컴퓨터 비전 세미나

12191584 김창수

3-2. Linear Filtering

- Linear filtering operators involve fixed weighted combinations of pixels in small neighborhoods.
- Output pixel's value is determined as a weighted sum of input pixel values within a small neighborhood N.

Correlation operator

$$g(i,j) = \sum_{k,l} f(i+k, j+l) h(k,l). \qquad g = f \otimes h.$$

- The entries in the weight kernel or mask h(k, l) are often called the filter coefficients.

Convolution operator

$$g(i,j) = \sum_{k,l} f(i-k, j-l) h(k,l) = \sum_{k,l} f(k,l) h(i-k, j-l), \qquad g = f * h,$$

- the sign of the offsets in f has been reversed and h is then called the impulse response function(임펄스 응답 함수)

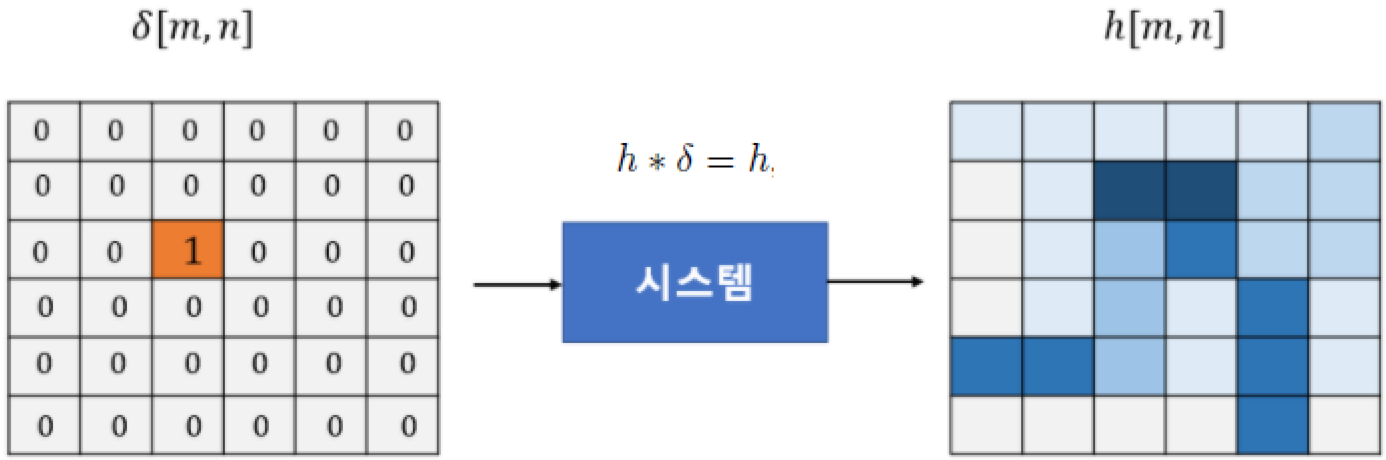
3-2. Linear Filtering

Convolution – 신호처리 내용

$$\delta[m,n] = \begin{cases} 1, & \text{if } m = 0 \text{ and } n = 0 \\ 0, & \text{if } m \neq 0 \text{ or } n \neq 0 \end{cases}$$

2D 임펄스

	$n = -2$	-1	0	1	2	$\cdots$
$m = -2$	0	0	0	0	0	0
-1	0	0	0	0	0	0
0	0	0	1	0	0	0
1	0	0	0	0	0	0
2	0	0	0	0	0	0
$\vdots$	0	0	0	0	0	0



2D 임펄스를 LSI(linear shift-invariant) 시스템의 입력으로 가했을 때의 출력을 h, impulse response function이라 함

3-2. Linear Filtering

Convolution – 신호처리 내용

내용을 확장해서 2D 임펄스 대신 LSI시스템에 임의의 입력 x를 가한다면

$$y[m,n] = \mathcal{F}\{x[m,n]\}$$

$$y[m,n] = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} x[k,l]h[m-k,n-l]$$

$$y[m,n] = h[m,n] * x[m,n]$$

출력(y)은 LSI시스템(h)과 입력(x)의 어떠한 연산으로 이루어지며, 이를 convolution이라 함

Image Processing에서는 LSI 시스템(h)을 필터 또는 커널이라 함

- Correlation and convolution can both be written as a matrix-vector multiply
  $\mathbf{g} = \mathbf{H}\mathbf{f}$, where the H matrix contains the convolution kernels

