Convergence of an iterative nonlinear scheme For denoising of piecewise constant images

Debwashis Borman

Universität Göttingen

April 19, 2024

Table of Contents

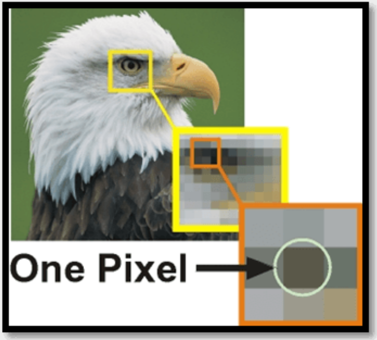
Introduction: some basic concepts

What is Pixel?

- Pixel: Pixel, the fundamental building block of digital images smallest controllable element of a picture represented on a screen or digital display. Each pixel contains color information, usually represented as combinations of three primary colors: red, green, and blue (RGB).

What is Resolution?

- The resolution of an image is measured in pixels and typically represented by the dimensions width x height. For instance, an image resolution of 1920 x 1080 means there are 1920 pixels in each row (width) and 1080 pixels in each column (height) of the image.



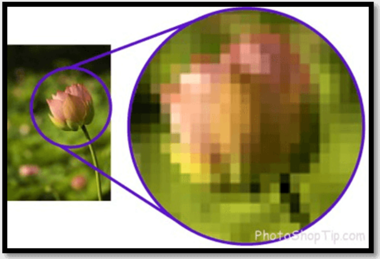


Figure: Pixel visualization

- **1-Channel Images (Grayscale)**: Each pixel is represented by a single intensity value, typically between 0 (black) and 255 (white). The pixel holds information about its intensity or brightness. Example:

$$\begin{pmatrix} 50 & 100 & 150 \\ 200 & 255 & 0 \\ 80 & 120 & 90 \end{pmatrix}$$

- **3-Channel Images (RGB)**: Each pixel is represented by three values, corresponding to the Red, Green, and Blue (RGB) channels. Each channel has intensity values ranging from 0 to 255. The pixel holds information about the intensity of red, green, and blue, which combine to form the pixel's final color. A 3-D matrix for RGB values:

$$\begin{pmatrix} (255, 0, 0) & (0, 255, 0) \\ (0, 0, 255) & (255, 255, 255) \end{pmatrix}$$

- Image noise can be understood as the unwanted random variations or disturbances in an image that degrade its quality. Similar to the concept of noise in signal processing, where noise distorts the true signal, image noise interferes with the accurate representation or interpretation of an image.

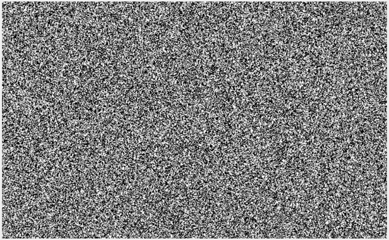


Figure: Different Types of Noise

Types of Image Noise

- White Noise: In images, the pixels of a white noise image is assumed to be independent random variables with a uniform probability distribution over some set interval.
- Shot noise or Poisson noise is a type of noise which can be modeled by a Poisson process.
- Salt and Pepper Noise: This kind of noise if added to images will result in dark pixels in bright regions and bright pixels in dark regions, with a random frequency of occurrence.

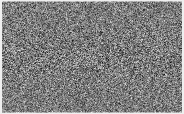


Figure: white noise

Figure: Salt and Pepper Noise

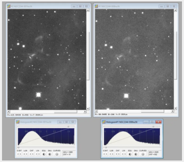


Figure: Short noise

Common Sources of Image Noise

- Sensor Noise: For example, in digital cameras, sensor noise can occur due to electronic disturbances or thermal noise within the camera's sensor.
- Environmental Factors: Poor lighting conditions can result in more noise in the image
- Transmission Errors: Lossy compression methods that discard some data to reduce the file size, or due to interference and loss in signal strength during wireless transmission.

