Computer Vision
and
Machine Learning

(Image smoothing / sharpening)

Bhabatosh Chanda

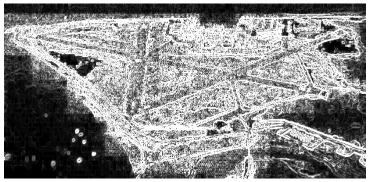
Outline

- Introduction
  - Signal and noise characteristics
- Noise cleaning or smoothing
  - Mean and Order statistics filters
  - Different kernels
- Sharpening
  - Laplacian
  - Smoothing method

Types of processing

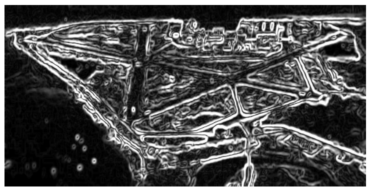
- Spatial domain processing
  - Directly operates on the pixel values in the spatial domain.
  - Point process
  - Neighbourhood process
    - Most common is convolution operation.
- Frequency domain processing
  - First transforms the image data to frequency domain using an orthogonal transform.
  - Appropriate filtering is applied on transformed data.
  - Inverse transform is applied on filtered data to get back into spatial domain.

Effect of noise and smoothing



along with the “correct” edges, contains too many false edges.

Effect of noise and smoothing



Many false edges are smoothed, unfortunately so are true edges.

Smoothing

Assumptions

(i) regarding noise

Signal independent and additive

Zero-mean and symmetrically distributed

(ii) regarding intensity

May be modeled by smooth surface (e.g. plane)

Noisy image: Example

- Let us consider a 5x5 block of a noise-free image

6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8

Original image

Noisy image: Example

- Let us consider a 5x5 block of a noise-free image
- A zero mean symmetrically distributed random noise is added to it.

6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8

Original image

+

0	0	2	0	1
-3	0	0	3	0
0	-1	4	0	-2
3	0	0	-1	0
0	-3	1	0	-4

Noise terms

Noisy image: Example

- Let us consider a 5x5 block of a noise-free image.
- A zero mean symmetrically distributed random noise is added to it.
- Average of pixel values of the original image and that of the noisy image is same!

Original image

6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8

+

Noise terms

0	0	2	0	1
-3	0	0	3	0
0	-1	4	0	-2
3	0	0	-1	0
0	-3	1	0	-4

=

Noisy image

6	7	9	7	9
3	7	7	10	8
6	6	11	7	6
9	7	7	6	8
6	4	8	7	4

Degradation model

- Noise is signal independent and additive

$$g(r,c) = f(r,c) + \eta(r,c)$$

- For n no. of noisy version of same image

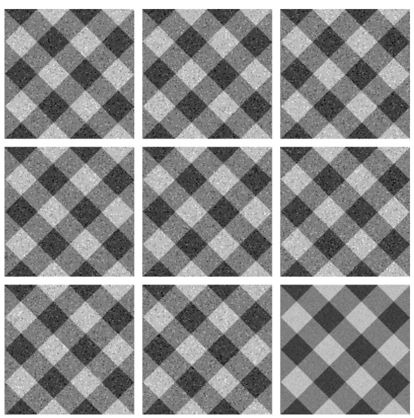
$$g_i(r,c) = f(r,c) + \eta_i(r,c) \quad \text{for} \quad i = 1,2,3,\ldots,n$$

- Let us take pixel-wise average over n image

$$\bar{g}(r,c) = \frac{1}{n}\sum_{i=1}^{n} g_i(r,c) = \frac{1}{n}\sum_{i=1}^{n} f(r,c) + \frac{1}{n}\sum_{i=1}^{n} \eta_i(r,c)$$

$$= f(r,c)$$

Multiple noisy image



Neighborhood process: Smoothing

- Noise causes abrupt change in graylevel.
- Noisy pixel is either much brighter or much darker than its neighbouring pixels.
- A pixel and its neighbourhood is considered to compute the value (colour) of the corresponding pixel in the output image.

