

CÁC ĐẶC TRƯNG ẢNH THÔNG THƯỜNG SIFT, SURF, HOG

1. Thư viện

```
from google.colab import drive
drive.mount('/content/drive')
```

```
import matplotlib.pyplot as plt
```

```
!pip install opencv-python==3.4.2.16
```

```
import cv2
```

2. Đọc ảnh và tìm SIFT

```
#reading image
img1 = cv2.imread('drive/My Drive/TryLogic/tajmahal2.jpg')
gray1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)

#keypoints
sift = cv2.xfeatures2d.SIFT_create()
keypoints_1, descriptors_1 = sift.detectAndCompute(img1, None)

img_1 = cv2.drawKeypoints(gray1, keypoints_1, img1)
plt.rcParams['figure.figsize'] = (21, 8)
plt.imshow(img_1)
plt.title('Key Point Detectors', fontsize = 20)
plt.show()
```

3. Đọc hai ảnh

```

import cv2
import matplotlib.pyplot as plt
%matplotlib inline

# read images
img1 = cv2.imread('drive/My Drive/TryLogic/tajmahal1.jpg')
img2 = cv2.imread('drive/My Drive/TryLogic/tajmahal2.jpg')

img1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
img2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)

figure, ax = plt.subplots(1, 2, figsize=(16, 8))

ax[0].imshow(img1, cmap='gray')
ax[1].imshow(img2, cmap='gray')
plt.title('Checking the Second Image', fontsize = 20)
plt.show()

```

4. Tính SIFT cho 2 ảnh

```

import cv2
import matplotlib.pyplot as plt
%matplotlib inline

# read images
img1 = cv2.imread('drive/My Drive/TryLogic/tajmahal1.jpg')
img2 = cv2.imread('drive/My Drive/TryLogic/tajmahal2.jpg')

img1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
img2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)

#sift
sift = cv2.xfeatures2d.SIFT_create()

keypoints_1, descriptors_1 = sift.detectAndCompute(img1, None)
keypoints_2, descriptors_2 = sift.detectAndCompute(img2, None)

len(keypoints_1), len(keypoints_2)

```

5. So khớp đặc trưng

```

import cv2
import matplotlib.pyplot as plt
%matplotlib inline

# read images
img1 = cv2.imread('drive/My Drive/TryLogic/tajmahal1.jpg')
img2 = cv2.imread('drive/My Drive/TryLogic/tajmahal2.jpg')

img1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
img2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)

#sift
sift = cv2.xfeatures2d.SIFT_create()

keypoints_1, descriptors_1 = sift.detectAndCompute(img1,None)
keypoints_2, descriptors_2 = sift.detectAndCompute(img2,None)

#feature matching
bf = cv2.BFMatcher(cv2.NORM_L1, crossCheck=True)

matches = bf.match(descriptors_1,descriptors_2)
matches = sorted(matches, key = lambda x:x.distance)

plt.rcParams['figure.figsize'] = (15, 18)
img3 = cv2.drawMatches(img1, keypoints_1, img2, keypoints_2, matches[:5], img2, flags=2)
plt.imshow(img3)
plt.title('Matching the Features', fontsize = 20)
plt.show()

```

6. So khớp tất cả đặc trưng

```

import cv2
import matplotlib.pyplot as plt
%matplotlib inline

# read images
img1 = cv2.imread('drive/My Drive/TryLogic/tajmahal1.jpg')
img2 = cv2.imread('drive/My Drive/TryLogic/tajmahal2.jpg')

img1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
img2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)

#sift
sift = cv2.xfeatures2d.SIFT_create()

keypoints_1, descriptors_1 = sift.detectAndCompute(img1, None)
keypoints_2, descriptors_2 = sift.detectAndCompute(img2, None)

#feature matching
bf = cv2.BFMatcher(cv2.NORM_L1, crossCheck=True)

matches = bf.match(descriptors_1, descriptors_2)
matches = sorted(matches, key = lambda x:x.distance)

plt.rcParams['figure.figsize'] = (15, 18)
img3 = cv2.drawMatches(img1, keypoints_1, img2, keypoints_2, matches[:50], img2, flags=2)
plt.imshow(img3)
plt.title('Matching Features with all the Detectors', fontsize = 20)
plt.show()

```

7. Đặc trưng SURF

```

#reading image
img1 = cv2.imread('drive/My Drive/TryLogic/tajmahal2.jpg')
gray1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)

#keypoints
surf = cv2.xfeatures2d.SURF_create()
keypoints_1, descriptors_1 = surf.detectAndCompute(img1, None)

img_1 = cv2.drawKeypoints(gray1, keypoints_1, img1)
plt.rcParams['figure.figsize'] = (21, 8)
plt.imshow(img_1)
plt.title('Key Point Detectors', fontsize = 20)
plt.show()

```

8. So khớp tất cả đặc trưng SURF

```

import cv2
import matplotlib.pyplot as plt
%matplotlib inline

# read images
img1 = cv2.imread('drive/My Drive/TryLogic/tajmahal1.jpg')
img2 = cv2.imread('drive/My Drive/TryLogic/tajmahal2.jpg')

img1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)
img2 = cv2.cvtColor(img2, cv2.COLOR_BGR2GRAY)

#surf
surf = cv2.xfeatures2d.SURF_create()

keypoints_1, descriptors_1 = surf.detectAndCompute(img1, None)
keypoints_2, descriptors_2 = surf.detectAndCompute(img2, None)

#feature matching
bf = cv2.BFMatcher(cv2.NORM_L1, crossCheck=True)

matches = bf.match(descriptors_1, descriptors_2)
matches = sorted(matches, key = lambda x:x.distance)

plt.rcParams['figure.figsize'] = (15, 18)
img3 = cv2.drawMatches(img1, keypoints_1, img2, keypoints_2, matches[:50], img2, flags=2)
plt.imshow(img3)
plt.title('Matching Features with all the Detectors', fontsize = 20)
plt.show()

```

Bài tập nộp ngay:

Yêu cầu: Sinh viên hãy thực hiện với đặc trưng HOG (hog = cv2.HOGDescriptor())

Hãy trích chọn đặc trưng HOG

Gợi ý một vài đặc trưng khác: GIST (<https://github.com/imoken1122/GIST-feature-extra-ctor>)