3-2. Linear Filtering

필터 또는 커널 -> h   입력 -> f   출력 -> g

Correlation Vs. Convolution

Correlation

$$g(i,j) = \sum_{k,l} f(i+k, j+l)h(k,l).$$

Convolution

$$g(i,j) = \sum_{k,l} f(i-k, j-l)h(k,l)$$

f->

.1	.4	.2
.3	.8	.5
.7	.5	.9

4	20	13
47	80	45
73	93	0

.1	.4	.2
.3	.8	.5
.7	.5	.9

h->

-1, -1	-1, 0	-1, 1
0, -1	0, 0	0, 1
1, -1	1, 0	1, 1

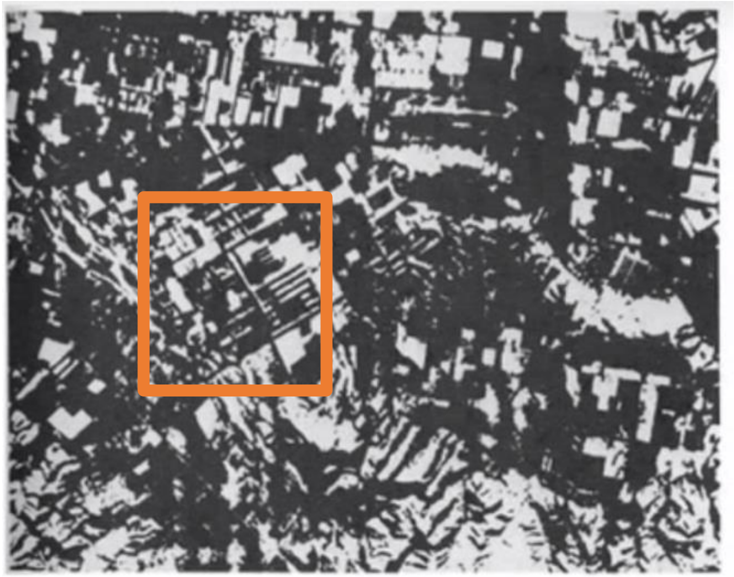
How to simplify convolution?

