

다양한 공간적 필터링 기법

- ▶ 공간적 필터링
- ▶ 영상 부드럽게 만들기
- ▶ 영상 날카롭게 만들기
- ▶ 잡음 생성
- ▶ 잡음 제거를 위한 비선형 필터

공간적 필터링

□ 필터링(Filtering)

- 영상에서 특정 주파수 성분을 제거하는 작업

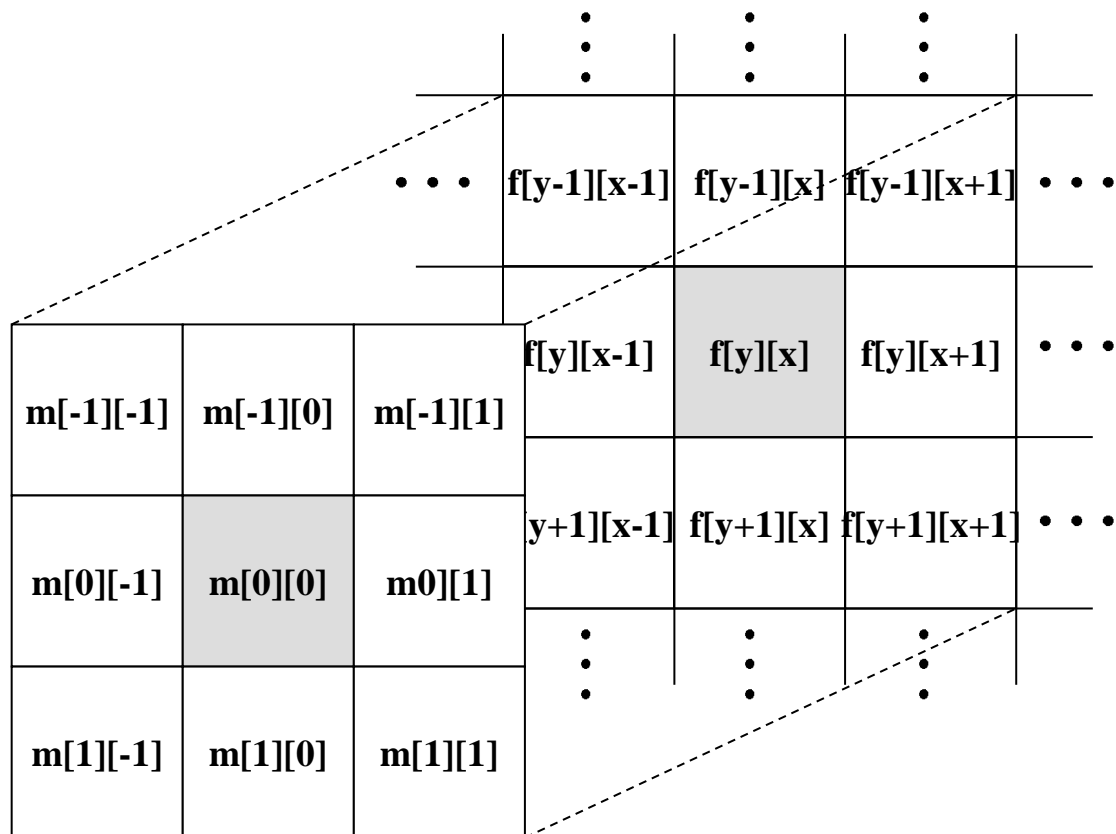
□ 공간적 필터링(Spatial filtering)

- 영상의 픽셀 값을 그대로 이용하여 영상에서 원하는 정보만을 걸러내는 영상 처리 기법
- 마스크(mask)를 이용하는 방법
 - 윈도우(window), 템플릿(template), 커널(kernel)

$$g[y][x] = \sum_{j=-w}^w \sum_{i=-w}^w m[j][i] f[y+j][x+i]$$

공간적 필터링

- 3×3 크기의 마스크를 이용한 필터링 방법



공간적 필터링

□ 최외곽 픽셀 처리 방법

- 최외곽 픽셀은 마스크 연산에서 제외
- 최외곽 바깥에 가상의 픽셀이 있다고 가정

| | | | | |
|----------|----------|-----|------------|------------|
| [0][0] | [0][1] | ... | [0][w-2] | [0][w-1] |
| [1][0] | [1][1] | ... | [1][w-2] | [1][w-1] |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| [h-2][0] | [h-2][1] | ... | [h-2][w-2] | [h-2][w-1] |
| [h-1][0] | [h-1][1] | ... | [h-1][w-2] | [h-1][w-1] |

