PYTHON PICAMERA

`python-picamera` is a pure Python interface to the Raspberry Pi camera module for Python 2.7 (or above) or Python 3.2 (or above). The library is written and maintained by Dave Jones.

Also see the camera setup page.

INSTALLATION

The `python-picamera` library is available in the Raspbian archives. Install with `apt`:

```
sudo apt-get update
sudo apt-get install python-picamera
```

Alternatively, the Python3 package is installed with `sudo apt-get install python3-picamera`. An offline version of the documentation is available with `sudo apt-get install python-picamera-docs`.

USAGE

First, at the Python prompt or at the top of a Python script, enter:

```
import picamera
```

This will make the library available to the script. Now create an instance of the PiCamera class:

```
camera = picamera.PiCamera()
```

And take a picture:

```
camera.capture('image.jpg')
```

HORIZONTAL AND VERTICAL FLIP

Like with the `raspistill` command, you can apply a horizontal and vertical flip if your camera is positioned upside-down. This is done by changing the `hflip` and `vflip` properties directly:

```
camera.hflip = True
camera.vflip = True
```

Be sure to use an upper case `T` in `True` as this is a keyword in Python.

PREVIEW

You can display a preview showing the camera feed on screen. Warning: this will overlay your Python session by default; if you have trouble stopping the preview, simply pressing `Ctrl+D` to terminate the Python session is usually enough to restore the display:

```
camera.start_preview()
```

You can use the `stop_preview` method to remove the preview overlay and restore the display:

```
camera.stop_preview()
```

Alternatively, you can access the Pi using SSH from another computer, open a Python prompt and enter these commands, displaying the preview on the monitor connected to the Pi (not the computer you're connected from).

CAMERA SETTINGS

You can change other camera configuration by editing property values, for example:

```
camera.brightness = 70
```

This will change the brightness setting from its default `50` to `70` (values between 0 and 100).

Other settings are available. Here is a list with their default values:

```
camera.sharpness = 0
camera.contrast = 0
```

```
camera.brightness = 50
camera.saturation = 0
camera.ISO = 0
camera.video_stabilization = False
camera.exposure_compensation = 0
camera.exposure_mode = 'auto'
camera.meter_mode = 'average'
camera.awb_mode = 'auto'
camera.image_effect = 'none'
camera.color_effects = None
camera.rotation = 0
camera.hflip = False
camera.vflip = False
camera.crop = (0.0, 0.0, 1.0, 1.0)
```

SLEEP

You can add pauses between commands using `sleep` from the `time` module:

```
import picamera
from time import sleep

camera = picamera.PiCamera()

camera.capture('image1.jpg')
sleep(5)
camera.capture('image2.jpg')
```

You can also use `sleep` in a preview to adjust settings over time:

```
camera.start_preview()

for i in range(100):
    camera.brightness = i
    sleep(0.2)
```

VIDEO RECORDING

Record 5 seconds of video:

```
camera.start_recording('video.h264')
sleep(5)
camera.stop_recording()
```

DOCUMENTATION

Full documentation for `python-picamera` is available at picamera.readthedocs.org

DEVELOPMENT

The `python-picamera` project is written and maintained by Dave Jones and the source can be found at github.com/waveform80/picamera where you can open issues or contribute to the project.

VIEW/EDIT THIS PAGE ON GITHUB

READ OUR USAGE AND CONTRIBUTIONS POLICY

CC BY SA