

5 Neighborhood Processing

소프트웨어융합 김태민

연습문제 1

원본

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

0으로 감싸기

0 0 0 0 0 0 0 0 0
0 20 20 20 10 10 10 10 10 0
0 20 20 20 20 20 20 20 10 0
0 20 20 20 10 10 10 10 20 10 0
0 20 20 10 10 10 10 10 20 10 0
0 20 10 10 10 10 10 10 20 10 0
0 10 10 10 10 20 10 10 20 10 0
0 10 10 10 10 10 10 10 10 10 0
0 20 10 20 20 10 10 10 20 20 0
0 20 10 10 20 10 10 20 10 20 0
0 0 0 0 0 0 0 0 0

a 마스크

a = [-1 -1 0;
 -1 0 1;
 0 1 1;]

원본 이미지에 a 마스크 적용 결과

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

$0 \cdot (-1) + 0 \cdot (-1) + 0 \cdot 0 +$
 $0 \cdot (-1) + 20 \cdot 0 + 20 \cdot 1 +$
 $0 \cdot 0 + 20 \cdot 1 + 20 \cdot 1 =$
 $0 + 0 + 0 + 0 + 0 + 20 + 0 + 20 + 20 =$
60

b 마스크

b = [0 -1 -1;
 1 0 -1;
 1 1 0;]

원본 이미지에 b 마스크 적용 결과

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

$0 \cdot 0 + 0 \cdot (-1) + 0 \cdot (-1) +$
 $0 \cdot 1 + 20 \cdot 0 + 20 \cdot (-1) +$
 $0 \cdot 1 + 20 \cdot 1 + 20 \cdot 0 =$
 $0 + 0 + 0 + 0 + 0 - 20 + 0 + 20 + 0 = 0$

d 마스크

```
d = [-1 2 -1;  
      -1 2 -1;  
      -1 2 -1;]
```

원본 이미지에 d 마스크 적용 결과

```
40 0 10 -10 0 0 0 10 10  
60 0 20 -20 0 0 -10 30 10  
60 10 0 -10 0 0 -20 50 0  
70 0 0 -10 0 0 -30 60 0  
60 0 -10 -10 20 -10 -30 60 0  
50 -10 0 -10 20 -10 -20 40 10  
50 -20 10 0 10 -10 -20 30 30  
70 -30 0 30 -20 -10 10 -10 60  
60 -30 0 30 -20 -10 10 -10 50
```

$$\begin{aligned} &0*(-1) + 0*2 + 0*(-1) + \\ &0*(-1) + 20*2 + 20*(-1) + \\ &0*(-1) + 20*2 + 20*(-1) = \\ &0 + 0 + 0 + 0 + 40 - 20 + 0 + 40 - 20 \\ &= 40 \end{aligned}$$

e 마스크

```
e = [-1 -1 -1;  
      -1 8 -1;  
      -1 -1 -1;]
```

원본 이미지에 e 마스크 적용 결과

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

$$\begin{aligned} &0*(-1) + 0*(-1) + 0*(-1) + \\ &0*(-1) + 20*8 + 20*(-1) + \\ &0*(-1) + 20*(-1) + 20*(-1) = \\ &0 + 0 + 0 + 0 + 160 - 20 + 0 - 20 - 20 = \\ &100 \end{aligned}$$

연습문제 2

```
>> x = [ 20 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;
```

```
]; % 원본 데이터
```

```
>> a = [ -1 -1 0;
        -1 0 1;
         0 1 1;]
```

```
a =
```

```
    -1    -1     0
    -1     0     1
     0     1     1
```

```
>> ax = filter2(a, x, 'same')
```

```
ax =
```

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

```
b =
```

```
     0     -1     -1
     1      0     -1
     1      1      0
```

```
>> bx = filter2(b, x, 'same')
```

```
bx =
```