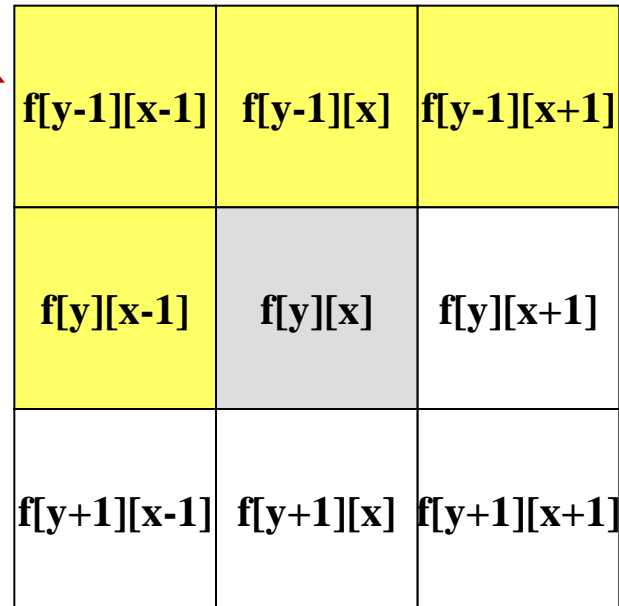
| | | | | |
|---|--------|--------|--------|-----|
| 0 | 0 | 0 | 0 | |
| 0 | [0][0] | [0][1] | [0][2] | |
| 0 | [1][0] | [1][1] | [1][2] | ... |
| 0 | [2][0] | [2][1] | [2][2] | |
| | ⋮ | ⋮ | ⋮ | |

공간적 필터링

□ 마스크 연산 시 주의할 점

- 일반적으로 원본 영상의 복사본을 만들어 픽셀 값을 참조해야 함.

마스크 연산을 이용하여 (x,y) 좌표의 값을 설정할 때, 노란색 부분의 픽셀은 이미 원본 영상의 픽셀 값이 변경된 상태임.



| | | |
|---------------|-------------|---------------|
| $f[y-1][x-1]$ | $f[y-1][x]$ | $f[y-1][x+1]$ |
| $f[y][x-1]$ | $f[y][x]$ | $f[y][x+1]$ |
| $f[y+1][x-1]$ | $f[y+1][x]$ | $f[y+1][x+1]$ |

영상 부드럽게 만들기

□ 평균 값 필터(Mean filter)

- 영상의 특정 좌표 값을 주변 픽셀들의 그레이스케일 값들의 산술 평균 값으로 설정하는 필터

$$\frac{1}{9} \times$$

| | | |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 1 | 1 |
| 1 | 1 | 1 |

$$\frac{1}{25} \times$$

| | | | | |
|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |

영상 부드럽게 만들기

- 실제 영상에 평균 값 필터를 적용한 결과
 - 입력 영상, 3x3, 5x5 크기의 마스크를 이용한 결과



영상 부드럽게 만들기

- 가중 평균 값 필터(Weighted mean filter)
 - 평균 값 필터의 일종
 - 필터 마스크의 가운데 위치한 픽셀에 가중치를 더 주는 방법