1, 1	1, 0	1, -1
0, 1	0, 0	0, -1
-1, 1	-1, 0	-1, -1

3-2. Linear Filtering

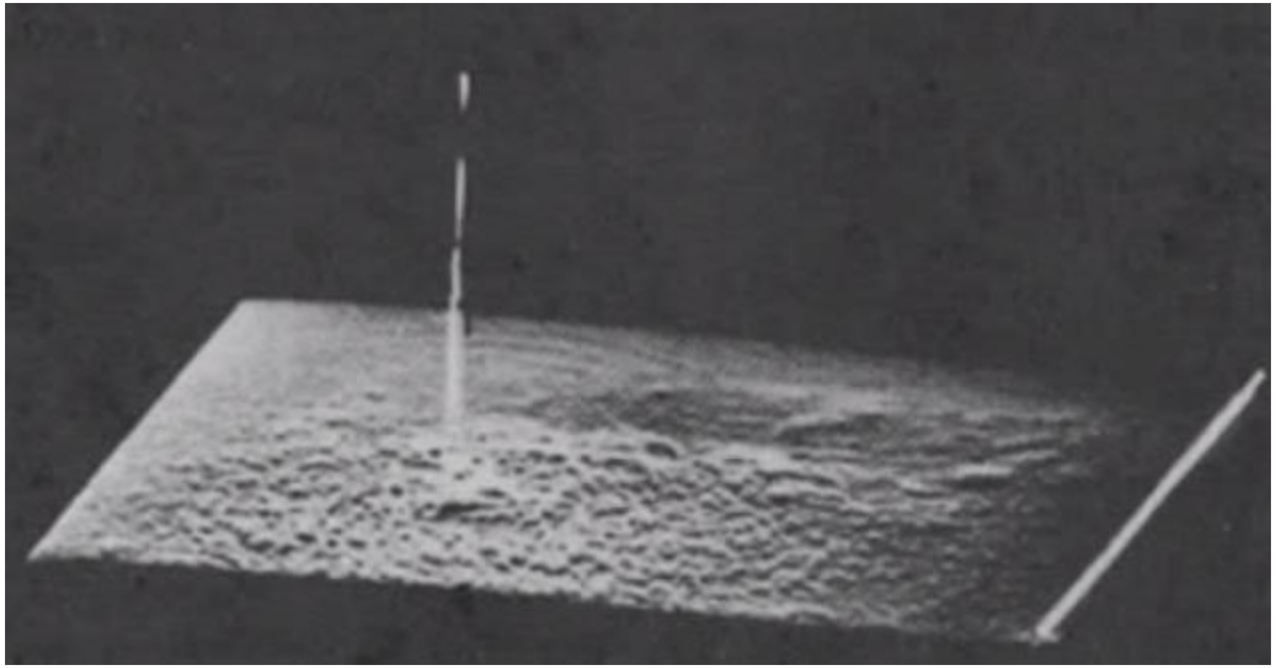
Correlation operator

f->



h->

g->



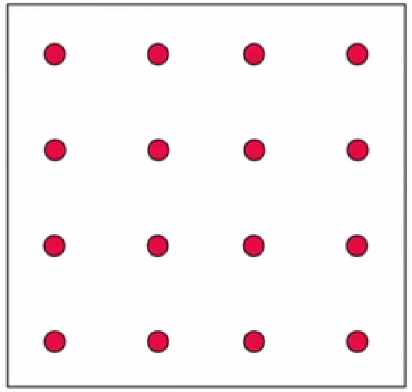
입력 함수(F)와 시스템 함수(커널, h)가 얼마나 유사한지 측정할 수 있음

3-2. Linear Filtering

Convolution operator

f->

h->



g->

입력 함수(F)가 시스템 함수(커널, h)를 통과할 때의 출력을 알 수 있음

3-2. Linear Filtering

Padding

45	60	98	127	132	133	137	133
46	65	98	123	126	128	131	133
47	65	96	115	119	123	135	137
47	63	91	107	113	122	138	134
50	59	80	97	110	123	133	134
49	53	68	83	97	113	128	133
50	50	58	70	84	102	116	126
50	50	52	58	69	86	101	120

*

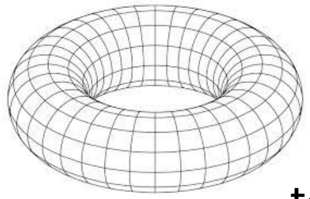
0.1	0.1	0.1
0.1	0.2	0.1
0.1	0.1	0.1

=

69	95	116	125	129	132
68	92	110	120	126	132
66	86	104	114	124	132
62	78	94	108	120	129
57	69	83	98	112	124
53	60	71	85	100	114

- The filtered images suffer from boundary effects , so operation produces a result that is smaller than the original image.
- To deal with this, a number of different padding or extension modes have been developed for operations.

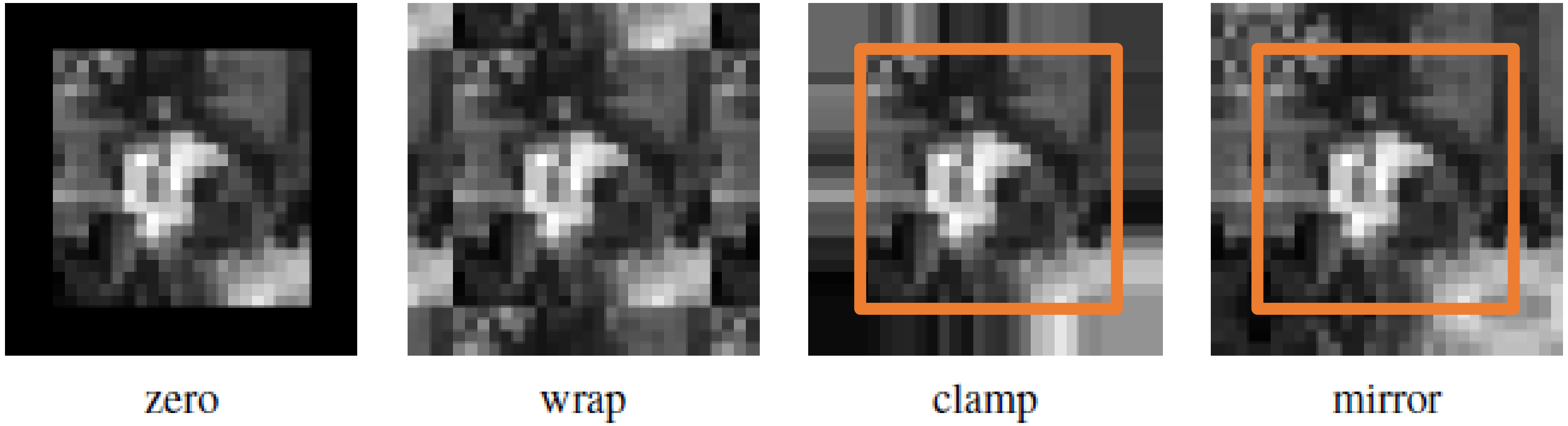
- zero: set all pixels outside the source image to 0
- constant: set all pixels outside the source image to a specified border value
- wrap : loop “around” the image in a “toroidal”(torus의) configuration
- clamp: repeat edge pixels indefinitely
- mirror: reflect pixels across the image edge
- extend: extend the signal by subtracting the mirrored version of the signal from the edge pixel value.