Impact of Noise on Image Quality

- Loss of Details: Blur or distort finer details in an image, making it harder to distinguish certain elements or patterns. This can be particularly problematic in fields like medical imaging or remote sensing, where fine details can be crucial.
- Reduced Visibility of Important Features: Difficult to identify certain objects or features, like a tumor in a medical image, or specific landforms in a satellite image.
- Increased Difficulty in Image Processing: Uncertainty introduced by the noise make segmentation, feature extraction, and object recognition more challenging.

Goals of Image Denoising

- Noise Reduction
- Edge Preservation
- Maintaining Image Fidelity

Denoising Techniques:

- Filtering Methods: Mean filter, Median filter, Wiener filter, Bilateral filter, Gaussian filter etc.
- Statistical Methods: Maximum Likelihood Estimation, Wavelet-based methods.
- Iterative Algorithms: Total Variation method, This paper!!!
- Machine Learning Models: PRIDNet, DnCNN , Generative adversarial network etc.
- In Mobile: Google Night Sight, Apple Deep Fusion

- Piecewise constant images are a class of images where the image intensity remains constant over different regions, and these regions are separated by sharp edges or boundaries. These are typical characteristics seen in many types of images, such as cartoon images, medical images (e.g., MRI, CT scans), or other images with clearly defined areas of uniform color or intensity.

Figure: Example of Piecewise constant image

Pixel Neighbourhood- 2D

- A pixel's neighborhood refers to the set of pixels that are located in a certain vicinity around it.
- 4-connected: They are both on and are connected along the horizontal or vertical direction.
- 8-connected: This means that if two adjoining pixels are on, they are part of the same object, regardless of whether they are connected along the horizontal, vertical, or diagonal direction.

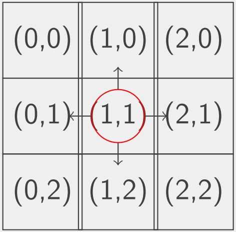


Figure: 4-Neighbourhood

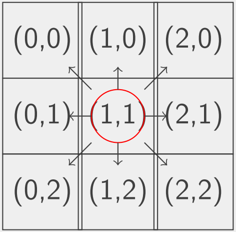


Figure: 8-Neighbourhood

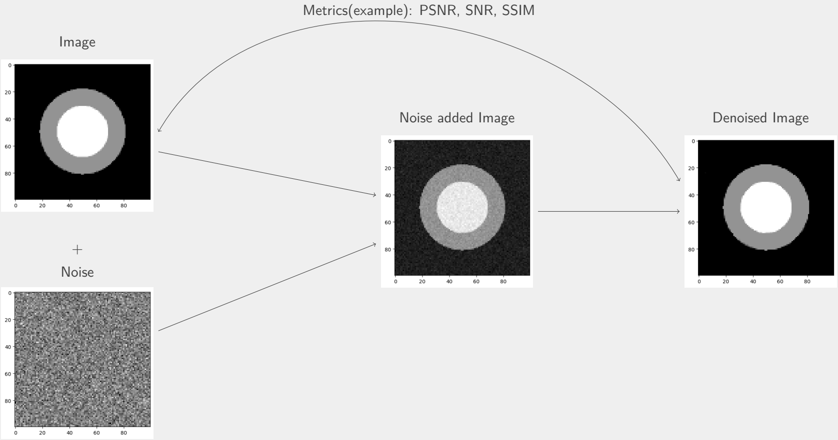


Figure: Image Denoising

Mean Filter

Given an input image I, we define its pixels at position (x, y) as $I(x, y)$. For each output pixel $I'(x, y)$ we find the mean value with a local region R surrounding (x, y)

