SOME TYPE OF NOISESAND DENOISING STRATEGIES USING SPATIAL FILTERS IN DIGITAL IMAGE PROCESSING

In this study, we are going to evaluate some types of noises that are applicable to the following classic (spatial) filters: gaussian, median and mean filters.

In this study, we are going to focus ourselves on:

- **Gaussian noise**
- **Salt-and-pepper noise**
- **Speckle noise**

1. **GAUSSIAN NOISE:** Gaussian noise follows a normal distribution, characterized by random intensity variations with a mean of zero and a specified standard deviation σ. It is additive, meaning each pixel's intensity is altered by a random value drawn from the Gaussian distribution.
   Causes: Gaussian noise typically arises from sensor noise in low-light conditions, thermal noise in camera electronics, or quantization errors during digitization.
   Effects: It introduces fine-grained distortions across the image, reducing contrast and affecting edge detection and feature extraction.

2. **SALT-AND-PEPPER NOISE**: Salt-and-Pepper Noise Salt-and-pepper noise, also known as impulse noise, manifests as random pixels being set to extreme values (example: 0 for black & 255 for white in an 8-bit grayscale image). It is characterized by sparse, high-contrast distortions.
   Causes: This noise is caused by faulty sensors, transmission errors, or bit errors in analog-to-digital conversion. .
   Effects: It creates noticeable black and white spots, degrading visual quality and impacting tasks like object recognition.

3. **SPECKLE NOISE:** Speckle noise is multiplicative, meaning it scales with the image: image intensity, often appearing as a granular pattern.
   Causes: Speckle noise arises from interference of coherent waves, such as in ultrasound, radar, or laser-based imaging systems.
   Effects: It introduces a mottled appearance, reducing image clarity and complicating feature extraction.

Implementation procedure of the type of noises

The below screenshot shows how the various noises stated above can be implemented using **python – cv2** and **numpy**:

```python
import cv2
import numpy as np

# Load clean image
img = cv2.imread('clean.png', 0)

# Add Gaussian noise
gaussian_noise = img + np.random.normal(0, 25, img.shape).astype(np.uint8)

# Add Salt-and-Pepper noise
def add_salt_pepper(image, prob=0.05):
    noisy = np.copy(image)
    salt = np.random.rand(*image.shape) < prob/2
    pepper = np.random.rand(*image.shape) < prob/2
    noisy[salt] = 255
    noisy[pepper] = 0
    return noisy

sp_noisy = add_salt_pepper(img)
```

```python
# Generate speckle noise
def add_speckle(img, var=0.1):
    noise = np.random.randn(*img.shape) * np.sqrt(var)
    noisy = img * (1 + noise)
    return np.clip(noisy, 0, 255).astype(np.uint8)
```

Fig 1: Example of the implementation of the various noises type

Summary of the noises type

Noise Type	Characteristics	Causes
Gaussian	Additive, normally distributed	Sensor noise, low-light conditions
Salt-and-Pepper	Random black/white pixels (impulse)	Dead pixels, transmission errors
Speckle	Multiplicative, granular	Ultrasound, SAR imaging

Recommendations

Noise Type	Optimal Filter	Reason
Gaussian	Gaussian	Matches noise distribution; smoothens without outlier sensitivity.
Salt-and-Pepper	Median	Removes outliers (black/white pixels) without averaging.
Speckle	Mean	Mild smoothing, but consider **Non-Local Means** for better results.