$$\frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

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

영상 부드럽게 만들기

- 실제 영상에 가중 평균 값 필터를 적용한 결과
 - 입력 영상, 3x3, 5x5 크기의 마스크를 이용한 결과

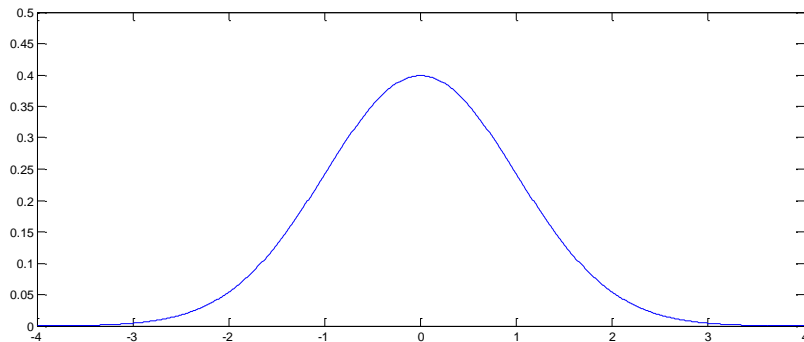


영상 부드럽게 만들기

□ 가우시안(Gaussian) 함수

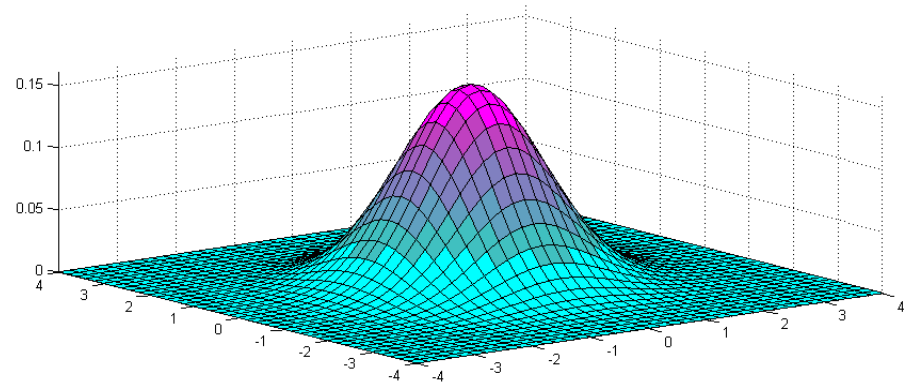
■ 1차원 가우시안 함수

$$G_{\sigma}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$



■ 2차원 가우시안 함수

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



영상 부드럽게 만들기

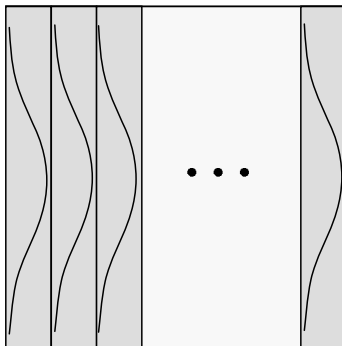
□ 2차원 가우시안 필터 마스크($\sigma = 1.0$)

| | | | | | | | | |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0001 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 0.0000 | 0.0000 | 0.0002 | 0.0011 | 0.0018 | 0.0011 | 0.0002 | 0.0000 | 0.0000 |
| 0.0000 | 0.0002 | 0.0029 | 0.0131 | 0.0215 | 0.0131 | 0.0029 | 0.0002 | 0.0000 |
| 0.0000 | 0.0011 | 0.0131 | 0.0585 | 0.0965 | 0.0585 | 0.0131 | 0.0011 | 0.0000 |
| 0.0001 | 0.0018 | 0.0215 | 0.0965 | 0.1592 | 0.0965 | 0.0215 | 0.0018 | 0.0001 |
| 0.0000 | 0.0011 | 0.0131 | 0.0585 | 0.0965 | 0.0585 | 0.0131 | 0.0011 | 0.0000 |
| 0.0000 | 0.0002 | 0.0029 | 0.0131 | 0.0215 | 0.0131 | 0.0029 | 0.0002 | 0.0000 |
| 0.0000 | 0.0000 | 0.0002 | 0.0011 | 0.0018 | 0.0011 | 0.0002 | 0.0000 | 0.0000 |
| 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0001 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |

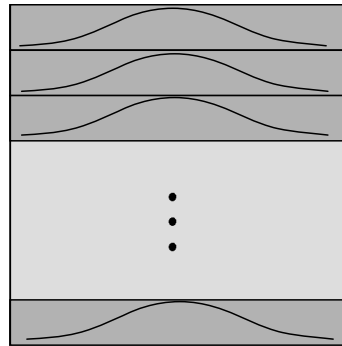
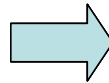
영상 부드럽게 만들기