$I'(x, y) = \text{mean}\{I(x+i, y+j) | (i, j) \in R\}$

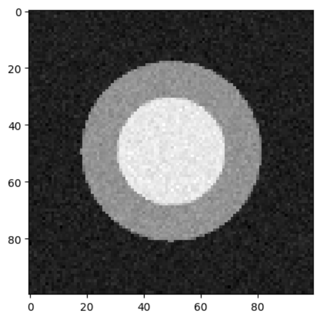


Figure: Image

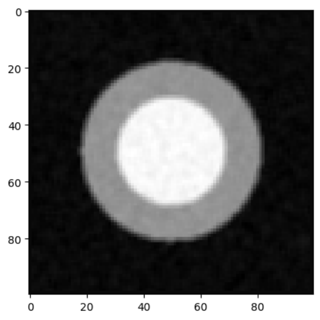


Figure: After Mean Filter

Median Filter

Given an input image I, we define its pixels at position (x, y) as $I(x, y)$. For each output pixel $I'(x, y)$ we find the mean value with a local region R surrounding (x, y)

$I'(x, y) = \text{median } \{I(x + i, y + j)|(i, j) \in R\}$

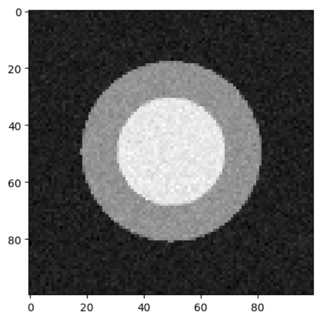


Figure: Image

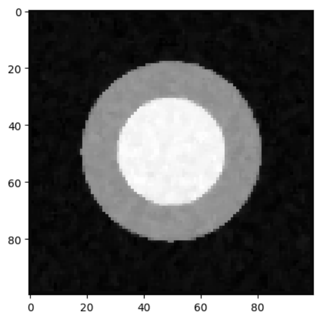


Figure: After Median Filter

Iterative scheme

- **Shrinkage Parameter (θ):** The shrinkage parameter, often represented by θ, is used in the shrinkage function $T_\theta(u)$ in the iterative scheme. Shrinkage is used to reduce the influence of noise while preserving the essential features of the image.
- **Smoothing Parameter (α):** Smoothing is a process that reduces the local variations in an image, making neighboring pixel intensities more similar. The smoothing parameter, often represented by α, is a factor that controls the extent of smoothing applied to the image in each iteration.

- Given an observed image $u_0 = (u^0_{i,j}), i = 0, \ldots, N_1 - 1; j = 0, \ldots, N_2 - 1$
- noise $n = (n_{i,j})_{i=0,\ldots,N_1-1;j=0,\ldots,N_2-1}$, n is Normally-distributed noise $n_{i,j} \in N(0, \sigma^2)$
- where $u_0 = f + n$
- shrinkage parameter $\theta > 0$ and a smoothing parameter $0 < \alpha \leq \frac{1}{6}$. iteration for $k = 0, 1, \ldots$ For $i = 0, \ldots, N_1 - 1$ and $j = 0, \ldots, N_2 - 1$, compute the new image values as:

$$u^{k+1}_{i,j} = u^k_{i,j} + \alpha \sum_{\substack{(l,m)\in N(i,j)\\(l,m)\neq(i,j)}} \frac{T_\theta(u^k_{l,m} - u^k_{i,j})}{(l-i)^2 + (m-j)^2} \tag{1}$$

- Here T_θ function is defined as $T_\theta(x) = \begin{cases} 0 & \text{if } |x| \geq \theta \\ x & \text{if } |x| < \theta \end{cases}$

- The neighborhood $N(i,j)$ is defined as $N(i,j) := \{(l,m) : |i-l| \leq 1; |j-m| \leq 1\}$, denotes the set of pixels (l,m) that fall within a (3×3) neighborhood centered at pixel (i,j).
- Then we can rewrite it:

$$u_{i,j}^{k+1} = u_{i,j}^{k} + \alpha \sum_{\substack{r=-1 \\ r \neq 0}}^{1} \sum_{\substack{s=-1 \\ s \neq 0}}^{1} \frac{T_\theta(u_{i+r,j+s}^{k} - u_{i,j}^{k})}{r^2 + s^2} \tag{2}$$