torus

3-2. Linear Filtering

Padding



- zero padding darkens the edges,
- clamp (replication) padding propagates border values inward,
- mirror (reflection) padding preserves colors near the borders

3-2-1. Separable Filtering

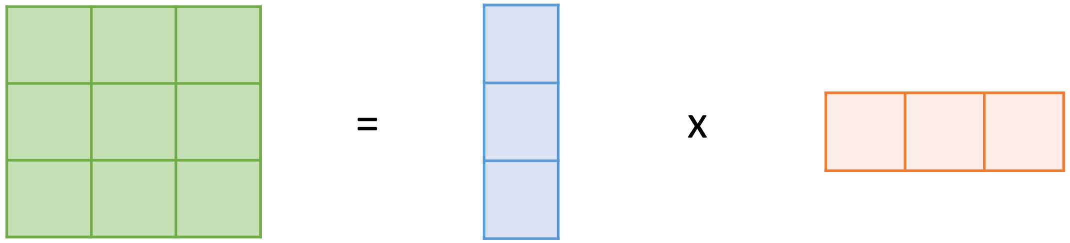
- The process of performing a convolution requires K^2 operations per pixel(K x K kernel).
- This operation can be sped up by first performing a 1D horizontal convolution followed by a 1D vertical convolution, which requires a total of 2K operations per pixel.
- A convolution kernel for which this is possible is said to be separable. $\mathbf{K} = \mathbf{v}\mathbf{h}^T$

- How can we tell if a given kernel K is indeed separable?
- A direct method is to treat the 2D kernel as a 2D matrix K and to take its singular value decomposition (SVD).

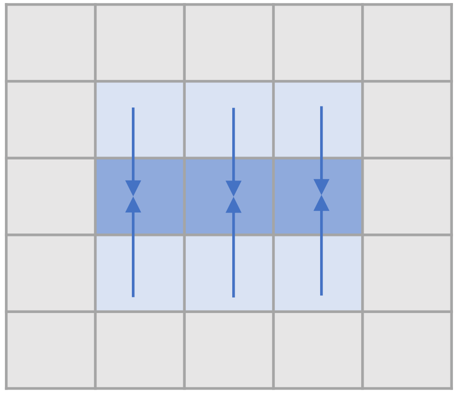
$$\mathbf{K} = \sum_i \sigma_i \mathbf{u}_i \mathbf{v}_i^T$$

- If only the first singular value 0 is non-zero, the kernel is separable. $\sqrt{\sigma_0}\mathbf{u}_0$ and $\sqrt{\sigma_0}\mathbf{v}_0^T$

