

# 컴퓨터 비전

[HW04]

학번	201203393
분반	00
이름	김현겸
과제번호	04

제출일 : 2016년 4월 10일 일요일

1. (MATLAB) (1) zero-padding or (2) mirror padding을 이용한 uniform한 mean filter를 직접 구현하시오.

```
function result = filter2_revision(filter, x, opt)

mask_size = size(filter);
x_size = size(x);
wr = floor(mask_size(1)/2);
wc = floor(mask_size(2)/2);

if strcmp(opt, 'zero')
    m_x = zeros(x_size(1) + mask_size(1) - 2, x_size(2) + mask_size(2) - 2);
    m_x(wr : 1 : x_size(1) + wr - 1, wc : 1 : x_size(2) + wc - 1) = x;

else if strcmp(opt, 'mirror')
    m_x = [x(wr:-1:1,:); x; x(end:-1:end-(wr-1),:)];
    m_x = [m_x(:, wc:-1:1), m_x, m_x(:, end:-1:end-(wc-1))];
end

end

result = filter2(filter, m_x)
```

기존에 작성되어 있는 filter2 및 책을 참고하여 zero와 mirror 기능을 구현한다. zero의 경우, image보다 사이즈가 큰 zero 매트릭스를 만들고 그 안에 filter된 데이터를 넣는 방법으로 구현하고, mirror의 경우 양 끝의 데이터들을 한층 확장하는 방법으로 만들면 된다.

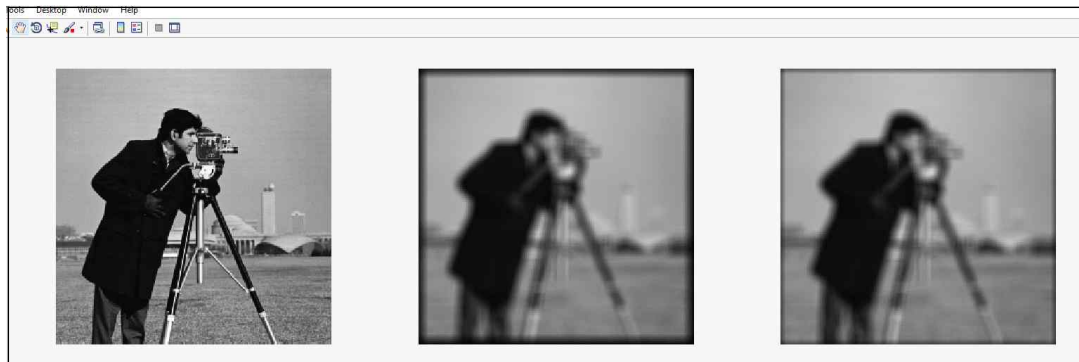
이때, 주의 할 것은 filter2 함수를 사용하는 과정에서 valid 옵션을 주었을 때, 이미지의 크기에만 맞게 나타나기 때문에 padding이 적용되었는지 확인하기 힘들다. 따라서 아무런 옵션 없이 사용한다.

```
function hw_1()

x = imread('cameraman.tif');
s = size(x);
filter = 1/121 * ones(11,11);

result = filter2_revision(filter, x, 'zero')
result2 = filter2_revision(filter, x, 'mirror')
figure,
subplot(1,3,1), imshow(x)
subplot(1,3,2), imshow(uint8(result))
subplot(1,3,3), imshow(uint8(result2))
```

filter2\_revision함수를 사용할 때, filter로는 11 x 11 mean 매트릭스를 만들어서 모든 값이 1/121을 갖도록 만든 다음 적용한다. 이렇게 적용한 데이터를 위처럼 작성하여 코드를 테스트 해보면 다음과 같다.



2. (MATLAB) Gaussian filtering in 2D image를 구현하시오. (separable way)

Gaussian filter

$$f(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

위와 같은 Gaussian 함수를 x와 y에 나누어서 적용한다.

```
function result = dim1_gaussian(filter, sigma)

[x] = -floor(filter/2) : floor(filter/2);

on_filter = (1/(sqrt(2*pi)*sigma))*exp(-(x.^2) / (2*(sigma^2)));

on_filter = on_filter / sum(on_filter(:));

result = on_filter.';
```

우선 x 한 방향에 대해서 gaussian 함수를 적용하는 함수를 작성한다. filter의 크기가 filter라는 이름의 변수로 주어지면 이를 정 가운데로 나눈 범위를 가지는 x를 만들고 각각의 데이터에 대해서 식을 적용한 다음, 그 값을 모두 합친다. 그리고 transposition하여 반환한다.

```
function hw_2()

im = imread('cameraman.tif');

filter_x = dim1_gaussian(5, 2);
filter_o = filter2(filter_x, im);

filter_result = filter2(filter_x', filter_o);

figure,
subplot(1,2,1), imshow(uint8(filter_result))
subplot(1,2,2), imshow(uint8(filter2(fspecial('gaussian', 5, 2), im)))
```

나머지 결과는 hw\_2를 통해서 완성한다. dim1\_gaussian을 통해 얻은 결과를 filter2 함수를 통해서 결과를 얻어낸 뒤, 두 결과물을 가지고 다시 filter2 함수를 사용하여 최종적으로 x축과 y축에 대해서 gaussian 필터를 적용한 결과를 얻는다.

이렇게 만들어진 결과와 fspecial 함수를 통해 얻을 수 있는 필터 결과를 비교해보면 다음과 같다.



### 3. (MATLAB) Exercise 1-b, d, e, g

```
function hw_3()

x = [20,20,20,10,10,10,10,10,10;
     20,20,20,20,20,20,20,20,10;
     20,20,20,10,10,10,10,20,10;
     20,20,10,10,10,10,10,20,10;
     20,10,10,10,10,10,10,20,10;
     10,10,10,10,20,10,10,20,10;
     10,10,10,10,10,10,10,10,10;
     20,10,20,20,10,10,10,20,20;
     20,10,10,20,10,10,20,10,20];

filter_b = [0,-1,-1; 1,0,-1; 1,1,0];
filter_d = [-1,-2,-1; -1,2,-1; -1,2,-1];
filter_e = [-1,-1,-1; -1,8,-1; -1,-1,-1];
filter_g = [-1,0,1; -1,0,1; -1,0,1];

figure,
subplot(2,2,1), imshow(uint8(filter2_revision(filter_b, x, 'zero')));
subplot(2,2,2), imshow(uint8(filter2_revision(filter_d, x, 'zero')));
subplot(2,2,3), imshow(uint8(filter2_revision(filter_e, x, 'zero')));
subplot(2,2,4), imshow(uint8(filter2_revision(filter_g, x, 'zero')));
```