- 1차원 가우시안 함수를 이용하여 2차원 가우시안 필터 마스크 만들기

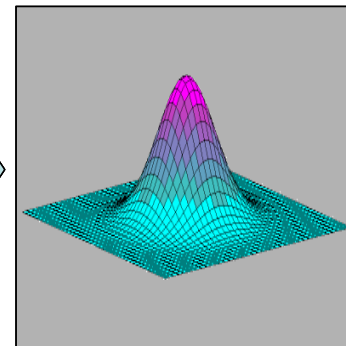
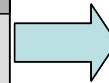
$$G_{\sigma}(x, y) = \frac{1}{2\pi\sigma^2} e^{\left(-\frac{x^2+y^2}{2\sigma^2}\right)} = \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{x^2}{2\sigma^2}\right)} \times \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{y^2}{2\sigma^2}\right)}$$



1D 세로방향
가우시안



1D 가로방향
가우시안



2D 가우시안
필터링

영상 부드럽게 만들기

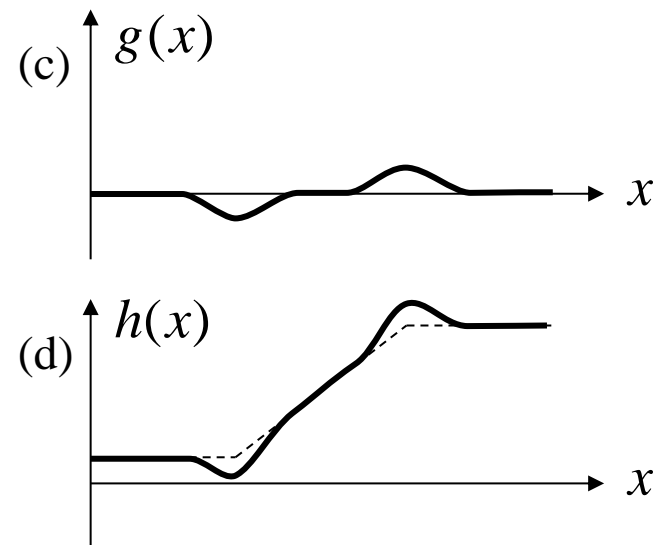
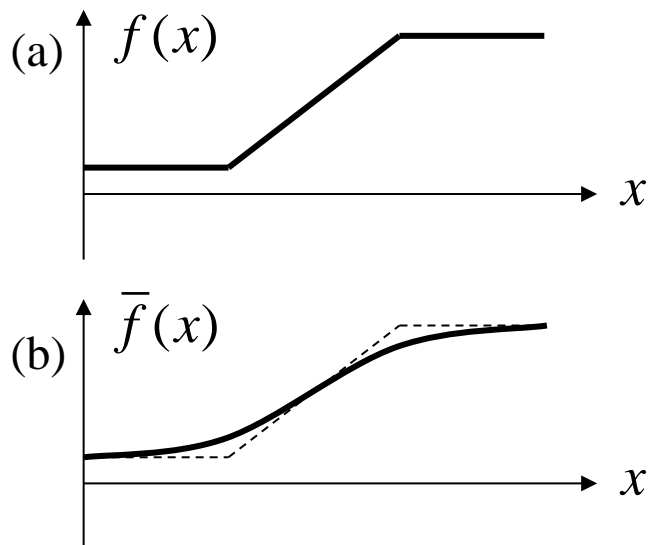
- 실제 영상에 가우시안 필터를 적용한 결과
(차례대로 입력 영상, $\sigma = 1.0, 2.0, 3.0, 4.0, 5.0$)



영상 날카롭게 만들기

□ 언샤프 마스크(Unsharp mask) 필터

$$\begin{aligned}h(x, y) &= f(x, y) + g(x, y) \\ &= f(x, y) + [f(x, y) - \bar{f}(x, y)]\end{aligned}$$



