Flare Detection

Problem Definition

Field inspections are often conducted by using visual cameras on robotic platforms to comprehensively capture high resolution images. However, due to experimental and environmental configurations some data can be corrupted due to lens flare. We would therefore like to tag these images in the dataset. The task here is to develop an algorithm that can automatically detect if an image is subject to lens flare or not.

Data

The dataset is uploaded here.
The folder structure is as follows:

```
dataset
  └── training
        ├── flare
        |      └── image-a.jpg
        |      └── image-b.jpg
        |      └── …..
        └── good
               └── image-x.jpg
               └── image-y.jpg
               └── …..
```

Each subfolder contains 40 images, one set containing “flared” images and another with “good" images.

Your task is to build an algorithm for **detection of presence of lens flare** in images. Put together a self contained python function, 'detector.py' that takes in as input **N** images outputs a sequence of **1**s for 'faulty' images and **0**s for ‘good’ images. Each item is returned as a new line:

```
>> ./detectory.py image01.jpg image02.jpg image03.jpg image04.jpg
1
0
0
1
```

Testing

The tests will be done on data not shared with you, with the algorithm running on Ubuntu 16.04 with python 2.7 and python 3.3. Feel free to use any third party packages/libraries, but please include clear installation instructions and scripts for the examiner. Ease of deployment will be assessed, and you're encouraged to use tools such as virtualenv, conda, or docker if you think your program has a lot of dependencies.

Submissions

Submit the following:

1. Upload the functions onto a git repo and instruction to its use;
2. **One page** document discussing system overview, outline of algorithm, intuition behind algorithm and summary of results; and
3. **Half page** document on exploring future work to address: how would you go about the task of flare removal

Success metrics

You'll be assessed on:

1. Classification accuracy;
2. Ease of code deployment and quality of code;
3. Explanation of process; and
4. Ideas explored in future work.

Example images:

‘Good’ image	‘Flare’ image