

Detecting SIFT feature points

Scale Invariant Feature Transform (SIFT) is one of the most popular features in the field of Computer Vision. David Lowe first proposed this in his seminal paper, which is available at <https://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf>. It has since become one of the most effective features to use for image recognition and content analysis. It is robust against scale, orientation, intensity, and so on. This forms the basis of our object recognition system. Let's take a look at how to detect these feature points.

How to do it...

1. Create a new Python file, and import the following packages:

```
import sys

import cv2
import numpy as np
```

2. Load the input image. We will use `table.jpg` :

```
# Load input image -- 'table.jpg'
input_file = sys.argv[1]
img = cv2.imread(input_file)
```

3. Convert this image to grayscale:

```
img_gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

4. Initialize the SIFT detector object and extract the keypoints:

```
sift = cv2.xfeatures2d.SIFT_create()
keypoints = sift.detect(img_gray, None)
```

5. The keypoints are the salient points, but they are not the features. This basically gives us the location of the salient points. SIFT also functions as a very effective feature extractor, but we will see this aspect of it in one of the later recipes.

6. Draw the keypoints on top of the input image, as follows:

```
img_sift = np.copy(img)
cv2.drawKeypoints(img, keypoints, img_sift, flags=cv2.DRAW_MATCHES_FLAGS_DRAW_RICH
```

7. Display the input and output images:

```
cv2.imshow('Input image', img)
cv2.imshow('SIFT features', img_sift)
cv2.waitKey()
```

8. The full code is given in the [feature_detector.py](#) file that is already provided to you. The input image is as follows:



9. The output image looks like the following:

