Experiment 4: Noise Models and Image Restoration using Filters

Aim

To study various noise models and restore a degraded image using the following spatial domain filters:

- Arithmetic Mean Filter
- Midpoint Filter
- Alpha-Trimmed Mean Filter

Software Required

- Python
- OpenCV
- NumPy
- Matplotlib

Theory

Noise Models

Noise is an unwanted disturbance that degrades image quality. Common noise models include:

- **Gaussian Noise**: Caused by sensor noise and electronic circuit noise.
- **Salt and Pepper Noise**: Appears as random black and white pixels due to transmission errors.
- **Uniform Noise**: Noise values are uniformly distributed over a range.

Noise models are mathematically characterized using probability density functions (PDFs).

Image Restoration

Image restoration aims to recover an original image from a degraded version by reducing or removing noise using appropriate filters.

—

Arithmetic Mean Filter

Theory

The arithmetic mean filter replaces each pixel value with the average of the pixel values in a neighborhood window of size $m \times n$.

$$\hat{f}(x, y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s, t)$$

This filter smooths the image and reduces Gaussian noise but may blur edges.

Algorithm

1. Select a window of size $m \times n$.
2. Move the window over the entire image.
3. Compute the average of pixel values inside the window.
4. Replace the center pixel with the computed average.

Pseudo Code (Python Style)

```
1. Read the noisy image
2. Define window size
3. for each pixel in image:
       • Extract neighborhood window
       • Compute mean of window pixels
       • Assign mean value to center pixel
4. Display restored image
```

—

Midpoint Filter

Theory

The midpoint filter replaces each pixel with the average of the maximum and minimum pixel values in the neighborhood.

$$\hat{f}(x, y) = \frac{1}{2}\left[\max(g(s, t)) + \min(g(s, t))\right]$$

This filter is effective for reducing uniform noise.

Algorithm

1. Select a neighborhood window.
2. Find the maximum and minimum pixel values in the window.
3. Compute their average.
4. Replace the center pixel with the midpoint value.

Pseudo Code (Python Style)

```
1. Read noisy image
2. Define window size
3. for each pixel:
       • Extract window
       • Find maximum and minimum values
       • Compute midpoint
       • Replace center pixel
4. Display restored image
```

—

Alpha-Trimmed Mean Filter

Theory

The alpha-trimmed mean filter removes a specified number of highest and lowest pixel values from the neighborhood before calculating the mean.

$$\hat{f}(x, y) = \frac{1}{mn - d} \sum g_r(s, t)$$

where d is the number of trimmed pixels.
This filter is effective for images corrupted by both Gaussian and impulse noise.

Algorithm

1. Select window size and trimming parameter d.
2. Sort pixel values in the window.
3. Remove $d/2$ lowest and $d/2$ highest values.
4. Compute the mean of remaining values.
5. Replace the center pixel with the computed mean.

Pseudo Code (Python Style)

```
1. Read noisy image
2. Define window size and trimming value $d$
3. for each pixel:
       • Extract neighborhood window
       • Sort pixel values
       • Remove extreme values
       • Compute mean of remaining pixels
       • Assign mean to center pixel
4. Display restored image
```

—

Result

Thus, various noise models were studied and the degraded image was successfully restored using Arithmetic Mean, Midpoint, and Alpha-Trimmed Mean filters.