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

c =

-1	-1	-1
2	2	2
-1	-1	-1

>> cx = filter2(c, x, 'same')

cx =

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

```
>> d = [ -1 2 -1;
        -1 2 -1;
        -1 2 -1;]
```

d =

-1	2	-1
-1	2	-1
-1	2	-1

>> dx = filter2(d, x, 'same')

dx =

40	0	10	-10	0	0	0	10	10
60	0	20	-20	0	0	-10	30	10
60	10	0	-10	0	0	-20	50	0
70	0	0	-10	0	0	-30	60	0
60	0	-10	-10	20	-10	-30	60	0
50	-10	0	-10	20	-10	-20	40	10
50	-20	10	0	10	-10	-20	30	30
70	-30	0	30	-20	-10	10	-10	60
60	-30	0	30	-20	-10	10	-10	50

```
>> e = [ -1 -1 -1;
        -1 8 -1;
        -1 -1 -1;]
```

e =

-1	-1	-1
-1	8	-1
-1	-1	-1

>> ex = filter2(e, x, 'same')

ex =

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

```
>> f = [ 1 1 1;
        1 1 1;
        1 1 1;]

f =

     1     1     1
     1     1     1
     1     1     1

>> fx = filter2(f, x, 'same')

fx =

    80    120    110    100     90     90     90     80     50
   120    180    160    140    120    120    130    120     80
   120    170    150    130    120    120    140    130     90
   110    150    120    100     90     90    120    120     90
    90    120    100    100    100    100    120    120     90
    70    100     90    100    100    100    110    110     80
    70    110    110    120    110    100    110    120     90
    80    120    120    120    110    100    110    130     90
    60     90     90     90     80     70     80    100     70
```

```
g =

    -1     0     1
    -1     0     1
    -1     0     1

>> gx = filter2(g, x, 'same')
|
gx =

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

```
h =

     0     -1     0
    -1     4    -1
     0     -1     0

>> hx = filter2(h, x, 'same')

hx =

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


연습문제 3

a 마스크 : 이미지의 우하향 대각선 경계를 강조하기 위해 디자인됨. 강조된 방향의 에지를 찾아내는 역할, 고주파 통과 필터

b 마스크 : 이미지의 좌하향 대각선 경계를 강조하기 위해 디자인됨. 이후 a와 동일

c 마스크 : 수평방향의 에지 강조를 위해 사용, 수평 방향의 픽셀값 변화를 강조하여 수평 에지를 더 뚜렷하게 만든다. 고주파 통과 필터

d 마스크 : 수직 방향의 에지 강조를 위해 사용, 수직 방향의 픽셀값 변화를 강조하여 수직 에지를 더 뚜렷하게 만든다. 고주파 통과 필터

e 마스크 : 라플라시안 마스크의 일종, 이미지 세부 정보를 강조하고 전반적인 에지 감지에 사용, 모든 방향의 에지를 강조하는 특성을 지님. 고주파 통과 필터

f 마스크 : 이미지의 평활화(블러링)에 사용, 이미지의 노이즈를 줄이고 부드러운 이미지 생성

g 마스크 : 수직 방향의 에지를 감지하는데 특화된 형태, 수직방향의 변화를 감지하며 수직 방향 에지 강조, 고주파 통과 필터

h 마스크 : 중심 픽셀 주위의 에지 강조에 사용, 특히 작은 디테일이나 에지 주변 변화를 강조, 이미지에서 작은 구조를 더 뚜렷하게 만드는데 사용

연습문제 4

원본



항등필터



명령 창

```
>> eq=[0 0 0; 0 1 0; 0 0 0];  
>> x = imread('cameraman.tif');  
>> x1 = uint8(filter2(eq, x));
```

```
>> subplot(1,2,1), imshow(x), title('원본')
```

경고: 도킹된 Figure에서 이미지의 초기 배율이 'fit'으로 설정되었습니다.

> imshow (322번 라인)

```
>> subplot(1,2,2), imshow(x1), title('항등필터')
```

경고: 도킹된 Figure에서 이미지의 초기 배율이 'fit'으로 설정되었습니다.

