeryfredmatplotlib.plotting.plotting, 14  
 sksurgeryfredmatplotlib.ui, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred, 14  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_command\_line, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_game, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_game\_command\_line, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_plotter, 15  
 sksurgeryfredmatplotlib.ui.sksurgeryfred\_plotter\_command\_line, 15  
 sksurgeryfredmatplotlib.widgets, 16  
 sksurgeryfredmatplotlib.widgets.fred\_common, 16

**P**

plot\_fiducials(), 14  
 plot\_registration\_result(), 14  
 plot\_results(), 14  
 PlotRegistrations (class in sksurgeryfredmatplotlib.plotting.interactive\_plots), 14  
 PlotRegStatistics (class in sksurgeryfredmatplotlib.plotting.interactive\_plots), 13

**R**

read\_log(), 13  
 RegistrationGame (class in sksurgeryfredmatplotlib.widgets.registration\_game), 16

module, 15  
sksurgeryfredmatplotlib.widgets.interactive\_registration  
module, 16  
sksurgeryfredmatplotlib.widgets.registration\_game  
module, 16  
sksurgeryfredmatplotlib\_game  
module, 17  
sksurgeryfredmatplotlib\_plotter  
module, 17

## T

to\_gray() (in module sksurgeryfredmat-  
plotlib.algorithms.fit\_contour), 12

## U

update\_fids\_stats() (sksurgeryfredmat-  
plotlib.plotting.interactive\_plots.PlotRegStatistics  
method), 13  
update\_last\_score() (sksurgeryfredmat-  
plotlib.plotting.interactive\_plots.PlotRegStatistics  
method), 14  
update\_margin\_stats() (sksurgeryfredmat-  
plotlib.plotting.interactive\_plots.PlotRegStatistics  
method), 14  
update\_repeats() (sksurgeryfredmat-  
plotlib.plotting.interactive\_plots.PlotRegStatistics  
method), 14  
update\_stats\_plot() (sksurgeryfredmat-  
plotlib.plotting.interactive\_plots.PlotRegStatistics  
method), 14  
update\_total\_score() (sksurgeryfredmat-  
plotlib.plotting.interactive\_plots.PlotRegStatistics  
method), 14

## V

VisibilitySettings (class in sksurgeryfredmat-  
plotlib.widgets.registration\_game), 16