$$f(x,y) = T_{(u,v) \in N(x,y)}[g(u,v)]$$

Mean square estimation

- Image graylevel over a patch is approximated by a plane
  $f(x,y) = A(x - x_0) + B(y - y_0) + C \quad \text{given } f(x_0, y_0) = C$
- Noisy graylevel may be modeled as

$$g(x,y) = f(x,y) + \eta(x,y) = A(x - x_0) + B(y - y_0) + C + \eta(x,y)$$

- Least square error is then defined as

$$e = \sum_{(x,y)\in W} [g(x,y) - A(x - x_0) - B(y - y_0) - C]^2 - \sum_{(x,y)\in W} [\eta(x,y)]^2$$

- Estimated noise free graylevel is

$$\bar{g}(x_0, y_0) = C = \frac{1}{|W|} \sum_{(x,y)\in W} g(x,y)$$

Noisy image: Example

- Let us consider a 5x5 block of a noise-free image

6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8

Original image

Noisy image: Example

- Let us consider a 5x5 block of a noise-free image
- A zero mean symmetrically distributed random noise is added to it.

6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8

Original image

$+$

0	0	2	0	1
-3	0	0	3	0
0	-1	4	0	-2
3	0	0	-1	0
0	-3	1	0	-4

Noise terms

Noisy image: Example

- Let us consider a 5x5 block of a noise-free image.
- A zero mean symmetrically distributed random noise is added to it.
- Average of pixel values of the original image and that of the noisy image is same!

6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8

Original image

$+$

0	0	2	0	1
-3	0	0	3	0
0	-1	4	0	-2
3	0	0	-1	0
0	-3	1	0	-4

Noise terms

$=$

6	7	9	7	9
3	7	7	10	8
6	6	11	7	6
9	7	7	6	8
6	4	8	7	4

Noisy image

Noisy image: Example

- Because of linear variation in graylevel in the original image, centre pixel has the average of the pixel values.
- Hence, if we replace the graylevel of the centre pixel of the noisy image by the average value of the block, we get back original value at that position.

6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8
6	7	7	7	8

Original image

+

0	0	2	0	1
-3	0	0	3	0
0	-1	4	0	-2
3	0	0	-1	0
0	-3	1	0	-4

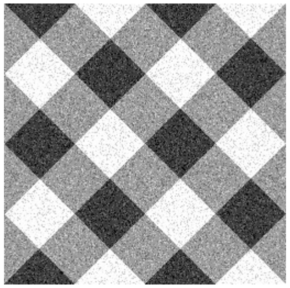
Noise terms

=

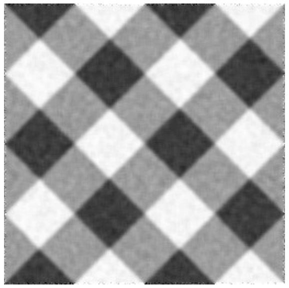
6	7	9	7	9
3	7	7	10	8
6	6	7	7	6
9	7	7	6	8
6	4	8	7	4

Noisy image

Noisy image:

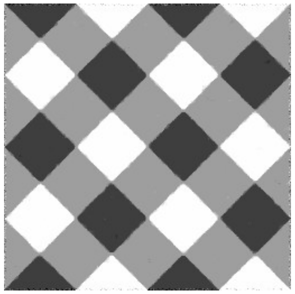


Mean filter:

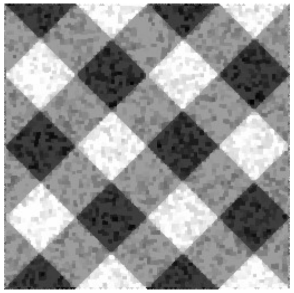


Advantage: Low computational cost.
Disadvantage: Blurs edge information.

Median filter:



Max-min-min-max filter:



Advantage: Preserves edge information.
Disadvantage: High computational cost.