Let us now consider the properties and convergence of the iteration scheme. For convenience, we restrict our considerations to the scheme (2). All ideas can be simply transferred to more complex schemes. We choose one index for pixel numbering of the digital image u^k. Put $N := N_1 \cdot N_2$ and $n = i + N_1 j$, $i = 0, \ldots, N_1 - 1$, $j = 0, \ldots, N_2 - 1$, such that the pixel n corresponds to (i, j). Then the iteration scheme (2) can be written in matrix-vector form as $u^{k+1} = A^k u^k$, where $u^k = (u_0^k, \ldots, u_{N-1}^k)^T$ and where $A^k = (A_{n,p}^k)_{N \times N}$ is a sparse matrix of the form

$$A_{n,p}^k := \begin{cases} 1 - \kappa_n \alpha & \text{for } p = n, \\ \alpha & \text{for } p \equiv \{n-1, n+1, n-N_1, n+N_1\} \pmod{N} \\ & \quad \text{and } |u_n^k - u_p^k| < \theta, \\ \alpha/2 & \text{for } p \equiv \\ & \{n-1+N_1, n+1+N_1, n+1-N_1, n-1-N_1\} \\ & \quad \pmod{N} \quad \text{and } |u_n^k - u_p^k| < \theta, \\ 0 & \text{elsewhere.} \end{cases}$$

Let's $u^1 = \begin{bmatrix} 96 & 182 & 45 \\ 199 & 184 & 162 \\ 138 & 212 & 201 \end{bmatrix}$.

Then after apply the conditions, $\mathsf{A}^1 =$

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1-\frac{3\alpha}{2} & 0 & \frac{\alpha}{2} & \alpha & \frac{\alpha}{2} & 0 & 0 & \frac{\alpha}{2} \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{\alpha}{2} & 0 & 1-\frac{3\alpha}{2} & \alpha & 0 & 0 & \frac{\alpha}{2} & \frac{\alpha}{2} \\
0 & \alpha & 0 & \alpha & 1-\frac{9\alpha}{2} & \alpha & 0 & \alpha & \frac{\alpha}{2} \\
0 & \frac{\alpha}{2} & 0 & 0 & \alpha & 1-2\alpha & \frac{\alpha}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{\alpha}{2} & 1-\frac{\alpha}{2} & 0 & 0 \\
0 & 0 & 0 & \frac{\alpha}{2} & \alpha & 0 & 0 & 1-\frac{3\alpha}{2} & \alpha \\
0 & \frac{\alpha}{2} & 0 & \frac{\alpha}{2} & \frac{\alpha}{2} & 0 & 0 & \alpha & 1-\frac{3\alpha}{2}
\end{bmatrix}.$$

Here κ_n (with $0 \leq \kappa_n \leq 6$) is chosen such that the sum of entries in the nth row of A^k is 1. Now we observe the following properties of the iteration matrix A^k.

1. The number of nonzero entries in each row (column) of A^k is at most 9.
2. For $\alpha \leq \frac{1}{6}$, all entries of A^k are non-negative, i.e., $A^k \geq 0$.
3. With $1 := (1, \dots, 1)^T \in \mathbb{R}^N$, we have $A^k 1 = 1$.
4. The iteration matrix A^k is symmetric, i.e., $A^k = (A^k)^T$.

- By suitable ordering of rows and columns of A^k, we can transfer A^k into a block diagonal matrix, where the nth and the pth row (column) belong to the same block if there exists a sequence of indices $n_1, n_2, \dots, n_\nu$ such that
  $n_1 \in N(n), n_2 \in N(n_1), \dots, p \in N(n_\nu)$ and if
  $|u_n^k - u_{n_1}^k| < \theta, |u_{n_1}^k - u_{n_2}^k| < \theta, \dots, |u_{n_\nu}^k - u_p^k| < \theta$ holds.
- The ordering of A^k into blocks defines a partition S^k of the pixel set $\{0, \dots, N-1\}$ into different sets, $S^k = \{S_m^k\}_{j=1}^{J^k}$, separating pixels whose pixel values differ strongly enough. Two pixels n, n' are in the same set S_m^k if, in the above procedure, the nth and the n'th row of A^k are transferred into the same block matrix A_m^k.