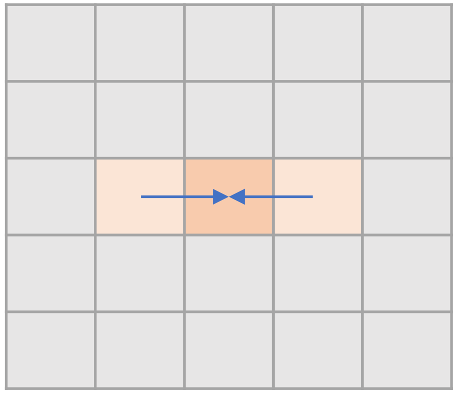
3-2-1. Separable Filtering



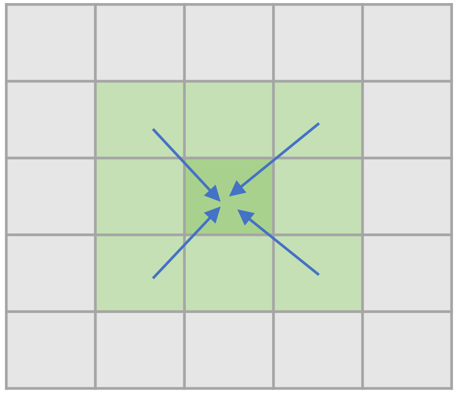
세로 vector convolution



가로 vector convolution



전체 matrix convolution



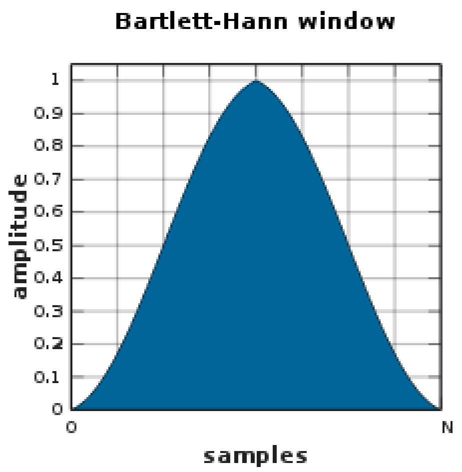
3-2-2. Examples of Linear Filtering

moving average or box filter

- simply averages the pixel values in a K x K window
- convolving the image with a kernel of all ones and then scaling

bilinear kernel

- convolving the image with a piecewise linear “tent” function(Bartlett filter)
- A smoother image can be obtained



$$\frac{1}{K^2}\begin{array}{|c|c|c|c|}\hline 1 & 1 & \cdots & 1 \\ \hline 1 & 1 & \cdots & 1 \\ \hline \vdots & \vdots & 1 & \vdots \\ \hline 1 & 1 & \cdots & 1 \\ \hline \end{array} \qquad \frac{1}{16}\begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

$$\frac{1}{K}\begin{array}{|c|c|c|c|}\hline 1 & 1 & \cdots & 1 \\ \hline \end{array} \qquad \frac{1}{4}\begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline \end{array}$$

(a) box, $K = 5$ (b) bilinear

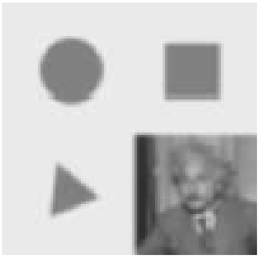
3-2-2. Examples of Linear Filtering

$$\frac{1}{256}\begin{array}{|c|c|c|c|c|}\hline 1 & 4 & 6 & 4 & 1 \\ \hline 4 & 16 & 24 & 16 & 4 \\ \hline 6 & 24 & 36 & 24 & 6 \\ \hline 4 & 16 & 24 & 16 & 4 \\ \hline 1 & 4 & 6 & 4 & 1 \\ \hline \end{array}$$

$$\frac{1}{16}\begin{array}{|c|c|c|c|c|}\hline 1 & 4 & 6 & 4 & 1 \\ \hline \end{array}$$

Gaussian kernel

- kernel from convolving the linear tent function with itself
- approximate kernel can also be obtained by iterated convolution with box filters
- In applications where the filters need to be rotationally symmetric, carefully tuned Gaussians should be used



(c) "Gaussian"

The kernels we just discussed are all examples of blurring (smoothing) or low-pass kernels, since they pass through the lower frequencies while attenuating higher frequencies.