Mean vs. median

- Mean is linear filter, while median is non-linear.
- Mean is affected by the outliers, but median is not.
- Mean is computationally less costly than median.
- Median can preserve edge much better than mean filter.
- Weighted averaging (with suitable set of weights) may lead to edge preserving smoothing by
  - sufficient intra-region smoothing
  - Insignificant inter-region smoothing

Linear neighborhood operation

Convolution: $g_{smooth}(r,c) = g_{noisy}(r,c) * h_{mask}(r,c)$

Mask: $h_{mask}(r,c)$ may be one such shown as follows.

$$\frac{1}{9}\begin{bmatrix}1&1&1\\1&1&1\\1&1&1\end{bmatrix} \quad \frac{1}{25}\begin{bmatrix}1&1&1&1&1\\1&1&1&1&1\\1&1&1&1&1\\1&1&1&1&1\\1&1&1&1&1\end{bmatrix} \quad \frac{1}{16}\begin{bmatrix}1&2&1\\2&4&2\\1&2&1\end{bmatrix} \quad \frac{1}{81}\begin{bmatrix}1&2&3&2&1\\2&4&6&4&2\\3&6&9&6&3\\2&4&6&4&2\\1&2&3&2&1\end{bmatrix}$$

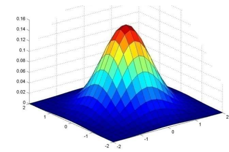
0.003	0.013	0.022	0.013	0.003
0.013	0.059	0.097	0.059	0.013
0.022	0.097	0.159	0.097	0.022
0.013	0.059	0.097	0.059	0.013
0.003	0.013	0.022	0.013	0.003

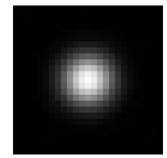
Non-linear neighborhood operation: Uses order statistic

Window: symmetric neighborhood (domain of the masks).

Gaussian Kernel

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





0.003	0.013	0.022	0.013	0.003
0.013	0.059	0.097	0.059	0.013
0.022	0.097	0.159	0.097	0.022
0.013	0.059	0.097	0.059	0.013
0.003	0.013	0.022	0.013	0.003

5 x 5, $\sigma = 1$

- Constant factor at front makes volume sum to 1 (can be ignored, as we should re-normalize weights to sum to 1 in any case)
- Replicates *isotropic* diffusion.

Source: C. Rasmussen

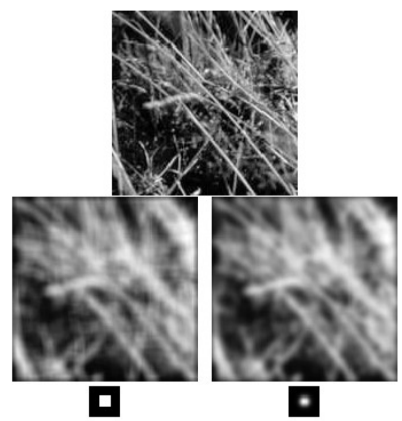
Gaussian smoothing

- Based on convolving a Gaussian kernel of size NxN with each and every pixel.
- A pixel's brightness value is determined by its own value as well as the values of its neighbor pixels.
- an appropriate definition of the transformation would be:

$$f_{t+1}(x,y) = f_t * G(x,y)$$

where $G(x,y) = \frac{1}{2\pi\sigma^2} * e^{-\frac{x^2+y^2}{2\sigma^2}}$ and $f_0(x,y) = f(x,y)$

Mean vs. Gaussian filtering



Edge preserving smoothing

- An edge divides two regions.
- A window covering single region may be characterized low variance of gray values.
- A window containing pixels from several region should have high variance.
- Neighborhood of a candidate pixel may be partitioned into various windows.

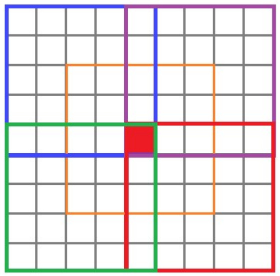
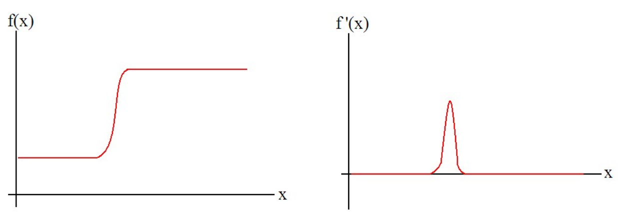


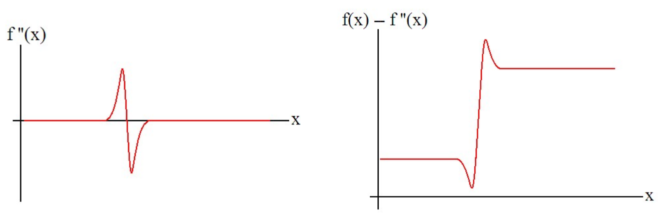
Image sharpening

- Also known as edge crispening
- Unblurs the edges and does not affect the interior
- Uses derivatives in spatial domain to highlight the change in graylevel at edges.
- High pass filter sharpens the edges giving emphasis to high frequency components.