- Let's consider this example($0 < \epsilon < \theta$):

$$\begin{pmatrix} 0 & \theta+\epsilon & \theta+\epsilon & \theta+\epsilon \\ 0 & \theta+\epsilon & 2\epsilon & \theta+\epsilon \\ 0 & \theta+\epsilon & \theta+\epsilon & \theta+\epsilon \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Partition/Block
positions:$\{\{0,4,8,12,13,14,15\},\{1,2,3,5,6,7,9,10,11\}\}$
Considering elements positions like- 0^{th} =0,1^{st} =$\theta+\epsilon$ ....so on

- Let's consider a new example:

$$u^0 = \begin{bmatrix} 96 & 120 & 100 \\ 70 & 180 & 140 \\ 50 & 100 & 180 \end{bmatrix}$$

Blocks coordinate: 3

$$[(0,0),(2,1),(0,2),(0,1),(1,2)],[(1,0),(2,0)],[(1,1),(2,2)]$$

- $$u^1 = u^k = \begin{bmatrix} 100 & 100 & 100 \\ 56 & 180 & 138 \\ 56 & 100 & 180 \end{bmatrix}$$

Blocks coordinate: 4

$$[(0,0),(2,1),(0,2),(0,1)],[(1,0),(2,0)],[(1,1),(2,2)],[(1,2)]$$

- Taking means per block and updating it doesn't change pixels further.

Let $S^k = \{S^k_m\}_{m=1}^{J^k}$ be the partition of the pixel set, Now let us consider the new partition S^{k+1} is different from S^k, then for a fixed set $S^k_m \in S^k$, the following types of changes are possible:

1. S^k_m is divided into two or more subsets $S^{k+1}_{m1}, S^{k+1}_{m2}, \ldots$.
2. S^k_m is united with one or more neighboring sets, i.e., $S^{k+1}_{m1} = S^k_m \cup S^k_m{}' \cup \ldots \cup S^k_m\nu$.
3. S^k_m is united with one or more neighboring subsets of $S^k_m{}'$ to obtain a new set S^{k+1}_{m1} of S^{k+1}.
4. S^k_m is divided into two or more subsets $S^{k+1}_{m1}, \ldots$ which are themselves united with one or more neighboring sets $S^k_m{}'$ (or neighboring subsets of $S^k_m{}'$).

- let

$$\mu^k_m = \frac{1}{\#S^k_m} \sum_{n \in S^k_m} u^k_n, \quad m = 1, \ldots, J_k \tag{3}$$

$\#S^k_m$ denotes number of the elements on a block.

Theorem

Now let's consider a noisy image u^0. The iteration scheme (2) with $\alpha \leq \frac{1}{6}$ and starting with the noisy image u^0. Then, after a finite number of k iterations, the partition $S_k = (S_m^k)_{m=1}^{j_k}$ of pixels corresponding to the blocks $A_m^k,\ m = 1, \dots, J_k$ of the ordered iteration matrix A_k is settled and will not change further, i.e., there exists a finite k such that $S^k = S^{k+\nu},\ \nu = 1, 2, \dots$.
The iteration scheme (2) will converge to the spatial average of u^k in all subsets S_m^k:

$$\lim_{\nu \to \infty} u^{k+\nu} = \sum_{m=1}^{J_k} \mu_m^k \chi_{S_m^k}$$

where μ_m^k is in (3) and where $\chi_{S_m^k}$ is the characteristic function corresponding to the index set S_m^k, i.e.,

$$\chi_{S_m^k}(n) = \begin{cases} 1 & \text{if } n \in S_m^k \\ 0 & \text{if } n \notin S_m^k \end{cases}$$

Algorithm and hints on implementation

Algorithm

Algorithm for Denoising Piecewise Constant Images

The complete algorithm for denoising piecewise constant images consists of three steps. Let $u^0 = (u^0_{i,j})_{i=1}^{N_1-1}, j_{i=1}^{N_2-1}$ be the given noisy image.