- smoothing kernels can also be used to sharpen images(unsharp masking)
- Since blurring the image reduces high frequencies, adding some of the difference between the original and the blurred image makes it sharper

$$g_{\text{sharp}} = f + \gamma(f - h_{\text{blur}} * f).$$

3-2-2. Examples of Linear Filtering

Linear filtering can also be used as a pre-processing stage
to edge extraction and interest point(특징점) detection algorithms

Sobel operator

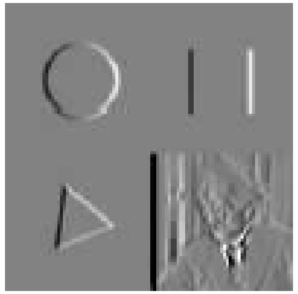
- separable combination of a horizontal central difference and a vertical tent filter

Corner detector

- corner detector looks for simultaneous horizontal and vertical second derivatives.
- it responds not only to the corners of the square, but also along diagonal edges.
- Better corner detectors that are more rotationally invariant

$$\frac{1}{8}\begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline -2 & 0 & 2 \\ \hline -1 & 0 & 1 \\ \hline \end{array} \qquad \frac{1}{4}\begin{array}{|c|c|c|}\hline 1 & -2 & 1 \\ \hline -2 & 4 & -2 \\ \hline 1 & -2 & 1 \\ \hline \end{array}$$

$$\frac{1}{2}\begin{array}{|c|c|c|}\hline -1 & 0 & 1 \\ \hline \end{array} \qquad \frac{1}{2}\begin{array}{|c|c|c|}\hline 1 & -2 & 1 \\ \hline \end{array}$$



(d) Sobel

(e) corner

3-2-3. Band-pass and steerable filters

The Sobel and corner operators are simple examples of band-pass and oriented filters

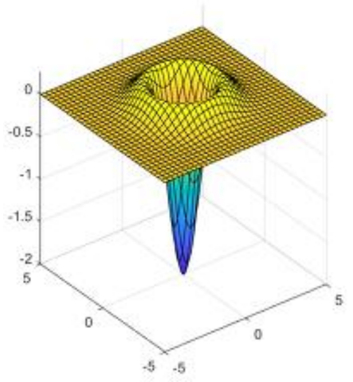
Band-pass filter

- 특정 주파수만 통과시키는 필터(filter out both low and high frequencies)
- created by first smoothing the image with a Gaussian filter, and then taking the first or second derivatives
- The second derivative of a two-dimensional image, is known as the Laplacian operator $\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2},$

- Blurring an image with a Gaussian and then taking its Laplacian is equivalent to Laplacian of Gaussian (LoG) filter

$$\nabla^2 G(x, y; \sigma) = \left(\frac{x^2 + y^2}{\sigma^4} - \frac{2}{\sigma^2} \right) G(x, y; \sigma),$$

$$G(x, y; \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2 + y^2}{2\sigma^2}},$$



- LoG 필터는 엣지를 검출하는데 사용됨(스무딩 -> 영교차 판단)
- 우리 눈 안의 망막 신경절 세포와 비슷함

3-2-3. Band-pass and steerable filters

Steerable filter $(\nabla_{\hat{\mathbf{u}}} G) * f$. $\hat{\mathbf{u}} = (\cos\theta, \sin\theta)$,

- Sobel operator can be obtained by smoothing with a Gaussian, and then taking a directional derivative.

- The smoothed directional derivative filter $G_{\hat{\mathbf{u}}} = uG_x + vG_y$ $\hat{\mathbf{u}} = (u, v)$. is an example of a steerable filter.
- can be computed by first convolving with the pair of filters (Gx, Gy) and then steering the filter by multiplying this gradient field with a unit vector u
- The advantage of this approach is that a whole family of filters can be evaluated with very little cost

(a) (b) (c)

Figure 3.15 *Second-order steerable filter (Freeman 1992) © 1992 IEEE: (a) original image of Einstein; (b) orientation map computed from the second-order oriented energy; (c) original image with oriented structures enhanced.*