영상 날카롭게 만들기

□ 언샤프 마스크 필터링 시뮬레이션



3x3 크기의 가중
평균 값 필터를 사
용하여 부드럽게
만든 영상

입력 영상에서
가중 평균값 필터
결과 영상을 이용
하여 뵤셈 연산을
한 결과



입력 영상에 뵤셈
결과 영상을 더한
영상

영상 날카롭게 만들기

□ 라플라시안(Laplacian) 필터

- 영상의 2차 미분을 이용

□ 영상의 x 방향으로의 일차 미분

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

□ 영상의 x 방향으로의 이차 미분

$$\begin{aligned}\frac{\partial^2 f}{\partial x^2} &= [f(x+1) - f(x)] - [f(x) - f(x-1)] \\ &= f(x+1) + f(x-1) - 2f(x)\end{aligned}$$

영상 날카롭게 만들기

□ 라플라시안 필터

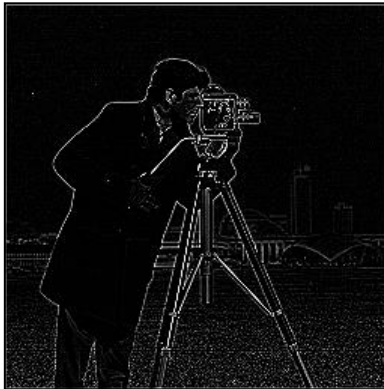
$$\begin{aligned}\nabla^2 f &= \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \\ &= [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] \\ &\quad - 4f(x, y)\end{aligned}$$

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

| | | |
|---|----|---|
| 1 | 1 | 1 |
| 1 | -8 | 1 |
| 1 | 1 | 1 |

영상 날카롭게 만들기

- 실제 영상에 라플라시안 필터를 적용한 결과
 - 입력 영상, 4방향, 8방향을 고려한 라플라시안 필터링 결과



영상 날카롭게 만들기

□ 언샤프 마스크 필터 수식 변경

$$\begin{aligned}h(x, y) &= f(x, y) + \nabla^2 f(x, y) \\&= f(x, y) - \{[f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y)\}\end{aligned}$$

$$\begin{aligned}h(x, y) &= 5f(x, y) \\&\quad - f(x+1, y) - f(x-1, y) - f(x, y+1) - f(x, y-1)\end{aligned}$$

