**HW 02 – REPORT**

소속 : 정보컴퓨터공학부

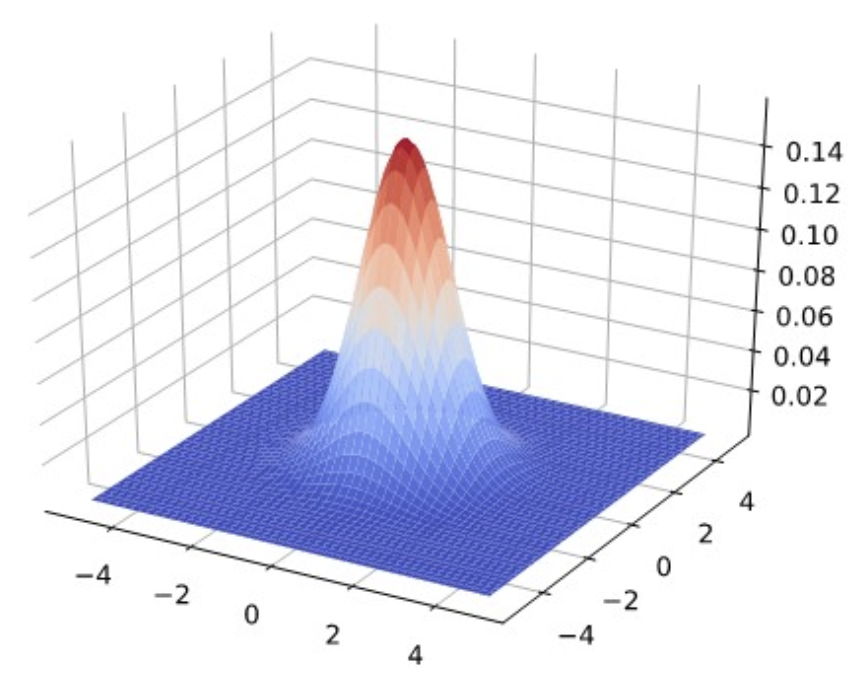
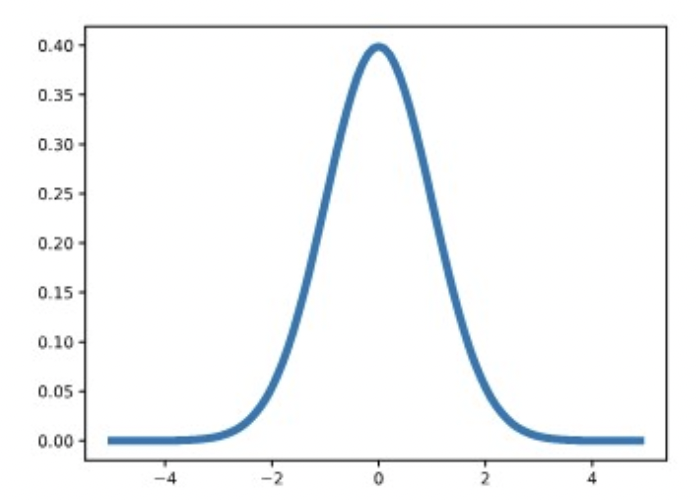
학번 : 201824633

이름 : 김유진

1. 서론

실습 목표 및 이론적 배경 기술 (1~2페이지)

Gaussian filter에 대한 이해도가 필요하다. 가우시안 필터는 근처에 있는 픽셀들에게 더 많은 가중치를 주고 계산을 한다. 1차원과 2차원 가우시안 필터를 그래프로 나타내면 다음과 같은 형태를 가진다.



<그림 1-1.> 1차원 가우시안 필터 <그림 1-2.> 2차원 가우시안 필터

5 x 5 형태이고, 감마의 값이 1일 때 가우시안 필터는 다음과 같은 값과 픽셀의 형태를 가진다.

테이블이(가) 표시된 사진

자동 생성된 설명광장이(가) 표시된 사진

자동 생성된 설명

<그림 2.> 5x5, 감마 = 1일때 가우시안 필터와 픽셀의 형태

이 때, 가우시안 필터는 모든 테이블의 합이 1으로 일반화되었다.

2. 본론

Part 1

1. boxfilter(n)

: n x n의 박스 필터를 리턴하는 함수이다. n은 홀수여야하고, 에러는 assert 를 이용해야한다. 필터는 numpy여야한다.

|  |
| --- |
| # 1. make 2D filters : odd number of rows and columns  def boxfilter(n):  # using assert statement  assert n % 2 == 1, "Dimension must be odd"  # filter should be a numpy array  filter = np.full((n, n), 1 / (n\*\*2))  return filter |

Show the results of your boxfilter(n) function for the cases n = 3, n = 4, n =7

n = 4일때, error가 발생하기 때문에, 3, 7, 4 순서대로 실행시켜줬다.

