Non-Local Mean

A. Buades, B. Coll and J. -. Morel, "A non-local algorithm for image denoising"

Jingqi Zhang **Jefferson Zhong**

CONTENTS

Problem Definition

Problem Definition

What is Noise

- Unwanted information which deteriorates image quality
- Due to illumination, temperature, or signal processing disturbances etc.

Problem Definition

Common Noises

- Salt and Pepper Noise
  - Sparse white and black pixels
  - Caused by camera heat, dust etc.

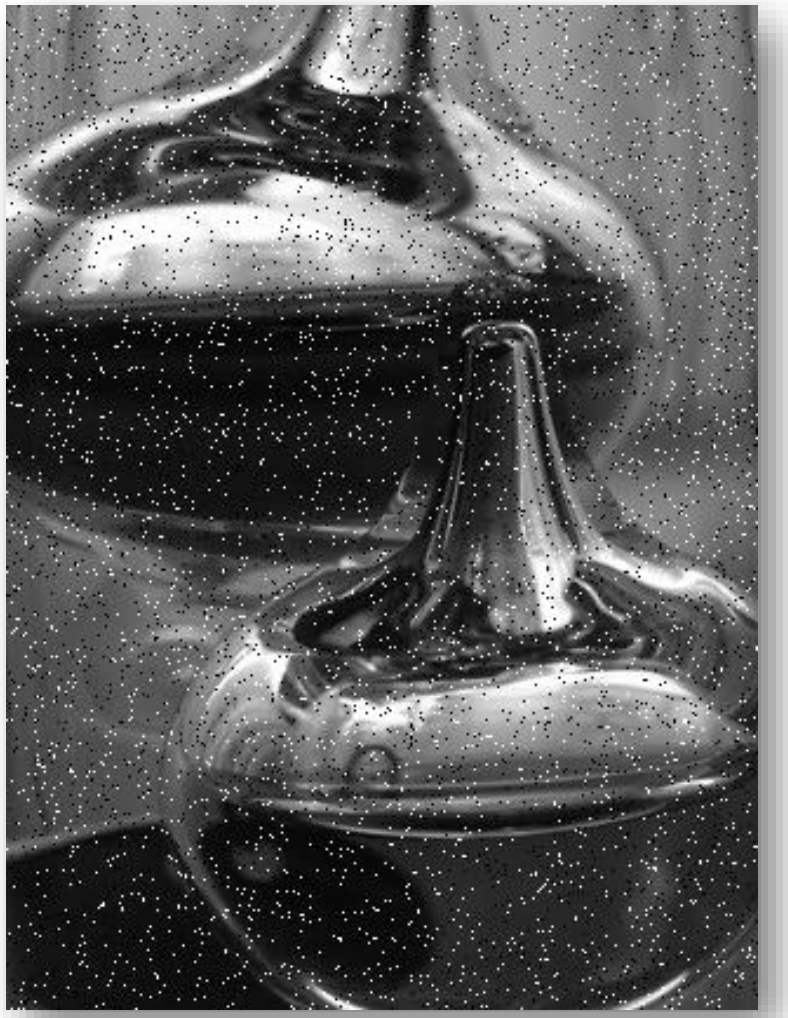


Image Source:
https://en.wikipedia.org/wiki/Image_noise#/media/File:Noise_salt_and_pepper.png

Problem Definition

Common Noises

- Gaussian Noise
  - Gaussian distributed
  - Good model of noise in many images
  - Poor lighting, heat, transmissions etc.

Image Source:
https://www.researchgate.net/figure/Noisy-image-Gaussian-noise-with-mean-and-variance-0005_fig2_252066070

Problem Definition

Image Denoising

Goal

Recover the original image from noisy image.

$$v = D_h v + n(D_h, v)$$

- v observed values
- D_h filter, $D_h v \rightarrow true\ values$ as $D_h \rightarrow perfect\ filter$
- $n(D_h, v)$ noise perturbation
  - Ideally would be gaussian white noise

Previous Work

Previous Work

Local Smoothing Filters

- Collect information from adjacent pixels to smooth out disturbance.