책에서 주어지는 매트릭스를 토대로 각 filter 값을 만들어 filter2\_revision 함수를 적용시키면 아래와 같은 결과를 얻을 수 있다.

```

result =

    0    40    50    50    40    40    40    40    40    20
   -40    0    10    10    0    0    0    20    40    20
   -40    0    0   -10   -20   -20   -30    0    40    20
   -40    0    0    0    0    0   -20    0    40    20
   -40    0    0    0    10   10   -20    0    40    20
   -30    0    0   -10    0   10   -20   -10    30    20
   -10   10   10   10    0    0   -10    0    40    30
   -10   10   -10   20   20    0    0    0    40    40
   -40  -20   -50   -30   -10   -30   -30   -40   -10    20
   -30  -20   -30   -30   -20   -30   -30   -30   -20    0

result =

    40    0    10   -10    0    0    0    10    10   -20
   -20   -80   -60   -60   -40   -40   -50   -10   -30   -30
   -20   -70   -80   -90   -80   -80  -100   -30   -40   -30
   -10   -80   -80   -50   -40   -40   -70   -20   -40   -30
   -20   -80   -50   -50   -20   -50   -70   -20   -40   -30
   -30   -50   -40   -50   -20   -50   -60   -40   -30   -30
    10   -60   -30   -40   -70   -50   -60   -50   -10   -40
    30   -70   -40   -10   -60   -50   -30   -50    20   -50
   -20   -70   -80   -50   -60   -50   -30   -90   -30   -40
   -50   -50   -50   -60   -50   -50   -60   -60   -50   -20

result =

   100    60    70   -10    0    0    0    10    40   -20
    60    0    20    40    60    60    50    60    10   -30
    60    10    30   -40   -30   -30   -50    50    0   -30
    70    30   -30   -10    0    0   -30    60    0   -30
    90   -30   -10   -10   -10   -10   -30    60    0   -30
    20   -10    0   -10    80   -10   -20    70    10   -30
    20   -20   -20   -30   -20   -10   -20   -30    0   -40
   100   -30    60    60   -20   -10   -20    50    90   -50
   120    0    0    90    10    20   100   -10   110   -40
   -30   -40   -40   -40   -40   -40   -40   -50   -30   -20

result =

    40    0   -10   -10    0    0    0   -10   -30   -20
    60    0   -20   -20    0    0   10   -10   -50   -30
    60   -10   -20   -10    0    0   20   -10   -60   -30
    50   -20   -20   -10    0    0   30    0   -60   -30
    40   -20   -10   10    0   -10   30    0   -60   -30
    30   -10    0   10    0   -10   20    0   -50   -30
    30    0   10    0   -10   -10   20   10   -50   -40
    30   -10   20   -10   -20   10   10   10   -40   -50
    20   -10   20   -10   -20   10   10   10   -30   -40
    10   -10   10    0   -10   10    0    0   -10   -20

```

x의 데이터들이 각 필터의 값에 따라서 변경이 되었음을 알 수 있다.

4. (Report) Exercise 13

$$4. \left[ \frac{1}{9}(a+b+c+f+g+h+k+l+m), \frac{1}{9}(b+c+d+g+h+i+l+m+n), \right. \\ \frac{1}{9}(c+d+e+h+i+j+m+n+o); \frac{1}{9}(f+g+h+k+l+n+p+q+r), \\ \frac{1}{9}(g+h+i+l+m+n+q+r+s), \frac{1}{9}(h+i+j+m+n+o+r+s+t); \\ \frac{1}{9}(k+l+m+p+q+r+u+v+w), \frac{1}{9}(l+m+n+q+r+s+v+w+x), \\ \left. \frac{1}{9}(m+n+o+r+s+t+w+x+y) \right]$$

각 값은 해당문자의 prime으로 두고, 3x3을 다시 적용하면 정 가운데의 m에 대한 1x1 한칸만 남는다. 여기서 각문자마다 사용되는 횟수는 다음과 같다.

$$\begin{bmatrix} a & b & c & d & e \\ f & g & h & i & j \\ k & l & m & n & o \\ p & q & r & s & t \\ u & v & w & x & y \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 4 & 6 & 4 & 2 \\ 3 & 6 & 9 & 6 & 3 \\ 2 & 4 & 6 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$$

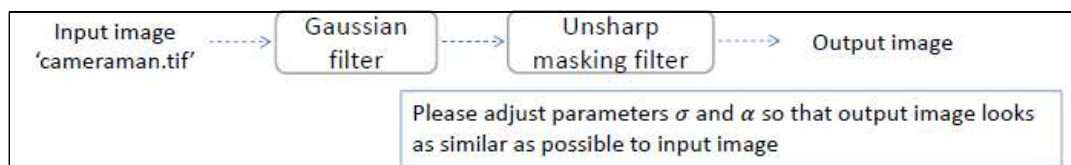
따라서 위의 배열에 맞게 맞출 수 있는 5x5 필터를 만들면 된다.

