scikit-surgeryutils Documentation

Matt Clarkson

Jan 18, 2021

Contents

CHAPTER 1

Features

- [Common overlay apps](#) - Examples of common uses of scikit-surgeryvtk's VTKOverlayWindow. Includes overlaying on a video feed, duplicating a feed.

Command Line Apps

- sksurgeryvideolag.py - shows a millisecond timer and video image to crudely measure measure lag.

- sksurgerycharucotest.py - extracts charuco points and annotates video image with each id detected.

- sksurgeryrendermodelslikecamera.py - renders a VTK model, over background image, using OpenCV camera intrinsics.

- sksurgerymakecalibrationdots.py - Create a calibraiton dot pattern.

- sksurgeryreslice.py - DICOM reslice widget demo.

- sksurgerytextoverlay.py - VTK text overlay demo.

- sksurgerytransformpolydata.py - Read a surface mesh (.vtk,.vtp,.stl,.ply file), transform by 4x4 matrix and write as .vtk.

Source code is avaialble on GitHub.

2.1 scikit-surgeryutils

scikit-surgeryutils is part of the SNAPPY software project, developed at the Wellcome EPSRC Centre for Interventional and Surgical Sciences, part of University College London (UCL).

scikit-surgeryutils containing small demo apps and utilities.

2.1.1 Features

- Common overlay apps - Examples of common uses of scikit-surgeryvtk's VTKOverlayWindow. Includes overlaying on a video feed, duplicating a feed.

2.1.2 Command Line Apps

- sksurgeryvideolag.py - shows a millisecond timer and video image to crudely measure measure lag.
- sksurgerycharucotest.py - extracts charuco points and annotates video image with each id detected.
- sksurgeryrendermodelslikecamera.py - renders a VTK model, over background image, using OpenCV camera intrinsics.
- sksurgerymakecalibrationdots.py - Create a calibraiton dot pattern.
- sksurgeryreslice.py - DICOM reslice widget demo.
- sksurgerytextoverlay.py - VTK text overlay demo.
- sksurgerytransformpolydata.py - Read a surface mesh (.vtk,.vtp,.stl,.ply file), transform by 4x4 matrix and write as .vtk.

2.1.3 Installing

You can pip install directly from the repository as follows:

```
pip install git+https://github.com/UCL/scikit-surgeryutils
```

2.1.4 Developing

Cloning

You can clone the repository using the following command:

```
git clone https://github.com/UCL/scikit-surgeryutils
```

Running the tests

You can run the unit tests by installing and running tox:

```
pip install tox
tox
```

Encountering Problems?

Please check list of common issues.

Contributing

Please see the contributing guidelines.

Useful links

- Source code repository
- Documentation

2.1.5 Licensing and copyright

Copyright 2018 University College London. scikit-surgeryutils is released under the BSD-3 license. Please see the license file for details.

2.1.6 Acknowledgements

Supported by Wellcome and EPSRC.

2.2 Common Overlay Apps [Temp Change]

Common use cases for vtk_overlay_window

class sksurgeryutils.common_overlay_apps.**DuplicateOverlayWindow**
>   Bases: sksurgeryutils.common_overlay_apps.OverlayOnVideoFeedCropRecord
>
>   Set the background of vtk_overlay_window to duplicate that of another vtk_overlay_window.
>
>   Example usage: video_source = 0 source_window = OverlayOnVideoFeedCropRecord(video_source)
>
>   duplicate_window = DuplicateOverlayWindow() duplicate_window.set_source_window(source_window)
>
>   **on_record_start**()
>>   Don't want to call the base class version, so override.
>
>   **on_record_stop**()
>>   Don't want to call the base class version, so override.
>
>   **set_roi**()
>>   Don't want to call the base class version, so override.
>
>   **set_source_window**(*source_window*)
>>   Set the source window. :param source_window: The window that contains the image to copy.
>
>   **update**()
>>   Update the frame with a new background image.

class sksurgeryutils.common_overlay_apps.**OverlayBaseApp**(*video_source*,
                                                                    dims=None)

    Bases: `object`

    Base class for applications that use vtk_overlay_window. The update() method should be implemented in the child class.

        Parameters `video_source` – OpenCV compatible video source (int or filename)

    add_vtk_models_from_dir(*directory*)
        Add VTK models to the foreground. :param: directory, location of models

    start()
        Show the overlay widget and set a timer running

    stop()
        Make sure that the VTK Interactor terminates nicely, otherwise it can throw some error messages, depending on the usage.

    update()
        Update the scene background and/or foreground. Should be implemented by sub class

class sksurgeryutils.common_overlay_apps.**OverlayOnVideoFeed**(*video_source*,
                                                                      dims=None)
    Bases: `sksurgeryutils.common_overlay_apps.OverlayBaseApp`

    Uses the acquired video feed as the background image, with no additional processing.

    update()
        Get the next frame of input and display it.

class sksurgeryutils.common_overlay_apps.**OverlayOnVideoFeedCropRecord**(*video_source*,
                                                                              output_filename=None,
                                                                              dims=None)
    Bases: `sksurgeryutils.common_overlay_apps.OverlayBaseApp`

    Add cropping of the incoming video feed, and the ability to record the vtk_overlay_window.

        Parameters

            • **video_source** – OpenCV compatible video source (int or filename)

            • **output_filename** – Location of output video file when recording. If none specified, the current date/time is used as the filename.

    get_output_frame()
        Get the output frame to write in numpy format.

    on_record_start()
        Start recording data on each frame update. It is expected that this will be triggered using a Qt signal e.g. from a button click. (see sksurgerydavinci.ui.Viewers for examples)

    on_record_stop()
        Stop recording data.

    set_roi()
        Crop the incoming video stream using ImageCropper.

    update()
        Get the next frame of input, crop and/or write to file (if either enabled).

2.3 Misc Utilities

Various image utilities that might be useful in this package.

sksurgeryutils.utils.image_utils.**image_to_pixmap**(*rgb_image*)
> Converts an OpenCV image to a Qt pixmap.

>> **Parameters** **rgb_image** – OpenCV image, 3 channel, RGB.

>> **Returns** QPixmap

Any useful utilities relating to displays/screens.

class sksurgeryutils.utils.screen_utils.**ScreenController**
> Bases: object

> This class detects the connected screens/monitors, and returns the primary screen and a list of any secondary screens.

> **list_of_screens**()
>> Return the primary screen and list of other available screens

Index