Denoising by Averaging

Local Smoothing Filters

Gaussian Filtering

- Optimal in flat parts of the image
- Blurred around edges and texture

Anisotropic Filtering

- The straight edges are well restored
- Flat and textured regions are degraded

Neighborhood Filtering

- Not robust when images is very noisy

Motivation

Local Filters Disadvantages

- **Visual Quality:**
  - Either edge or texture details are degraded

- **Method Noise:**
  - Smooths out even when the image has not much noise

What if we consider non locally?

Non-Local Mean

Non-Local Mean

- Denoising with weighted average across all pixels.

Definition

$$NL[v](i) = \sum_{j \in I} w(i,j)v(j)$$

- $v = \{v(i) | \ i \in I\}$ the input, a noisy image
- $NL[v](i)$ the estimated value
- $w(i,j)$ weight of similarity between pixel i and j

Weight of Similarity Motivation

- want to smooth out patch p
- Weights of q1 and q2 should have value larger than q3, since they are similar

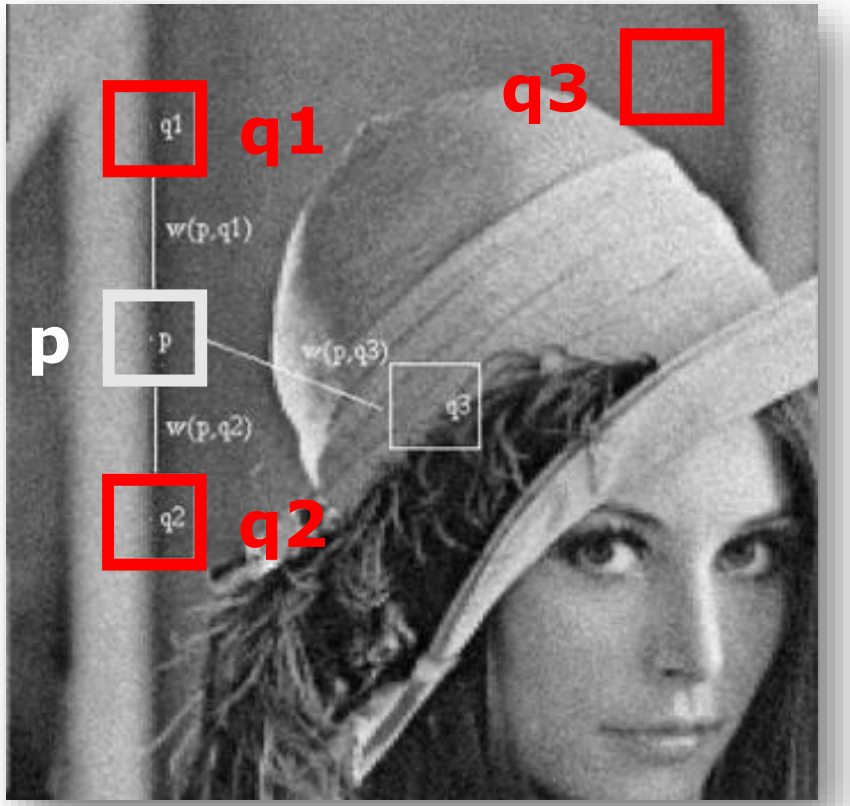


Image Source: A. Buades, B. Coll and J. -. Morel, "A non-local algorithm for image denoising"

Similarity Defined

- Q: When are two pixels i, j considered "similar" ?