$$\frac{1}{81} \begin{bmatrix} 1 & 2 & 3 & 2 & 1 \\ 2 & 4 & 6 & 4 & 2 \\ 3 & 6 & 9 & 6 & 3 \\ 2 & 4 & 6 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$$

그리고 이 필터 또한 분리가 가능하다.

$$\frac{1}{81} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 2 \\ 1 \end{bmatrix} [1 \ 2 \ 3 \ 2 \ 1]$$

5. (MATLAB) (1)가우시안 필터(7 x 7 window with  $\sigma$ )를 적용시키고, unsharp masking 필터(with  $\alpha$ )를 적용시키는 함수를 직접 구현하시오.



```
function f2 = unsharp_filter()

f = fspecial('average');

k = 0.3;
fi = zeros(3); fi(2,2)=1;
f2 = (fi - k*f) / (1-k);
```

ppt에 나와 있는 unsharp\_filter 함수를 우선 작성한다. average 하게 fspecial 함수를 사용한 f에 대해서 위와 같은 식으로 unsharp\_filter를 만든다.

```
function hw_5()

im = imread('cameraman.tif');

filter_x = dim1_gaussian(7, 0.5);
filter_o = filter2(filter_x, im);

filter_result = filter2(filter_x', filter_o);

figure, imshow(uint8(filter2(unsharp_filter(), filter_result)))
```

그리고 위처럼 gaussian 필터를 적용하여 얻어낸 결과에 마지막으로 unsharp\_filter를 사용하여 filter2 함수의 결과를 얻어내면 아래와 같다.



6. (Report) convolution의 결과를 계산하시오.

6.  $n$ 에 대한 범위는  $-5 \sim 7$ 까지 계산해 볼 수 있다.

( $n-m$ 의 최소범위가  $-3$ 이고,  $m$ 의 최소가  $-2$ 이므로  $n=-5$ ,  
 $n-m$ 의 최대범위가  $3$ 이고,  $m$ 의 최대가  $4$ 이므로  $n=7$ 이다)

$$Y[-5] = 1 \times 2 = 2$$

$$Y[-4] = 1 \times 3 + 2 \times 2 = 7$$

$$Y[-3] = 1 \times 3 + 2 \times 3 + 2 \times 3 = 15$$

$$Y[-2] = 1 \times 3 + 2 \times 3 + 3 \times 3 + 4 \times 2 = 9 + 9 + 8 = 26$$

$$Y[-1] = 1 \times 2 + 2 \times 3 + 3 \times 3 + 4 \times 3 + 3 \times 2 = 8 + 9 + 12 + 6 = 35$$

$$Y[0] = 1 \times 3 + 2 \times 2 + 3 \times 3 + 4 \times 3 + 3 \times 3 + 2 \times 2 = 7 + 9 + 12 + 9 + 4 = 41$$

$$Y[1] = 1 \times 3 + 2 \times 3 + 3 \times 2 + 4 \times 3 + 3 \times 3 + 2 \times 3 + 1 \times 2 = 9 + 6 + 12 + 9 + 6 + 2 = 44$$

$$Y[2] = 2 \times 3 + 3 \times 3 + 4 \times 2 + 3 \times 3 + 2 \times 3 + 1 \times 2 = 6 + 9 + 8 + 9 + 6 + 3 = 41$$

$$Y[3] = 3 \times 3 + 4 \times 3 + 3 \times 2 + 2 \times 3 + 1 \times 3 = 9 + 12 + 6 + 6 + 3 = 36$$

$$Y[4] = 4 \times 3 + 3 \times 3 + 2 \times 2 + 1 \times 3 = 12 + 9 + 4 + 3 = 28$$

$$Y[5] = 3 \times 3 + 2 \times 3 + 1 \times 2 = 9 + 6 + 2 = 17$$

$$Y[6] = 2 \times 3 + 1 \times 3 = 9$$

$$Y[7] = 1 \times 3 = 3$$