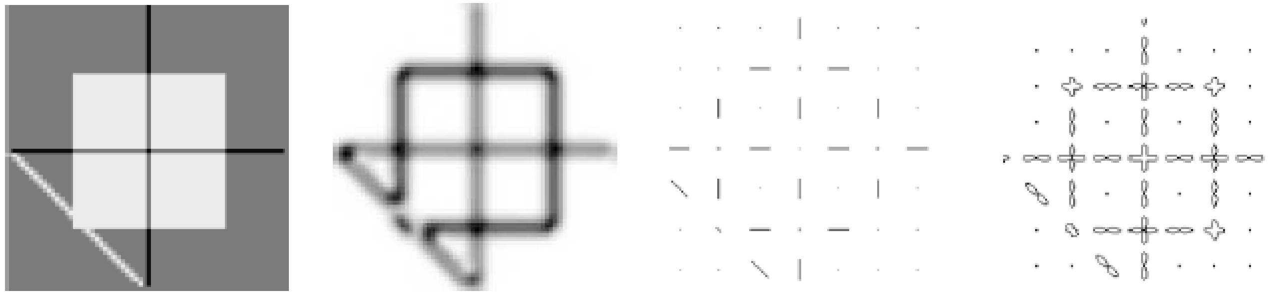
3-2-3. Band-pass and steerable filters

- for directional Gaussian derivatives, it is possible to steer any order of derivative with a small number of basis functions

$$G_{\hat{\mathbf{u}}\hat{\mathbf{u}}} = u^2 G_{xx} + 2uvG_{xy} + v^2 G_{yy}.$$

- each of the basis filters can be computed using a linear combination of a small number of separable filters

- This makes it possible to construct directional derivative filters of increasingly greater directional selectivity
- higher order steerable filters can respond to potentially more than a single edge orientation, and they can respond to both bar edges and the classic step edges



(a) (b) (c) (d)

Figure 3.16 *Fourth-order steerable filter (Freeman and Adelson 1991) © 1991 IEEE:*

- Steerable filters are often used to construct both feature descriptors and edge detectors

3-2-3. Summed area table

$$\frac{1}{K^2}\begin{array}{|c|c|c|c|}\hline 1 & 1 & \cdots & 1 \\ \hline 1 & 1 & \cdots & 1 \\ \hline \vdots & \vdots & 1 & \vdots \\ \hline 1 & 1 & \cdots & 1 \\ \hline \end{array}$$

- If an image is going to be repeatedly convolved with different box filters, you can precompute the summed area table.

$$s(i,j) = \sum_{k=0}^{i}\sum_{l=0}^{j} f(k,l). \qquad s(i,j) = s(i-1,j) + s(i,j-1) - s(i-1,j-1) + f(i,j).$$

- 처음부터 자기 위치까지의 모든 픽셀 값의 합을 저장하는 새로운 행렬
- The image s(I, j) is often called an integral image
- To find the summed area (integral) inside a rectangle [i0, i1] x [j0, j1],

$$S(i_0 \ldots i_1, j_0 \ldots j_1) = s(i_1, j_1) - s(i_1, j_0 - 1) - s(i_0 - 1, j_1) + s(i_0 - 1, j_0 - 1).$$

- A disadvantage of summed area tables is that they require logM + logN(M x N image) extra bits in the accumulation
- In computer vision, they have been used in face detection to compute simple multi-scale low-level features

