**OSP: Lec7-Edge and Corner Detection**

1871098 손수민

1. Laplacian of Gaussian

Laplacian of Gaussian을 구현해 edge를 찾아낸 이미지를 보여주는 예제이다.

우선 가우시안 필터를 적용하여 blur 효과를 줘 노이즈를 제거해주는 과정을 거쳐야한다. Mirroring 함수를 통해 input\_mirror에는 row+2\*n, col+2\*n 사이즈를 가지는 mirroring된 이미지가 반환된다. 이를 4개의 for문을 통해 중심 픽셀 (I,j)로 하는 주변 -n~n 픽셀들에 접근한다. 그 후 input\_mirror 이미지에 kernel value에 해당하는 kernel.at<G>(a + n, b + n)을 곱해줘 output에 저장해준다.

텍스트이(가) 표시된 사진

자동 생성된 설명

이렇게 가우시안 필터를 거친 이미지는 다시 라플라시안 필터로 가게 된다. 앞선 가우시안 필터와 마찬가지로 똑 같은 형태를 띄지만 kernel에서 get\_Laplacian\_kernel로 라플라시안 커널을 가져오는 것이 차이점이다.

텍스트이(가) 표시된 사진

자동 생성된 설명텍스트이(가) 표시된 사진

자동 생성된 설명

Main 함수에서는 위와 같이 이미지를 회색이미지로 변환한 후 가우시안필터를 통과한 h\_f를 다시 라플라시안 필터로 넣어 normalized 과정을 거친다.

위의 결과는 다음과 같이 도출된다.

텍스트이(가) 표시된 사진

자동 생성된 설명

1. Canny Edge Detector

다음은 opencv의 canny를 이용하여 canny edge detector를 구현하는 예제이다.

테이블이(가) 표시된 사진

자동 생성된 설명

이미지를 회색이미지로 변환한 후 이를 canny에 넣고 반환될 output도 넣어준다. 그리고 threshold1은 80, threshold2는 160으로 설정해주었다.

텍스트이(가) 표시된 사진

자동 생성된 설명

결과는 다음과 같이 나왔다.

텍스트, 하얀색이(가) 표시된 사진

자동 생성된 설명

1. Harris Corner Detector

다음은 opencv의 cornerHarris를 이용해 이미지의 코너를 찾아내는 예제이다.

<https://docs.opencv.org/3.1.0/d4/d7d/tutorial_harris_detector.html>

를 참고하여 cornerHarris의 파라미터를 설정해주었다.

(blocksize=2, apertureSize=3, k=0.04)

텍스트이(가) 표시된 사진

자동 생성된 설명

다음은 Subpixel = true일 때, 실행되는 부분으로 탐지된 corners에 대해 정제를 해주는 과정이다. 여기서는 또한 opencv의 cornerSubPix를 사용해준다.

<https://docs.opencv.org/3.4.1/d2/d0a/lkdemo_8cpp-example.html>

위의 링크의 글을 참고하여 작성해주었다.

텍스트, 실내, 장치이(가) 표시된 사진

자동 생성된 설명

다음은 nonmaximum\_suppression 함수이다.

Non-maximum suppression은 small window 안에서의 가장 큰 edge magnitude를 가지는 only pixels만 살리는 것이 목적이다. 위와 똑같이 Mirroring을 사용해 input\_mirror 이미지를 받아온다. 이 후 이중 for문을 돌며 (i,j)를 중심으로하는 small window(radius가 사이즈)를 안의 이중 for문으로 접근하는 구조이다.

만약 corner\_mat.at<uchar>(i - radius, j - radius) == 1 이면 코너로 표시된 픽셀이므로, 이 경우에만 해당 과정을 거쳐준다.

위의 과정을 거쳐 window 안의 픽셀을 탐색하는 과정에서 max가 되어야하는 center 점 (i,j) 보다 큰 픽셀이 존재한다면 해당 점은 더 이상 corner 점이 되지 않으므로 0으로 세팅해주고 for문을 빠져나온다.

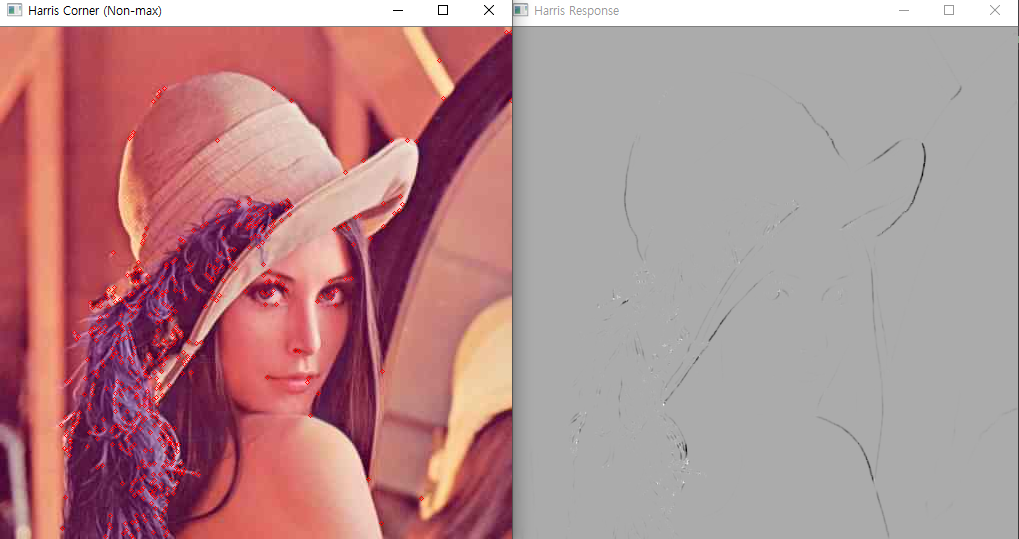
텍스트이(가) 표시된 사진

자동 생성된 설명

다음은 위의 결과값으로 도출된 결과이다. Non-maximum suppression을 거친 결과 corner에 해당하는 픽셀들의 개수가 줄어들고 좀 더 정제된 것을 볼 수 있다.

텍스트, 의류, 머리카락, 입은이(가) 표시된 사진

자동 생성된 설명



텍스트이(가) 표시된 사진

자동 생성된 설명