

컴퓨터비전 프로젝트 #1

한경빈

Computer Vision Project #1

Han Gyeongbin

요 약

두 영상을 입력 받아 마우스 클릭으로 4개의 동일점을 저장한 후 각 영역의 그래디언트 히스토그램을 구하고, 히스토그램 차이를 계산하여 각 점 별로 가장 유사한 결과를 제시하는 실험을 수행한다.

Abstract

Storing 4 identical points of 2 images with a mouse click, get a gradient histogram of each area, then calculating the histogram difference and performing an experiment that presents the most similar results of each point.

Key words

Corner detection, Gradient histogram

I. 서 론

두 입력 영상(이미지)에서 동일 코너점을 찾기 위해, 마우스 클릭을 통해 이미지당 네 개의 점을 저장한 후 각각의 ROI에 대한 그래디언트 분포 히스토그램을 구하여 비교해 결과를 출력한다.

II. 문제 제기

선택한 점을 중심으로 한 사각형을 ROI로 지정해 히스토그램을 구하는 방법은 다소 정확도가 떨어질 수 있는데, 이미지의 물체가 회전되어 있는 경우 동일점을 선택하였음에도 불구하고 ROI로 지정된 영역에 큰 차이를 보일 수 있다.

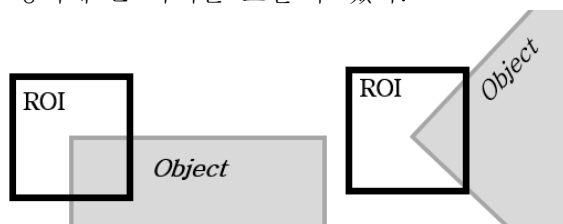


그림 1. 회전도에 따른 ROI 영역 차이

위 그림은 같은 코너점을 선택했음에도 불구하고 사각 모양의 ROI가 선택됨으로 인해 결과 히스토그램이 매우 다르게 추출될 것으로 예상된다.

III. ROI 설정 방법

이러한 문제를 해결하기 위하여, 본 프로젝트에서는 ROI에서 히스토그램을 구할 때에 정사각 모양의 ROI에 내접하는 원에 포함되는 픽셀로만 계산하도록 한다.

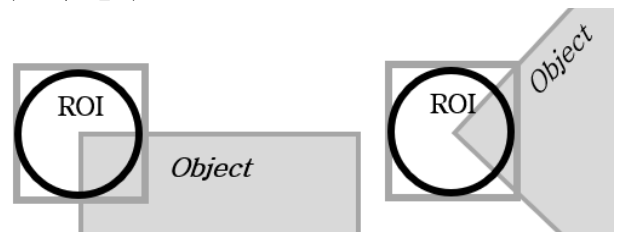


그림 2. 새롭게 설정된 ROI 영역

위 그림처럼 원 모양의 ROI를 설정한다면, 물체의 회전도에 상관없이 동일점에 대하여 같은 히스토그램을 얻을 수 있다.

IV. 실험 내용

프로그램을 실행하면 흑백으로 변환한 두 이미지를 이어 붙여 출력하고, 마우스 클릭으로 8개 점 위치를 입력 받는다.

```
image0 point0
342 184
image0 point1
33 477
image0 point2
432 905
image0 point3
746 605
image1 point0
1399 276
image1 point1
951 276
image1 point2
923 891
image1 point3
1394 912
image1 point4
1394 912
```

그림 3. 콘솔에 출력된 각 점의 포지션

모든 점을 입력 받고 각 점을 기준으로 생성된 ROI에 대해 그래디언트 히스토그램을 계산한다.

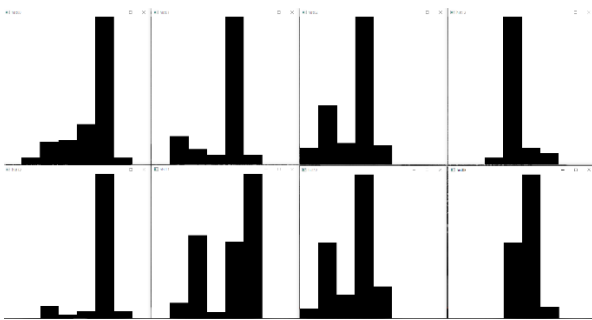


그림 4. 각 점에 대한 히스토그램

위의 히스토그램의 값들을 모두 비교하고 차이를 더하여 가장 비슷한 점을 매칭시킨다.

```
first:0, second:0, diff:211
first:0, second:1, diff:476
first:0, second:2, diff:1294
first:0, second:3, diff:1082
0 similar with 0
first:1, second:0, diff:1088
first:1, second:1, diff:995
first:1, second:2, diff:1166
first:1, second:3, diff:1069
1 similar with 1
first:2, second:0, diff:1337
first:2, second:1, diff:1627
first:2, second:2, diff:156
first:2, second:3, diff:364
2 similar with 2
first:3, second:0, diff:1280
first:3, second:1, diff:1291
first:3, second:2, diff:1088
first:3, second:3, diff:736
3 similar with 3
```

그림 5. 콘솔에 출력된 각 히스토그램의 차와 매칭된 결과

V. 실험 결과

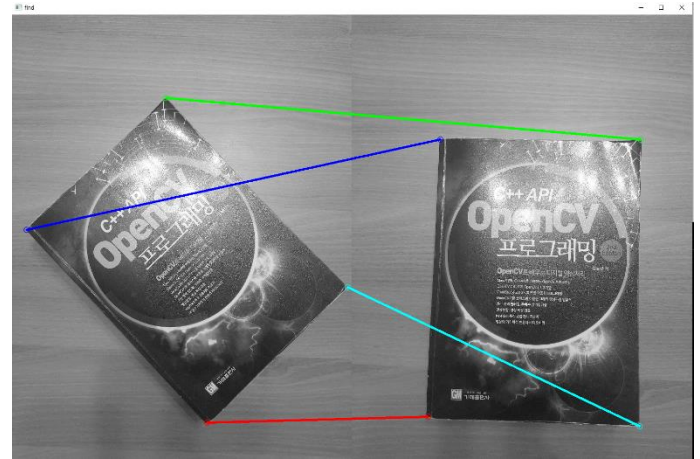


그림 6. 가장 비슷한 코너점을 연결한 결과

원 모양의 ROI에 대해, 각 코너점과 가장 비슷한 코너점을 서로 매칭시킨 결과로, 정확하게 서로 매칭된다.

VI. 결론

사각 모양의 ROI에 대해서도 시도해본 결과, 위의 원 모양의 ROI로 구한 결과에 비해 틀린 결과가 더 많이 도출되었다. 즉 본 프로젝트에서 정한 ROI 설정 방법이 효과가 있었음을 확인하였다.

VII. 한 계

본 프로젝트에서는 단순히 두 이미지를 흑백으로 변환하여 사용하기 때문에, 조명이 다르거나, 장애물이 물체를 가리는 상황을 고려할 수 없다. 이러한 부분을 고려하기 위해서는 추가적인 전처리를 통해 두 이미지의 특성을 균일하게 맞추는 과정이 필요할 것이다.

VIII. 부 록

깃허브 주소

<https://github.com/hanabzu/Computer-Vision-Project-01>

코드는 위 프로젝트의 main.cpp 파일에 작성함.