- A: By euclidean distance of intensity gray level

$$\left\| v(N_i) - v(N_j) \right\|_{2,a}^2$$

- N_i a patch centered at pixel i
- $v(N_i)$ intensity gray level vector
- $a > 0$ the standard deviation of the Gaussian kernel

Weight of Similarity Motivation

- Also want to respect texture
- q1, q3 same intensity, but q1 preserves textures
- Pixels nearby have similar texture thus should be weighted more

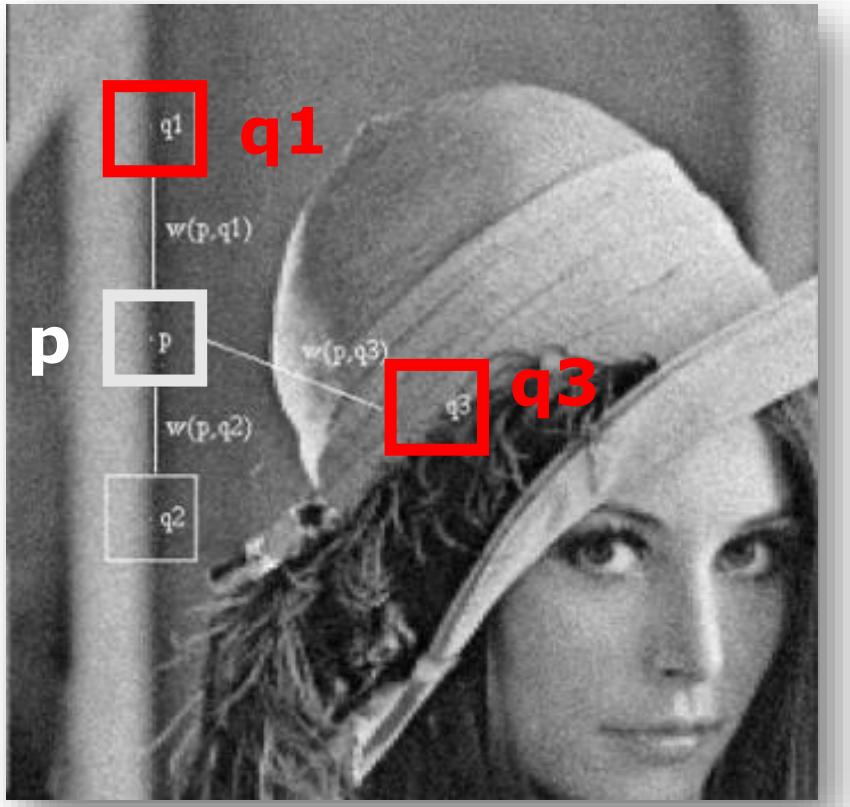
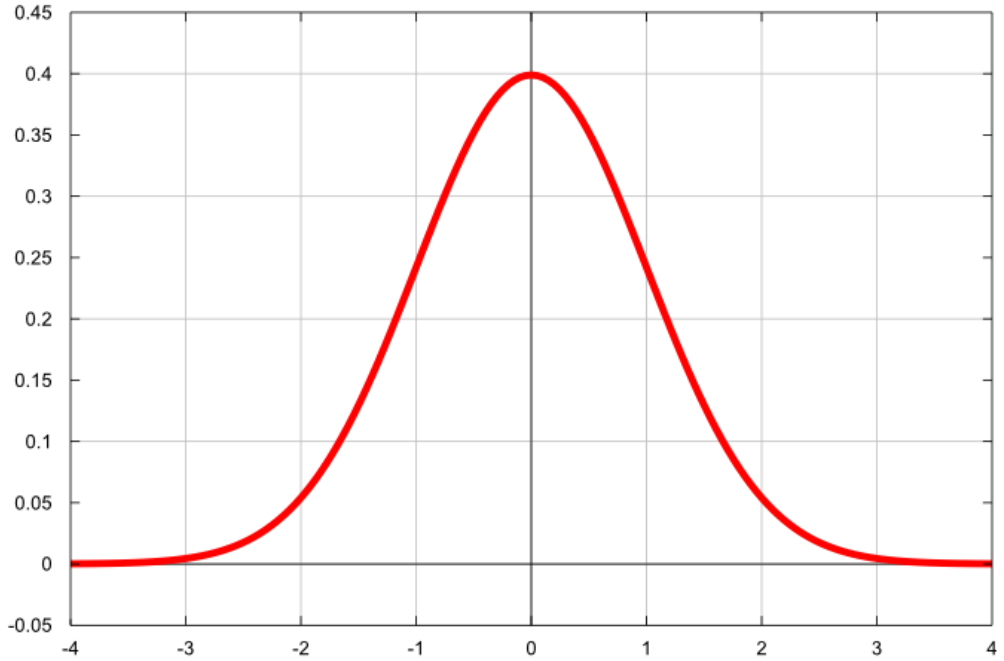