3-2-3. Summed area table

3	2	7	2	3
1	5	1	3	4
5	1	**3**	5	1
4	3	2	1	6
2	4	1	4	8

(a) S = 24

3	5	12	14	17
4	*11*	**19**	24	31
9	**17**	28	38	46
13	24	37	48	62
15	30	44	59	81

(b) s = 28

3	5	12	*14*	17
4	11	19	24	31
9	17	28	38	46
13	24	37	**48**	62
15	30	44	59	81

(c) S = 24

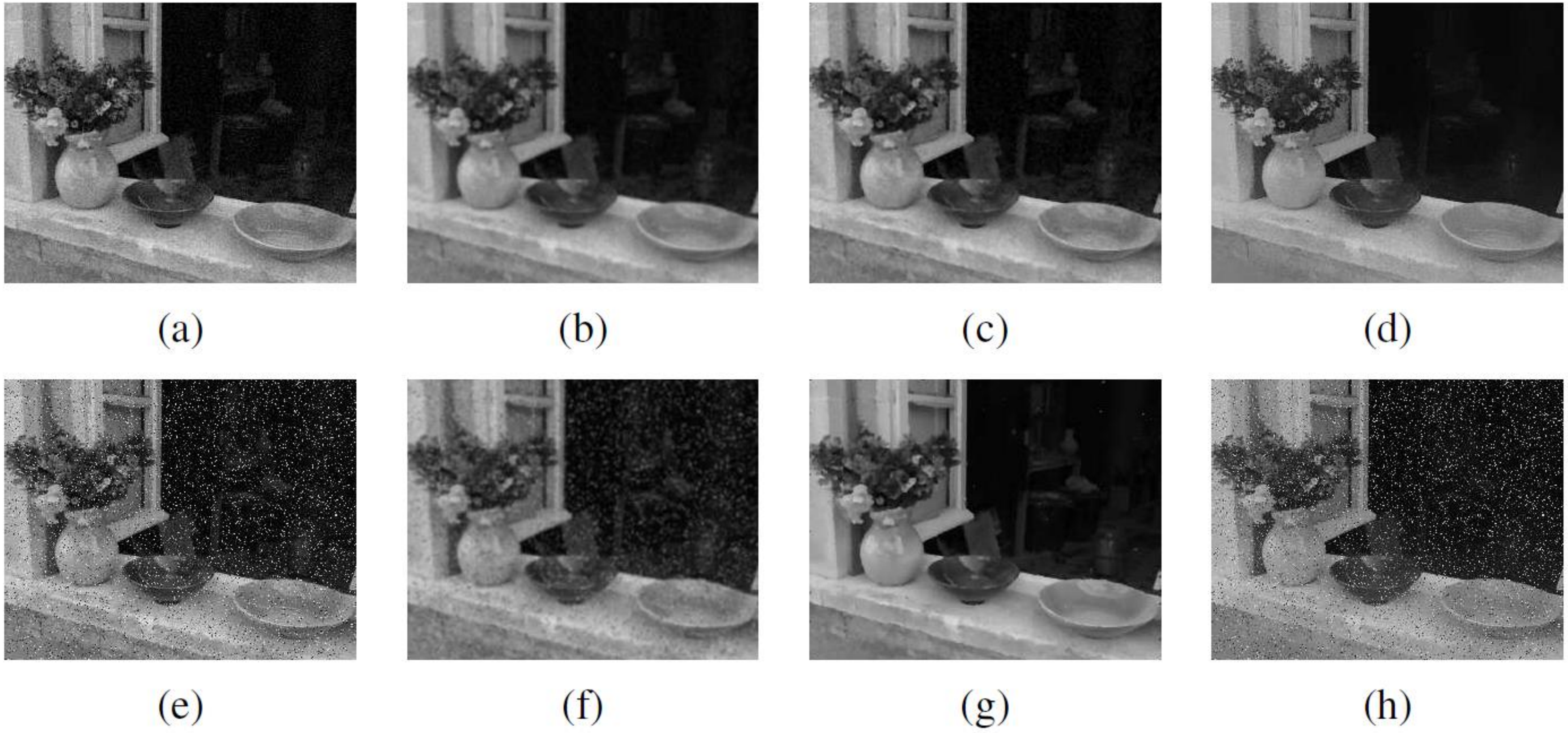
- (a) original image (b) summed area table (c) computation of area sum
- red value in the summed area table is computed recursively from its three adjacent blue neighbors
- the sum of the green values are computed by combining the four purple values

- In practice, separable moving average filters are usually preferred, unless many different window shapes and sizes are being considered

3-2-3. Recursive filtering

$$s(i,j) = s(i-1,j) + s(i,j-1) - s(i-1,j-1) + f(i,j).$$

- is an example of a recursive filter, one whose values depends on previous filter outputs
- such filters are known as infinite impulse response (IIR), since the output of the filter to an impulse goes on forever
- The filters which involve the image with a finite extent kernel, are known as finite impulse response (FIR)



(a) original image with Gaussian noise;
(b) Gaussian filtered;
(c) median filtered;
(d) bilaterally filtered;

(e) original image with shot noise;
(f) Gaussian filtered;
(g) median filtered;
(h) bilaterally filtered.

bilateral filter fails to remove the shot noise

Gaussian 필터 : 픽셀 주변 값들의 가중치 평균을 계산 -> 노이즈를 완화하지만 엣지 무뎌짐

Median 필터 : 픽셀 주변 값들의 중앙값으로 계산 -> 임펄스성으로 튀는 노이즈 완화

Bilateral 필터 : 픽셀 주변 값들의 가중치 평균과 픽셀 차이를 계산 -> 엣지를 보존하면서 노이즈 완화