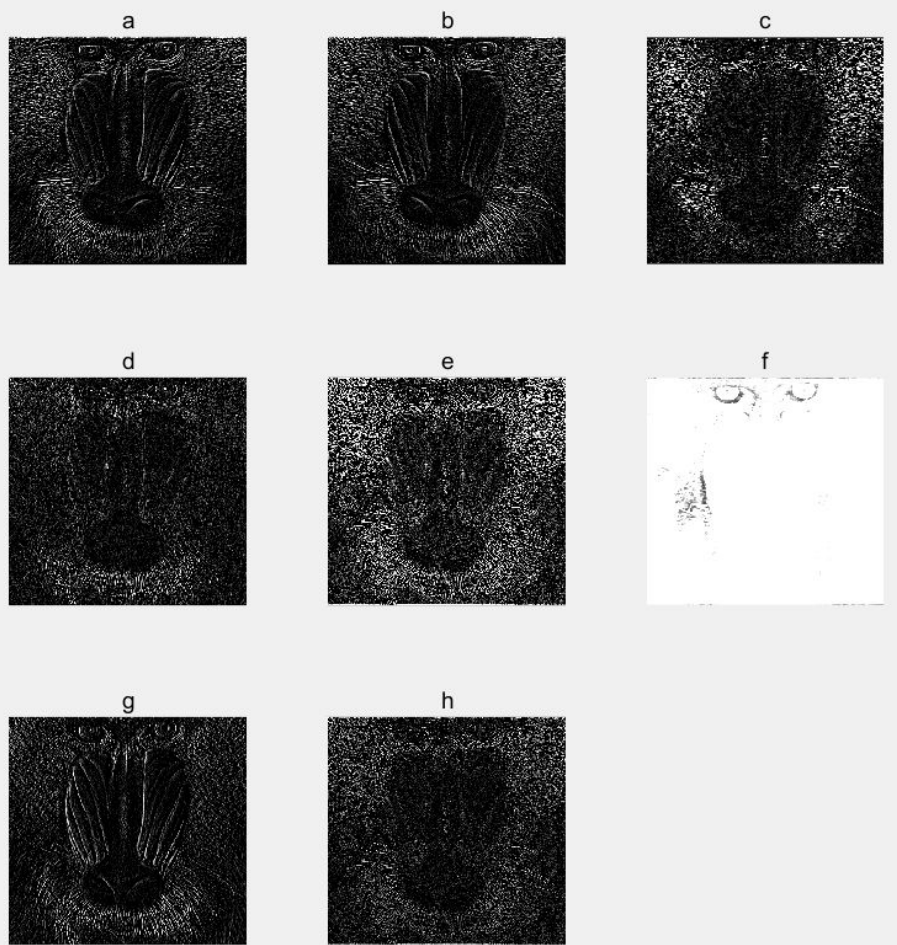
> imshow (322번 라인)

연습문제 5

```

>> load('mandrill.mat');
m = im2uint8(ind2gray(X, map));
>> % 필터 정의
a = [-1 -1 0; -1 0 1; 0 1 1];
b = [0 -1 -1; 1 0 -1; 1 1 0];
c = [-1 -1 -1; 2 2 2; -1 -1 -1];
d = [-1 2 -1; -1 2 -1; -1 2 -1];
e = [-1 -1 -1; -1 8 -1; -1 -1 -1];
f = [1 1 1; 1 1 1; 1 1 1];
g = [-1 0 1; -1 0 1; -1 0 1];
h = [0 -1 0; -1 4 -1; 0 -1 0];
>> % 각 필터 적용
am = uint8(filter2(a, m, 'same'));
bm = uint8(filter2(b, m, 'same'));
cm = uint8(filter2(c, m, 'same'));
dm = uint8(filter2(d, m, 'same'));
em = uint8(filter2(e, m, 'same'));
fm = uint8(filter2(f, m, 'same'));
gm = uint8(filter2(g, m, 'same'));
hm = uint8(filter2(h, m, 'same'));
>> % 결과 표시
subplot(3,3,1), imshow(am), title('a')
subplot(3,3,2), imshow(bm), title('b')
subplot(3,3,3), imshow(cm), title('c')
subplot(3,3,4), imshow(dm), title('d')
subplot(3,3,5), imshow(em), title('e')
subplot(3,3,6), imshow(fm), title('f')
subplot(3,3,7), imshow(gm), title('g')
subplot(3,3,8), imshow(hm), title('h')

```



연습문제 6

```
>> load('mandrill.mat');  
m = im2uint8(ind2gray(X, map));  
>>  
>> filterSizes = [3, 9, 15, 21];  
>>  
>> figure;  
for k = 1:length(filterSizes)  
    filterSize = filterSizes(k); % 필터 크기 선택  
    h = fspecial('average', filterSize); % 평균 필터 생성  
    filtered_image = imfilter(m, h, 'replicate'); % 이미지에 필터 적용  
  
    subplot(2, 2, k), imshow(filtered_image), title(['Average Filter ',  
end
```


Average Filter 3x3



Average Filter 9x9



Average Filter 15x15



Average Filter 21x21

