

2021학년도 1학기 컴퓨터비전 [ADS5034-01] 과제 #2

(제출마감 : 5월 9일 23:59)

➤ 유의 사항

- 제출물은 “HW2_학번_이름.zip”으로 압축하여 제출해 주세요. “HW2_학번_이름” 폴더를 생성 후, 그 안에 파이썬 코드와 보고서를 함께 넣어, 압축 후 제출해주시면 됩니다. 이때 코드 구동에 필요한 이미지 등을 같이 포함하여 제출해주세요.
- 함께 제공되는 템플릿 코드는 하나의 예시로서, 함수/변수 이름, 입/출력, 내부 구조 등은 자유롭게 작성하시면 됩니다.
- 보고서(자유 형식)를 반드시 제출하세요. 보고서에는 **중요 코드의 대략적인 설명, 과제 결과와 분석**이 포함되어야 합니다. 채점 시에는 보고서에 담긴 내용을 위주로 채점합니다.
- **다른 사람의 제출물, 온라인 소스를 그대로 베껴 내지 마세요.**
- **OpenCV 3.4.x 버전을 권장드립니다.** 특히, OpenCV 4.x.x 버전의 경우, SIFT와 같은 함수가 공개되어 있지 않을 수 있습니다.
- 과제 관련 질문 사항은, 문의 게시판을 이용해주세요.

[문제 1. Feature Descriptor and Feature Matching (40 pts)]

- 1-1. 임의의 이미지에서 그리드 형식으로 작은 이미지(패치)를 생성하는 함수를 구현하시오.
- 1-2. 주어진 패치에서 1) image gradient, 2) color histogram 디스크립터를 추출하는 함수를 각각 구현하시오.
- 1-3. 앞서 구한 1) image patch, 2) image gradient, 3) color histogram 디스크립터 특징을 이용하여, “paired images” 폴더에 있는 이미지들로 feature matching을 해보시오. 이 때, opencv의 built-in 함수(Brute Force Matcher와 같은)를 사용하시면 됩니다. 또한, cv2.drawMatches(...)를 이용하여 결과를 나타내어 보시오.
- 1-4. OpenCV의 함수를 사용해 SIFT와 같은 더 나은 디스크립터를 추출해 feature matching을 수행하고 결과를 그려보시오. 앞서 직접 구현한 3가지 디스크립터와 결과를 비교해보시오.

[문제 2. Homography and Stitching (30 pts)]

- 2-1. Homography 행렬 H를 1-4에서 feature matching을 통해 얻어진 매칭 포인트들을 이용해 구

해보시오. 이 때, `cv2.findHomography` 함수를 이용하고 method 값은 0(SVD 혹은 least-square 방식의 일반적 해법)으로 두어 모든 포인트들을 사용 하시오.

2-2. Homography 행렬 H 를 RANSAC을 사용해 구해보시오. 마찬가지로, `cv2.findHomography` 함수를 이용하시오.

2-3. 앞서 2-1과 2-2에서 구한 두 가지의 Homography 행렬을 이용해, “paired images” 폴더에 있는 이미지들로 image stitching을 해보시오. 이 때, `cv2.warpPerspective(...)` 함수를 사용하시오. 2-1과 2-2의 결과를 보이고, 차이에 대한 분석을 서술하시오.

[문제 3. Homography with RANSAC 직접 구현 (30 pts)]

3-1. RANSAC 알고리즘을 통한 Homography 행렬 H 를 구하는 함수를 직접 구현해 구해보시오.

이 때, `cv2.findHomography`와 같이 H 를 직접적으로 구해주는 Library를 사용하지 마세요. 다만, Numpy와 같이 중간 결과를 편하게 계산해주는 Library 사용은 가능합니다.

3-2. 앞서 3-1에서 구한 두 가지의 Homography 행렬을 이용해, “paired images” 폴더에 있는 이미지들로 image stitching을 해보시오. 이 때, 2-3과 동일하게, `cv2.warpPerspective(...)` 함수를 사용하시오. 또한, 해당 결과를 2-2 결과와 비교해 구현이 올바른지 검증해보시오.