

HW2: Image stitching

1. Answer:

Here is the code:

```
from utils import harris, dist2, find_sift
import matplotlib.pyplot as plt
import matplotlib.patches as patches
import skimage
import numpy as np

# add your imports below
import cv2

# read the image
img1 = skimage.io.imread("./uttower_left.jpg")
img2 = skimage.io.imread("./uttower_right.jpg")

# Convert images to float
img1 = img1.astype(np.float32)
img2 = img2.astype(np.float32)

# Convert images to grayscale
img1 = skimage.color.rgb2gray(img1)
img2 = skimage.color.rgb2gray(img2)
```

Here is the result:

2. Answer:

Here is the result:

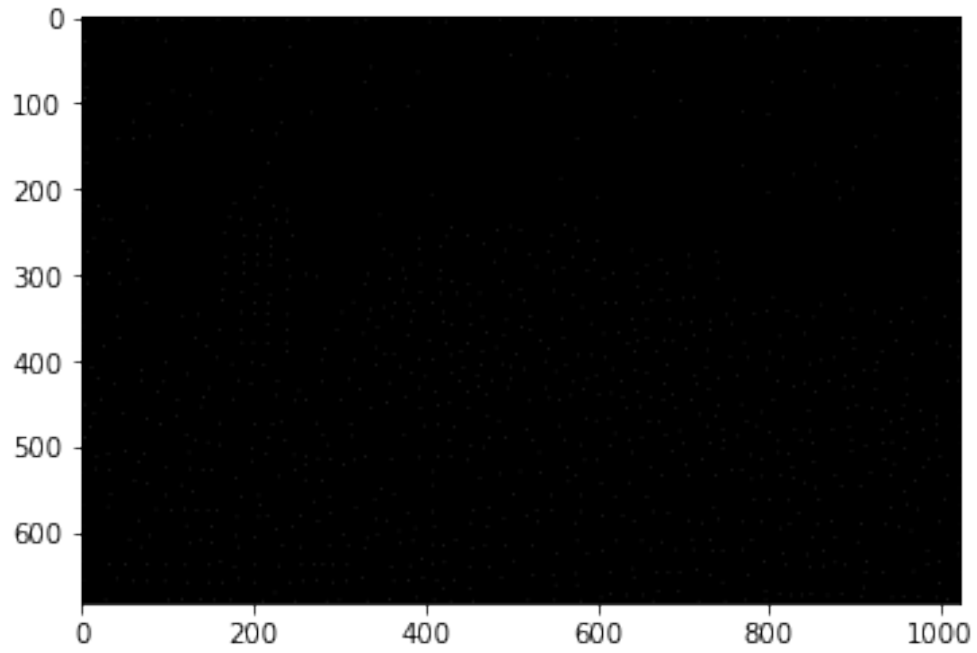


Figure 1: cim1.png

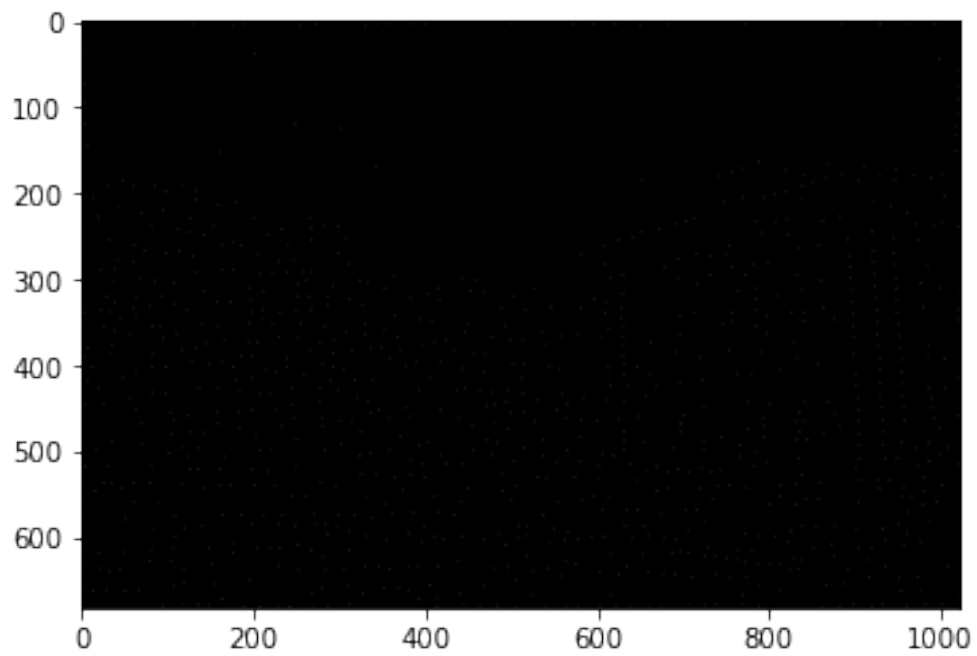


Figure 2: cim2.png



Figure 3: corner1.png