Image Source: A. Buades, B. Coll and J. -. Morel, "A non-local algorithm for image denoising"

Weight of Similarity Motivation

- Gaussian weighting
- Centered at pixel i
- As the distance increase, weighting decrease



Weights Defined

- Patches close together with a similar grey level have larger weights.

$$w(i,j) = \frac{1}{Z(i)} e^{-\frac{\|v(N_i) - v(N_j)\|_{2,a}^2}{h^2}}$$

- $Z(i) = \sum_{j \in I} e^{-\frac{\|v(N_i) - n(N_j)\|_{2,a}^2}{h^2}}$, the normalizing constant
- h, decay of exponential function, the degree of filtering

Weights Defined

- Patches with a similar grey level have larger weights

$$w(i,j) = \frac{1}{Z(i)} e^{-\frac{\|v(N_i) - v(N_j)\|_{2,a}^2}{h^2}}$$

Properties

- $0 \leq w(i,j) \leq 1$
- $\sum_{j \in I} w(i,j) = 1$
- $E\left[\|v(N_i) - v(N_j)\|_{2,a}^2\right] = \|u(N_i) - u(N_j)\|_{2,a}^2 + 2\sigma^2$

Non-Local Mean

Putting together

$$NL[v](i) = \sum_{j \in I} \frac{1}{Z(i)} e^{-\frac{\|v(N_i) - v(N_j)\|_{2,a}^2}{h^2}} v(j)$$

- $v = \{v(i) | \, i \in I\}$ the input, a noisy image
- $NL[v](i)$ the estimated value

Implementation

Implementation

Code

- **GitHub:** https://github.com/surprise777/CSC320

- **Complexity:** $O(wmN^2)$
  - w search window size, e.g. $w = 441$ for 21×21 pixels windows
  - m patch size, e.g. $m = 49$ for 7×7 patch
  - N^2 number of pixels

Comparisons

Comparisons – Visual Quality

Preserves more detail on edge and texture

1) Noisy Image
2) Gauss Filtering
3) Anisotropic Filter
4) Total Variation
5) Neighborhood Filtering
6) Non-Local Mean

