

메디치소프트 기술연구소

CH15. OpenCV: Canny Edge Detection

15. Canny Edge Detection

- 15.1 테스트 환경

순 번	제 목	설치 버전
1	운영 체 제	Windows 10 64bit
2	프로그래밍 언어	Python3.6.5

15. Canny Edge Detection

- 15.2 Canny Edge Detection 이란?

- Canny Edge Detection은 가장 인기있는 에지 찾기 알고리즘 중 하나이다. 이 알고리즘은 1986년 John F Canny 라는 사람에 의해 개발되었다. Canny Edge Detection은 아래와 같은 다단계 알고리즘으로 구성되어 있다.

- Canny Edge Detection 과정

(1) 1단계 : 노이즈 제거 (Noise Reduction)

이미지에서 노이즈가 있으면 에지를 제대로 찾는 것이 어려울 수 있다. 따라서 첫단계로 5x5 가우시안 필터(Gaussian Filter)를 이용해 이미지의 노이즈를 줄여줍니다.

(2) 2단계 : Gradient 값이 높은 부분 찾기

가우시안 필터로 노이즈가 제거된 이미지를 앞서 배웠던 Sobel 커널을 수평방향, 수직방향으로 적용하여 각 방향의 gradient를 획득한다. 픽셀 (x, y)에서 edge gradient는 아래의 식으로 찾을 수 있다.

$$\begin{aligned} \text{Edge_Gradient } (G) &= \sqrt{G_x^2 + G_y^2} \\ \text{Angle } (\theta) &= \tan^{-1} \left(\frac{G_y}{G_x} \right) \end{aligned}$$

[그림] edge gradient 수식

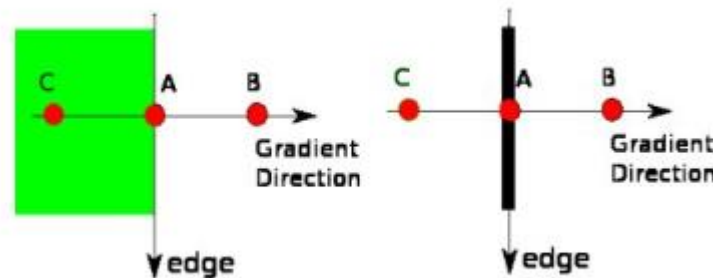
15. Canny Edge Detection

▪ 15.2 Canny Edge Detection 이란?

- Canny Edge Detection 과정

(3) 3단계 : 최대값이 아닌 픽셀의 값을 0으로 만들기

2단계를 거친 후, 에지에 기여하지 아니한 픽셀을 제거하기 위해 이미지 전체를 스캔한다. 이미지를 스캔하는 동안 gradient 방향으로 스캔 구역에서 gradient의 최대값을 가진 픽셀을 찾는다. 아래 그림에서 A는 수직방향의 에지위에 있는 픽셀이다. Gradient 방향은 수평방향이다. B와 C는 Gradient 방향에 놓인 픽셀이다. A지점에서 gradient 값이 B, C보다 값이 큰지 확인한다. A에서의 값이 가장 크면 그냥 다음 단계로 넘어가고 그렇지 않으면 값을 0으로 만든다.



[그림] 픽셀처리과정

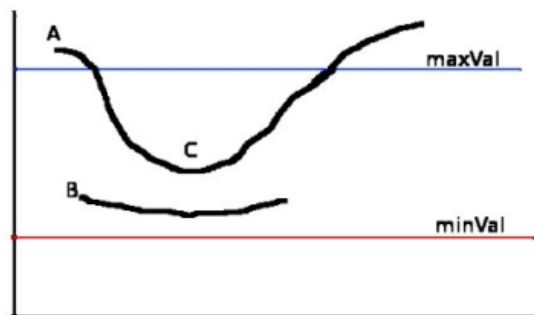
15. Canny Edge Detection

▪ 15.2 Canny Edge Detection 이란?

- Canny Edge Detection 과정

(4) 4단계 : Hyteresis Thresholding

4단계는 3단계를 거친 것들이 실제 에지인지 아닌지 판단하는 단계이다. 먼저 문턱값을 minVal , maxVal 2개 잡는다. maxVal 보다 높은 부분은 확실한 에지이고 minVal 보다 낮은 부분은 에지가 아니라고 판단한다. minVal , maxVal 사이에 있는 값들은 이 픽셀들의 연결구조를 보고 에지인지 아닌지 판단한다. A는 maxVal 보다 위에 있으므로 확실한 에지이다. B와 C는 minVal , maxVal 사이에 있는 픽셀이다. 여기서 B는 확실한 에지와 연결되어 있지 않지만, C는 확실한 에지와 연결이 되어 있다. 따라서 B는 에지가 아니므로 제거하고 C는 에지이므로 남겨둔다.



[그림] 에지 판단

15. Canny Edge Detection

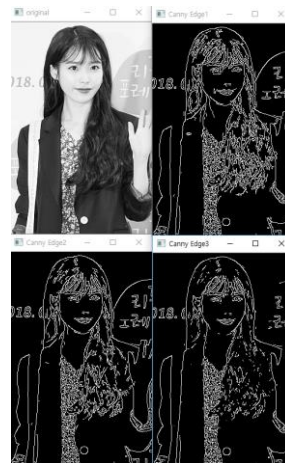
- 15.3 [15-01 example.py] – Canny Edge Detection (에지 찾기 알고리즘)
 - 다음은 15-01 example.py 예제의 code이다.

[그림] 06-01 example.py의 code (1/3)

[그림] 06-01 example.py의 code (2/3)

15. Canny Edge Detection

- 15.3 [15-01 example.py] – Canny Edge Detection (에지 찾기 알고리즘)
 - 다음은 15-01 example.py 예제의 실행 결과로 나오는 화면이다. 뒷장 부터 15-01 example.py code의 주요 함수를 line 순서대로 분석해보도록 한다.



[그림] 15-01 example.py의 실행 결과

15. Canny Edge Detection

- 15.3 [15-01 example.py] – Canny Edge Detection (에지 찾기 알고리즘)

```
>> LINE 7) edge1 = cv2.Canny(img, 50, 200)
```

```
>> LINE 8) edge2 = cv2.Canny(img, 100, 200)
```

```
>> LINE 9) edge3 = cv2.Canny(img, 170, 200)
```

- cv2.Canny() 함수는 여태 설명했던 Canny Edge Detection 알고리즘을 구현한 함수이다.
이 함수의 인자는 다음과 같다.

(1) img: Canny Edge Detection을 수행할 원본 Grayscale 이미지

(2) 50: minimum thresholding value

(3) 200: maximum thresholding value


```
import numpy as np
import cv2
import matplotlib.pyplot as plt

def canny():
    img = cv2.imread('images/10.jpg', cv2.IMREAD_GRAYSCALE)
    edge1 = cv2.Canny(img, 50, 200)
    edge2 = cv2.Canny(img, 100, 200)
    edge3 = cv2.Canny(img, 170, 200)

    cv2.imshow('original', img)
    cv2.imshow('Canny Edge1', edge1)
    cv2.imshow('Canny Edge2', edge2)
    cv2.imshow('Canny Edge3', edge3)
    cv2.waitKey(0)
    cv2.destroyAllWindows()

canny()
```