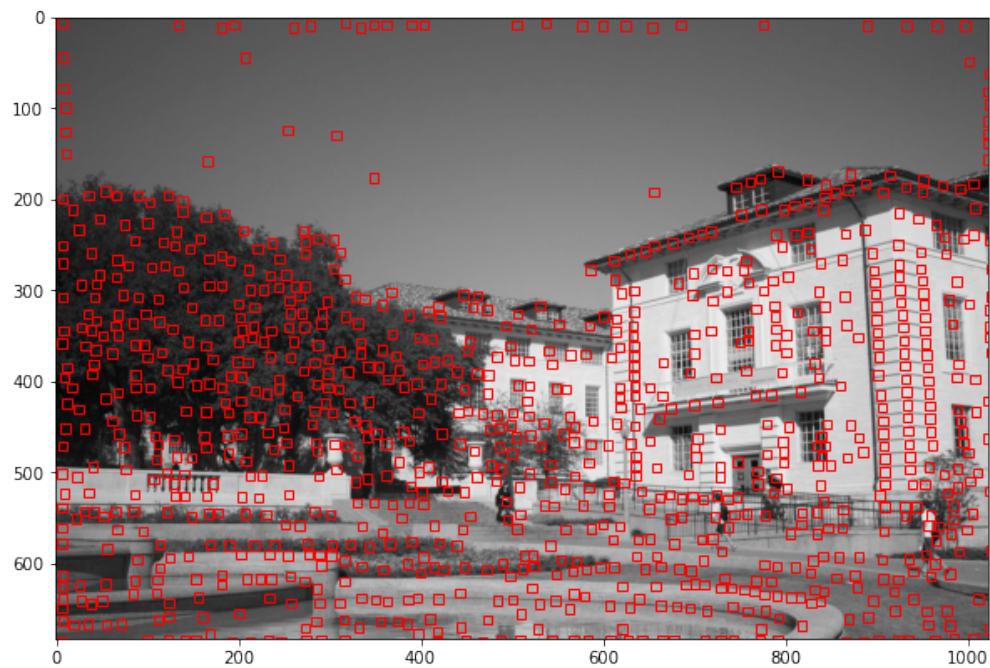


Figure 4: corner2.png

3. Answer:
Code:

```
def neighbor_descriptors(img, r, c, width):
```

```
radius=(width-1)//2
circles = np.vstack((c, r, np.full_like(c, radius))).T
return find_sift(img, circles)
```

```
radius=7
width = 2 * radius +1
descriptors_1 =neighbor_descriptors(img1, r1, c1, width)
descriptors_2 =neighbor_descriptors(img2, r2, c2, width)
# print(f"descriptors1 are:\n{descriptors_1}")
# print(f"descriptors2 are:\n{descriptors_2}")
# print(f"shape of descriptor1{descriptors_1.shape}, shape of descriptor2:{descriptors_2.shape}")
```

4. Answer:

Code:

```
# use dist2 from utils.py to compute dist between descriptors
distances = dist2(descriptors_1, descriptors_2)
```