| | | |
|----|----|----|
| 0 | -1 | 0 |
| -1 | 5 | -1 |
| 0 | -1 | 0 |

| | | |
|----|----|----|
| -1 | -1 | -1 |
| -1 | 9 | -1 |
| -1 | -1 | -1 |

영상 날카롭게 만들기

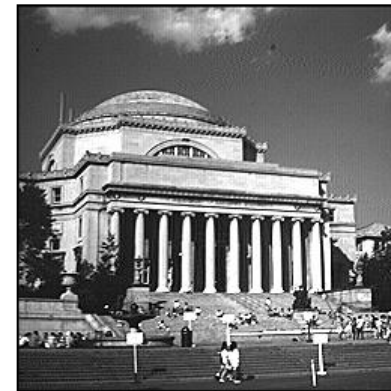
□ 하이부스트(High-boost filter) 필터

$$h(x, y) = \alpha f(x, y) + \nabla^2 f(x, y)$$

- $\alpha = 1$ 이면 언샤프 마스크 필터와 동일



$\alpha = 1$



$\alpha = 1.5$

잡음 생성

□ 소금&후추(Salt&Pepper) 잡음

- 입력 영상의 임의의 좌표 픽셀 값을 0 또는 255로 설정하는 잡음



□ 실제 영상에 소금&후추 잡음을 추가한 결과

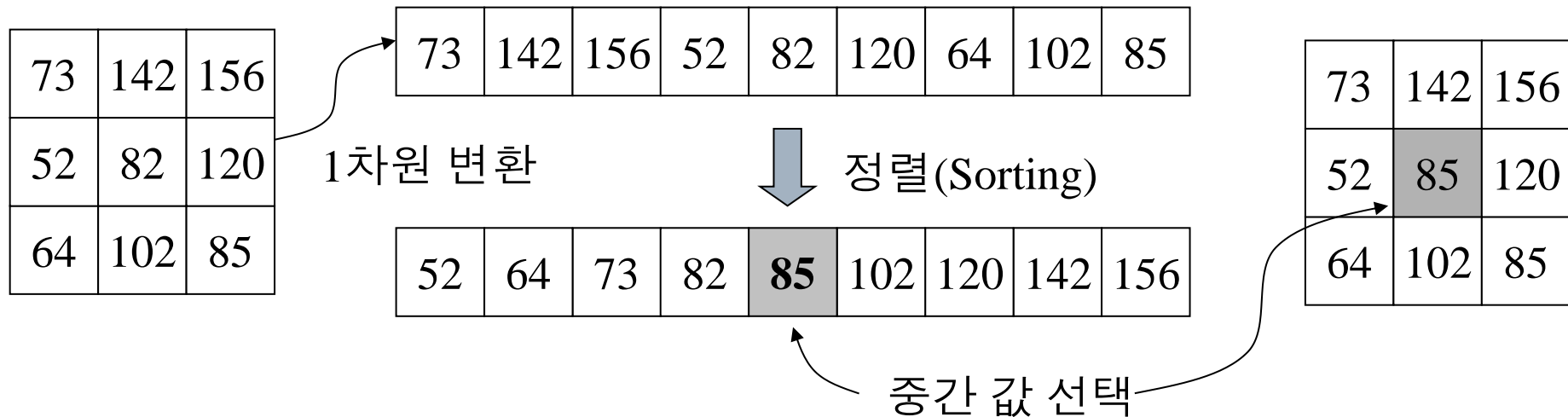
- 입력 영상, 10%, 30% 의 소금&후추 잡음을 추가한 결과



잡음 제거를 위한 비선형 필터

□ 미디언(Median) 필터

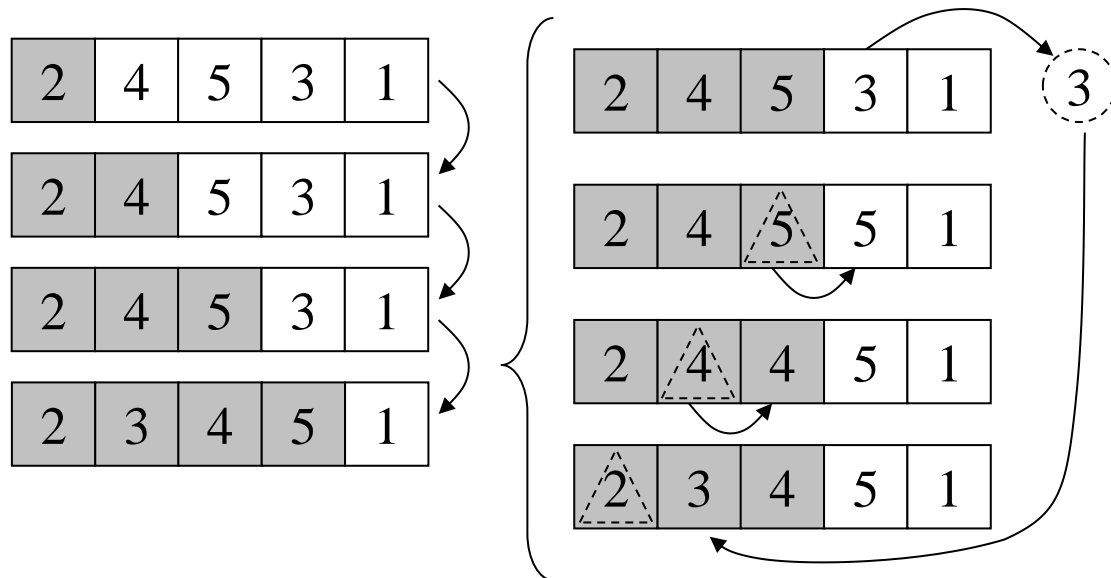
- 입력 영상의 (x,y) 좌표 주변 픽셀들의 값들을 오름 또는 내림 차순으로 정렬하여 그 중앙에 있는 픽셀 값을 사용



삽입 제거를 위한 비선형 필터

□ 삽입 정렬

- 배열의 맨 첫 번째 원소부터 이미 정렬되어있는 원소들을 순회하면서 데이터가 들어갈 위치를 찾아 삽입하는 정렬 알고리즘



잡음 제거를 위한 비선형 필터

- 소금&후추 잡음이 포함된 영상에 미디언 필터를 적용한 결과

10%



20%



잡음 제거를 위한 비선형 필터

- 경계선 보전(Edge-preserving) 잡음 제거
 - 경계선 정보는 살리고 잡음은 제거
- 비등방성 확산(Anisotropic diffusion) 필터

$$I_t = \text{div}(c(x, y, t)\nabla I) = c(x, y, t)\Delta I + \nabla c \cdot \nabla I$$