1. For a fixed thresholding parameter $\theta > 0$, a smoothing parameter $\alpha \leq \frac{1}{6}$, and a fixed number $K \in \mathbb{N}$, perform the iteration:

$$u_{i,j}^{k+1} = u_{i,j}^{k} + \alpha \sum_{\substack{r,s=-1 \\ (r,s)\neq(0,0)}}^{1} \frac{T_\theta(u^k_{i+r,j+s} - u^k_{i,j})}{r^2 + s^2}$$

for $k = 0, 1, 2, \ldots, K-1$, using periodic boundary conditions

Algorithm for Denoising Piecewise Constant Images

2. Apply the following mean value procedure. Establish a new iteration matrix A_K with a suitable shrinkage parameter θ_1. Take the partition $(S_m^K)_{m=1}^{J_K}$ determined by this iteration matrix A_K. Compute for $m = 1, \ldots, J_K$ the mean values:

$$\mu_m^K = \frac{1}{\# S_m^K} \sum_{n \in S_m^K} u_n^K$$

Replace each value u_n^K with $n \in S_m^K$ by μ_m^K.

Algorithm for Denoising Piecewise Constant Images

3. Using the partition $(S_m^K)_{m=1}^{J_K}$, apply the following median value procedure. All pixel values belonging to subsets of the partition with less than 6 components are replaced by the median of their neighbor pixel values.

Methodology

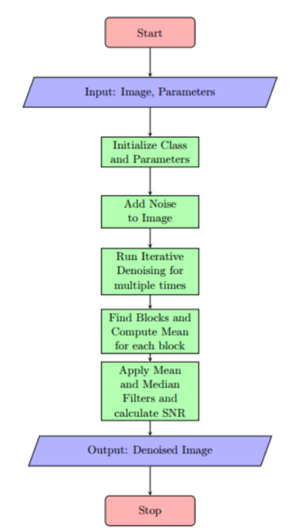


Figure: Flowchart of the Plonka-Hoch and Jianwei Ma Denoising Process

Performance

- the following equation has been used for calculating "Signal-to-noise ratio".

$$\mathsf{SNR} = 20\log_{10}\frac{\|\mathbf{f}-\bar{\mathbf{f}}\|}{\|\mathbf{n}\|} \tag{4}$$

- For finding good starting points for shrinkage parameters for the denoising process (theoretically), determining contrast "c" in an image which is the difference between the maximum and minimum pixel intensity is necessary. It quantifies how much the intensity varies across the image, with higher values indicating more variation. "c" is defined as -

$$c = \min\left\{\left|f_{i,j} - f_{i',j'}\right|, (i',j') \in N(i,j), f_{i,j} \neq f(i',j')\right\} \tag{5}$$

- For small contrast-to-noise ratio i.e. $\frac{c}{\sigma} \leq 1.5$, initiate the iteration process (2) with $\theta = \infty$.
- The smoothing parameter α need not be chosen very small. An α value in the range of $\left[\frac{1}{10}, \frac{1}{6}\right]$ provides quick smoothing while preserving the convergence result.
- The optimal number of iterations K is image-dependent and varies with the signal-to-noise ratio.
- If the partition process isn't complete after the first step, paper recommends to use a new shrinkage parameter $\theta_1 = \frac{\theta c}{10\sigma}$ (see page no. 15).

- Image with good contrast-to-noise ratio: $\theta = 30$, $\theta_1 = 4$

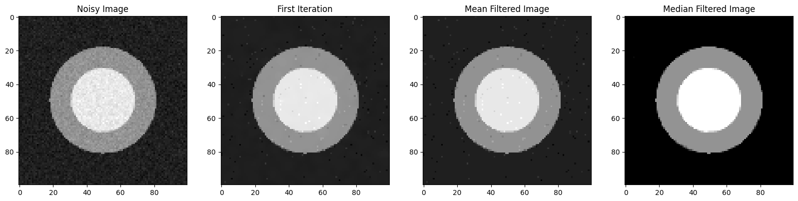


Figure: Denoising image with different gray level

- Influence of the shrinkage parameter on an image with a good contrast-to-noise ratio: $\theta = 40$, $\theta_1 = 40$