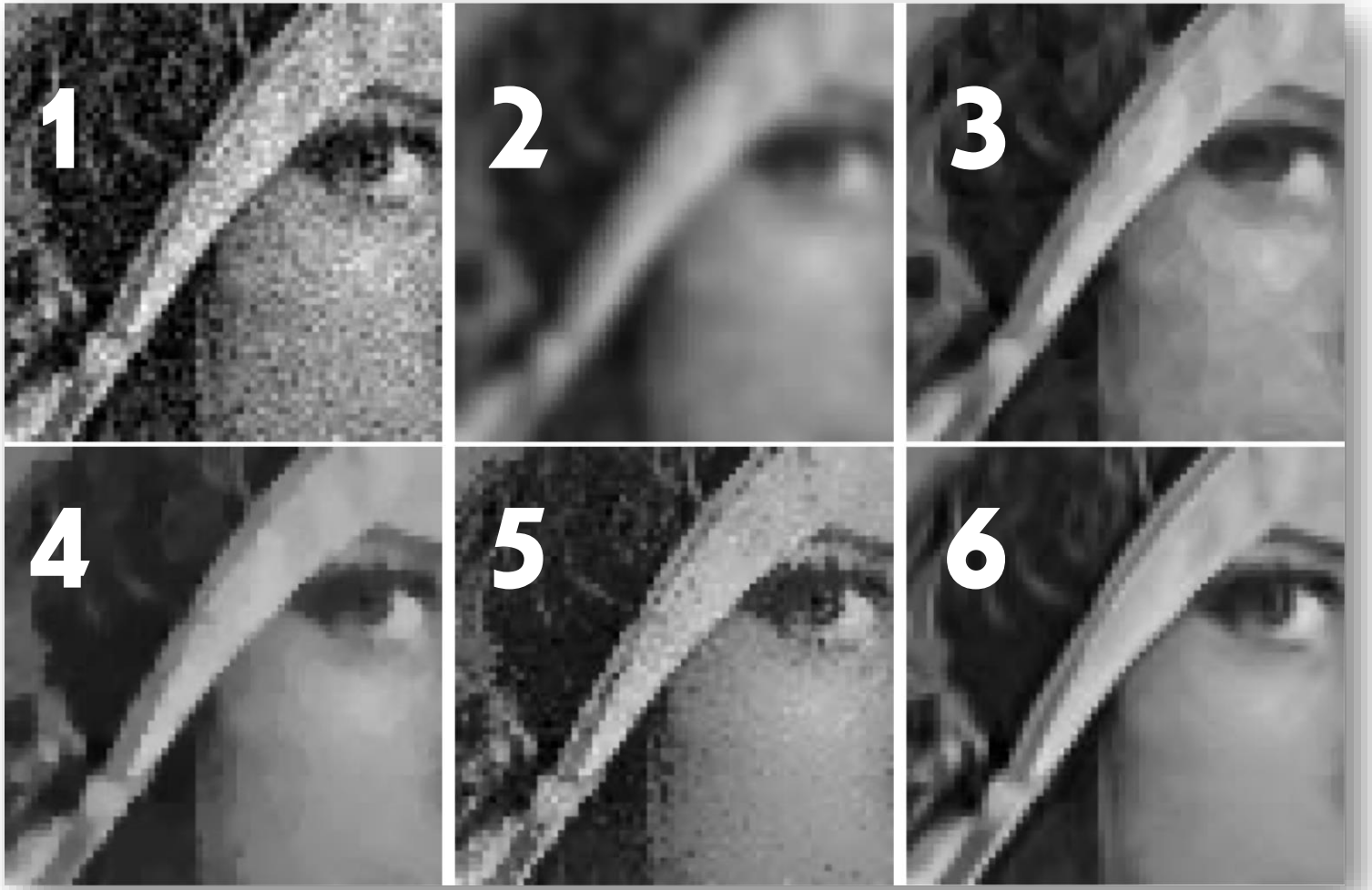


Image Source: A. Buades, B. Coll and J. -. Morel, "A non-local algorithm for image denoising"

Comparisons – Method Noise

Method noise looks like Gaussian white noise

1) Noisy Image
2) Gauss Filtering
3) Anisotropic Filter
4) Total Variation
5) Neighborhood Filtering
6) Non-Local Mean