Sharpening: 1D example



Sharpening: 1D example



Operators

- 1D second derivative (continuous domain) $\frac{d^2 f}{dx^2}$
- 2D second derivative (continuous domain) $\frac{\partial^2 f}{\partial x^2}+\frac{\partial^2 f}{\partial y^2}$
- 1D second difference (discrete domain)

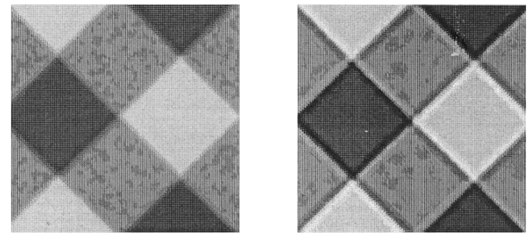
1	-2	1

- 2D second difference (discrete domain)

  Also called Laplacean operator.

0	1	0
1	-4	1
0	1	0

Sharpening: 2D example

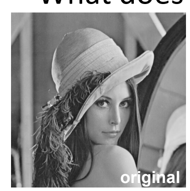


Sharpening: another approach

- Unlike smoothing, sharpening highlights the high frequency.
- Sharpening enhances edges (noise too!)
- Basic operator originates from smoothing itself.
- In frequency domain, sharpening can be achieved by high-pass filtering.

Sharpening

What does blurring take away?



–

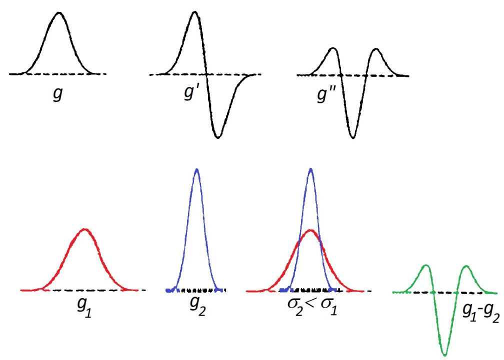
=

Let's add it back:

+ α

=

LoG ≈ DoG



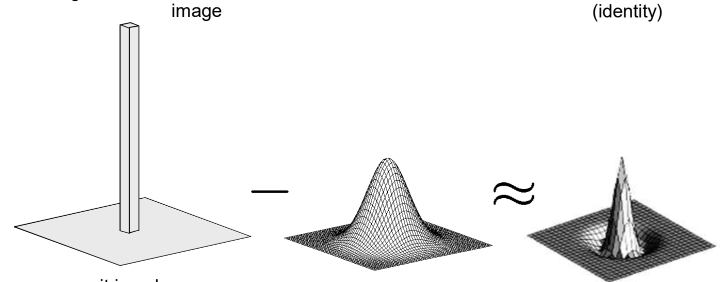
Unsharp mask filter

$$f + \alpha(f - f * G) = (1+\alpha)f - \alpha f * G \approx f * ((1+\alpha)\delta - G)$$

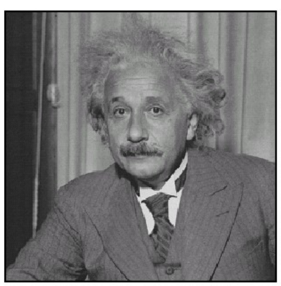
image

blurred image

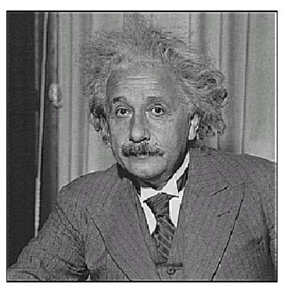
unit impulse (identity)



Sharpening: Results



before



after

Source: D. Lowe

Thank you !

Any question?