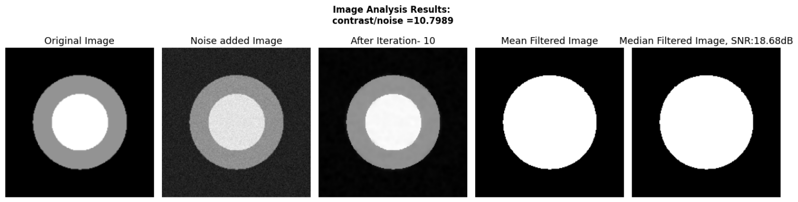


Figure: Denoising image with different gray level - Shrinkage parameter influence

- Example for small contrast-to-noise ratio image: $\frac{c}{\sigma} < 1.5$ and $\theta = \infty$, $\theta_1 = 4$

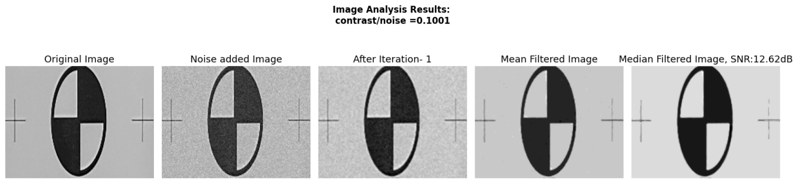


Figure: Denoising image with different gray level - low contrast-to-noise ratio

- Here is another example of a small contrast-to-noise ratio image: $\frac{c}{\sigma} < 1.5$ and $\theta = \infty$, $\theta_1 = 6$

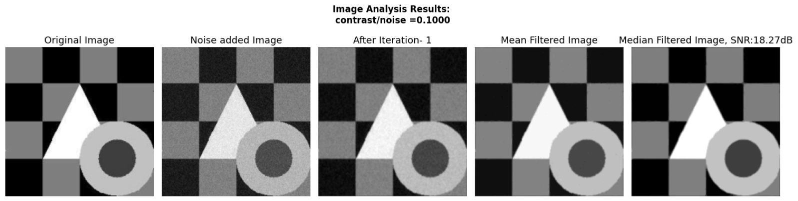


Figure: Denoising image with multiple gray level - low contrast-to-noise ratio

- Does an increase in the number of iterations lead to significantly improved denoising results? This is something we will examine closely. In tandem, we'll also monitor the evolution of block sizes.

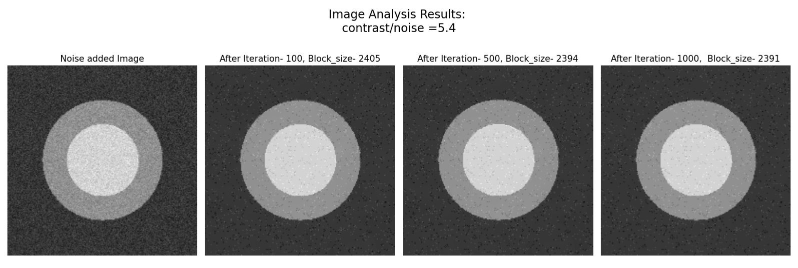


Figure: Denoising image with multiple gray level - after $100, 500, 1000$ iteration

- In medical imaging, such as MRI or CT scans, noise reduction is crucial for accurate diagnosis. One of such example can be found here [1].
- Images captured by remote sensing satellites or aerial drones often suffer from noise due to atmospheric conditions or sensor limitations.
- In industries such as manufacturing or quality control, image denoising is essential for accurate defect detection and inspection.
- In Fingerprint Denoising algorithm can highlight these features by reducing the noise within each region.

Thank You!

GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

[1] G. Ongie and M. Jacob, "Off-the-grid recovery of piecewise constant images from few fourier samples," *SIAM Journal on Imaging Sciences*, vol. 9, no. 3, pp. 1004–1041, 2016.