5. Answer:

```
def filter_descriptors_by_dist(distances, thresh):
    matches = []
    for i in range(distances.shape[0]):

        indices = np.argsort(distances[i])

        if distances[i, indices[0]] < thresh * distances[i, indices[1]]:
            matches.append((i, indices[0]))

    return matches

threshold = 0.8
filtered_matches = filter_descriptors_by_dist(distances, threshold)
```

6. Answer:

number of inliers: 409

residual: 13.782913853102992

7. Answer:

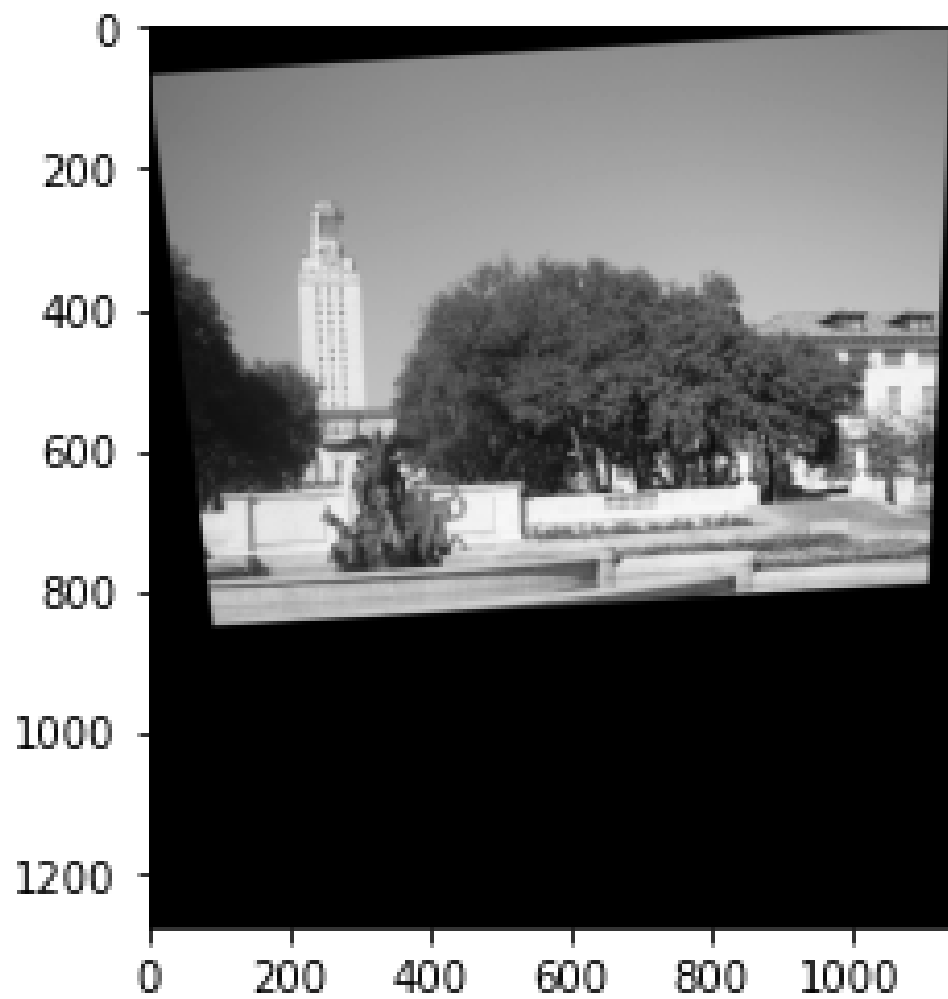


Figure 5: wrappedimg1.png

8. Answer:
Here is the result:

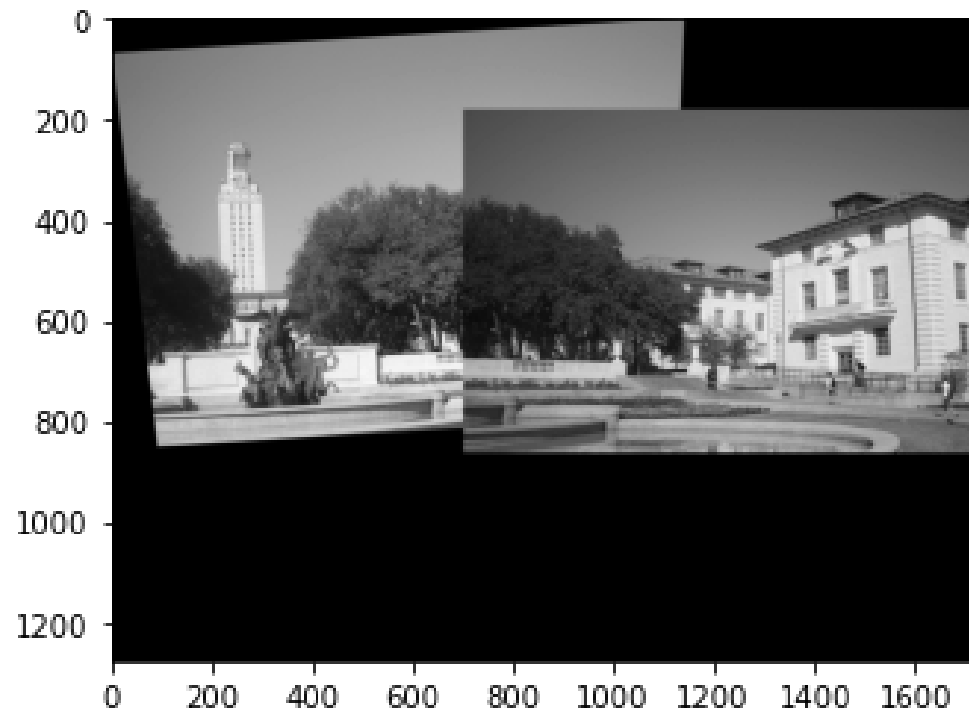


Figure 6: panorama.png

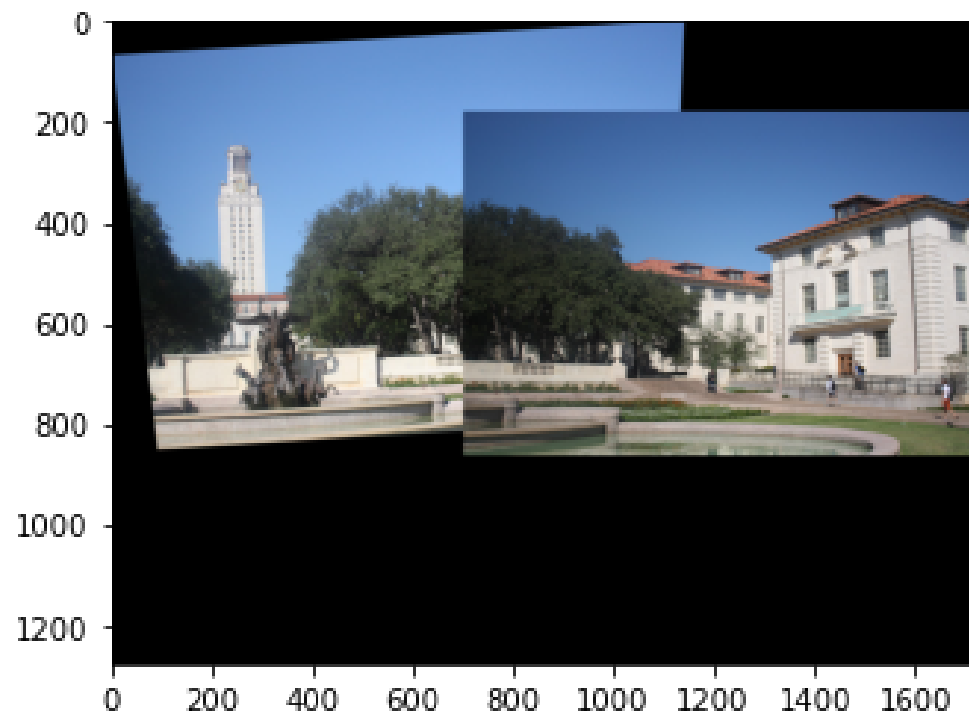


Figure 7: panoramacolor.png