텍스트이(가) 표시된 사진

자동 생성된 설명

1. gauss1d(sigma)

: 1차원 가우시안 필터를 만들어 리턴해주는 함수이다. 매개변수로 sigma를 받아 가우시안 함수에 적용시킨다.

|  |
| --- |
| # 2. 1D Gaussian filter  def gauss1d(sigma):  # length : sigma \* 6, and then rounded up to the next odd integer  length = math.ceil(sigma \* 6) // 2 \* 2 + 1  # generate 1D array of x values  x = np.arange(length // 2 \* (-1), length // 2 + 1)  # x is the ditance of an array value from the center, center is 0  x = np.abs(np.subtract(x, 0))  # apply function  filter = np.exp(x \*\* 2 \* -1 / (2 \* (sigma \*\* 2)))  # normalize the values in the filter : sum to 1  sum\_of\_filter = np.sum(filter)  filter = np.divide(filter, sum\_of\_filter)  return filter |

텍스트이(가) 표시된 사진

자동 생성된 설명

1. gauss2d(sigma)

: 2차원 가우시안 필터를 리턴해주는 함수이다. 위의 gauss1d 함수를 이용한다.

|  |
| --- |
| # 3. 2D Gaussian filter  def gauss2d(sigma):  # user np.outer with the 1d array from the function gauss1d(sigma)  filter = gauss1d(sigma)  filter = np.outer(filter, filter)  # normalize the values in the filter : sum to 1  sum\_of\_filter = np.sum(filter)  filter = np.divide(filter, sum\_of\_filter)  return filter |

텍스트, 신문, 영수증이(가) 표시된 사진

자동 생성된 설명

1. dog.bmp 에 filter 적용하기
   1. convolve2d(array, filter)

: 상하좌우에 padding을 더해준 뒤, convolution을 해주는 함수를 작성한다.

|  |
| --- |
| # 4-a. zero paddings  def convolve2d(array, filter):  # f x f kernel, m = (f-1)/2 : m space < fill with zeros  f = len(filter)  m = (f-1) // 2  # add zero padding  array\_padding = np.pad(array, ((m, m), (m, m)), 'constant', constant\_values=0)  # for convolution : up, down, left and right changes  for\_convolution = np.flip(filter)  # array : numpy array, so no len() => asarray => array  array\_numpy = np.asarray(array)  # for result : len  row, col = len(array\_numpy), len(array\_numpy[0])  # result => save to 'image' numpy array variable.  image = np.zeros((row, col), dtype=np.float32)  # performs convolution to the image with zero paddings  for i in range(row):  for j in range(col):  image[i][j] = np.sum(array\_padding[i:i+len(filter), j:j+len(filter)] \* for\_convolution)  return image |

* 1. gaussconvolve2d(array, sigma)

: 가우시안 컨벌루션을 2차원 배열로 적용한다. 앞에서 만든 gauss2d 함수와 convolve2d 함수를 이용한다.

|  |
| --- |
| # 4-b. Gaussian convolution to a 2D array  def gaussconvolve2d(array, sigma):  # generating a filter with gauss2d  filter = gauss2d(sigma)  # apply it to the array with convolve2d  padding = convolve2d(array, filter)  return padding |

* 1. 예시를 만들어보자. dog.bmp 파일을 이용한다.

|  |
| --- |
| # 4-c. Apply gaussconvolve2d  # Load Dog Image into Python  dog = Image.open('[HW02] Image Filtering/hw2\_image/2b\_dog.bmp')  # Convert it to a greyscale  grey\_dog = dog.convert('L')  # for calculate : numpy -> array, input variable type : np.float32  grey\_dog\_array = np.asarray(grey\_dog, dtype=np.float32)  # apply filter  grey\_dog\_array = gaussconvolve2d(grey\_dog\_array, 3)  # for PIL show  grey\_dog = Image.fromarray(grey\_dog\_array) |

* 1. PIL 모듈을 이용해 original 이미지와 가우시안 필터가 적용된 이미지를 show 한다.

|  |
| --- |
| # 4-d. Use PIL to show both the original and filtered images.  dog.show()  grey\_dog.show() |

개, 가장, 실내, 갈색이(가) 표시된 사진

자동 생성된 설명개, 실내, 앉아있는, 보는이(가) 표시된 사진

자동 생성된 설명

<그림 5-a.> 원본 dog.bmp(좌) <그림 5-b.> greyscale + gaussian blur => dog.bmp(우)

3. 결론

토의 및 결론 (1페이지)