Sinh viên: Nguyễn Thành Đạt

MSSV: 207CT58538

1. **Mục tiêu**

* Tìm điểm cần so sánh (Harris Corner Detector)
* Xem xét vùng chọn hình chữ nhật xung quanh những điểm cần so sánh
* Tính mô tả đặc trưng cục bộ cho mỗi điểm của mỗi ảnh
* Kiểm tra độ tương đồng giữa hai ảnh

1. **Công cụ**

- Ngôn ngữ: Python

- Thư viện: OpenCV, NumPy

1. **Quy trình thực hiện**

* Bước 1: Tìm điểm góc (Harris Corner)

- Sử dụng hàm `cv2.cornerHarris` để phát hiện các điểm góc trong ảnh.

- Các điểm góc được dùng làm keypoint để tính descriptor.

* Bước 2: Xác định vùng chọn quanh điểm góc

- Xác định các vùng hình vuông xung quanh điểm góc.

- Tạo keypoint bằng `cv2.KeyPoint` với size vùng chọn.

* Bước 3: Tính descriptor đặc trưng

- Dùng ORB (`cv2.ORB\_create()`) để tính toán vector đặc trưng cho các điểm góc.

* Bước 4: So khớp đặc trưng

- Dùng `cv2.BFMatcher` để so khớp descriptor giữa 2 ảnh.

- Sắp xếp các cặp điểm khớp theo độ tương đồng.

- Hiển thị các điểm khớp tốt nhất bằng `cv2.drawMatches`.

4️⃣ Mã nguồn

```python

import cv2

import numpy as np

# Bước 1: Đọc ảnh và kiểm tra

img1 = cv2.imread('image1.jpg', cv2.IMREAD\_GRAYSCALE)

img2 = cv2.imread('image2.jpg', cv2.IMREAD\_GRAYSCALE)

if img1 is None or img2 is None:

print("Không tìm thấy ảnh!")

exit()

# Bước 2: Tìm điểm góc Harris

def tim\_diem\_goc(img):

img\_float = np.float32(img)

dst = cv2.cornerHarris(img\_float, 2, 3, 0.04)

dst = cv2.dilate(dst, None)

return dst

harris1 = tim\_diem\_goc(img1)

harris2 = tim\_diem\_goc(img2)

# Bước 3: Tạo keypoint từ điểm góc

def tao\_keypoint(dst):

keypoints = []

y, x = np.where(dst > 0.01 \* dst.max())

for i in range(len(x)):

kp = cv2.KeyPoint(float(x[i]), float(y[i]), 20)

keypoints.append(kp)

return keypoints

kp1 = tao\_keypoint(harris1)

kp2 = tao\_keypoint(harris2)

# Bước 4: Tính descriptor ORB

orb = cv2.ORB\_create()

kp1, des1 = orb.compute(img1, kp1)

kp2, des2 = orb.compute(img2, kp2)

# So khớp

bf = cv2.BFMatcher(cv2.NORM\_HAMMING, crossCheck=True)

matches = bf.match(des1, des2)

matches = sorted(matches, key=lambda x: x.distance)

# Vẽ kết quả

anh\_ket\_qua = cv2.drawMatches(img1, kp1, img2, kp2, matches[:30], None, flags=2)

cv2.imshow('Ket qua', anh\_ket\_qua)

cv2.waitKey(0)

cv2.destroyAllWindows()