- div : 발산(divergence) 연산자
- ∇ : 그래디언트(gradient) 연산자
- $\Delta = \nabla^2$: 라플라시안(Laplacian) 연산자

잡음 제거를 위한 비선형 필터

□ 비등방성 확산 수식을 이산 함수 형태로 변환

$$I_{i,j}^{t+1} = I_{i,j}^t + \lambda [c_N \cdot \nabla_N I + c_S \cdot \nabla_S I + c_E \cdot \nabla_E I + c_W \cdot \nabla_W I]$$

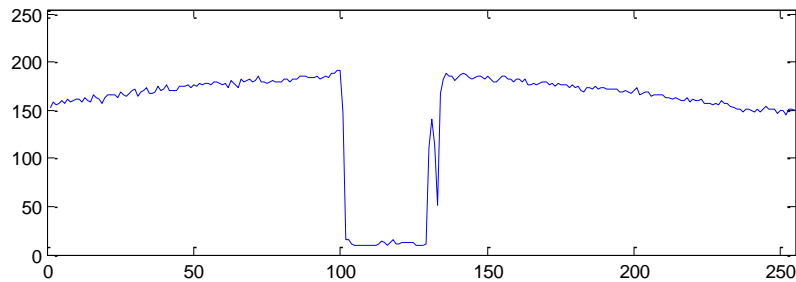
$$\begin{aligned} \nabla_N(I_{i,j}) &= I_{i,j-1} - I_{i,j} & \nabla_S(I_{i,j}) &= I_{i,j+1} - I_{i,j} \\ \nabla_E(I_{i,j}) &= I_{i+1,j} - I_{i,j} & \nabla_W(I_{i,j}) &= I_{i-1,j} - I_{i,j} \end{aligned}$$

■ 전달 계수 함수

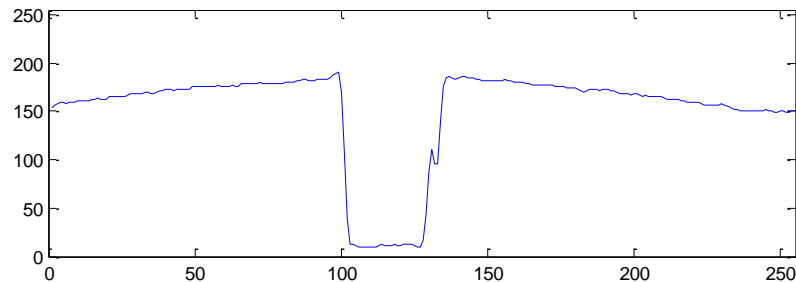
$$c_{N \sim W} = g(\|\nabla_{N \sim W} I\|) = \frac{1}{1 + \left(\frac{\|\nabla_{N \sim W} I\|}{K} \right)^2}$$

잡음 제거를 위한 비선형 필터

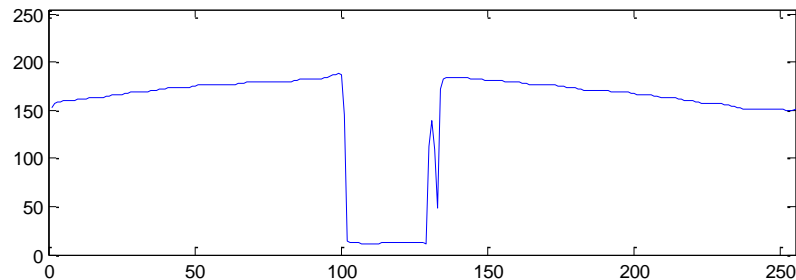
원본영상



가우시안
필터링



비등방성
확산



잡음 제거를 위한 비선형 필터

- 실제 영상에 비등방성 확산 필터를 적용한 결과
 - 입력 영상, 반복 횟수 10, 반복 횟수 30을 적용한 결과