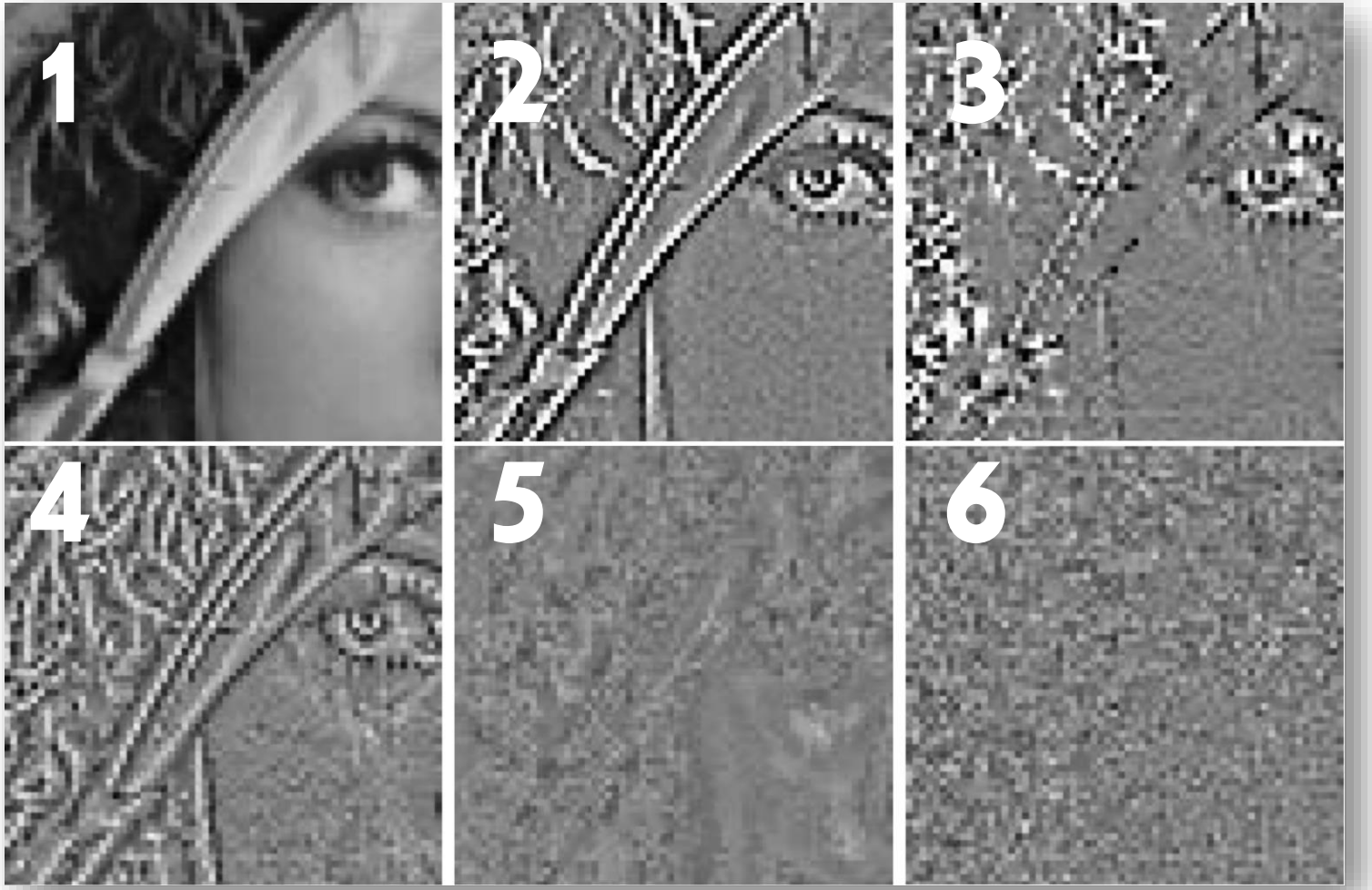


Image Source: A. Buades, B. Coll and J. -. Morel, "A non-local algorithm for image denoising"

Limitations & Further Work

Limitations & Further Works

01 Efficiency Concern

- Cannot be used on device that requires instant result e.g. phone

- When number of patches is too large or the image is too noisy, will require infeasible time to detect similarities

- Further work on improving runtime:
  - Thaipanich T, Oh BT, Wu PH, Xu DR, Kuo CCJ (2010) Improved image denoising with adaptive nonlocal means (ANL-means) algorithm.
  - Karnaukhov, V.N., Mozerov, M.G. Fast Non-Local Mean Filter Algorithm Based on Recursive Calculation of Similarity Weights.

Limitations & Further Work

02 Types of Noise to Filter

- Original algorithm designed toward additive noise e.g. Gaussian noise, but not so much in multiplicative noise e.g. Speckle noise
  - Speckle noise often appear in Ultrasonic Imaging

- Further work on extension toward multiplicative noise:
  - Teuber T., Lang A. (2012) Nonlocal Filters for Removing Multiplicative Noise.
  - Radlak K., Smolka B. (2014) Adaptive Non-local Means Filtering for Speckle Noise Reduction

THANKS