Aim: Image Processing with Salt-and-Pepper Noise, Convolution, and Metrics Evaluation

Introduction:

This is a report on implementing and analyzing many of the image processing techniques. Tasks included adding salt-and-pepper noise, calculation of quality metrics of images, and results after applying 3x3 convolution kernels on the images. The primary objective is to try to understand how different operations tend to affect the quality and visual appearance of an image and how to measure such effects quantitatively and visually.

Adding Salt-and-Pepper Noise: Visualization

An image is employed for this exercise. A Python code was written that introduces salt-and-pepper noise at various stages of degradation. Salt and pepper noise randomly sets some pixels to the color black (pepper) or white (salt), which mimics one of the typical degradations of an image.

Implementation

A function `add_salt_pepper_noise()` was defined that takes an image and probabilities for salt and pepper noise.

- The image was iterated pixel by pixel, and noise was added based on random values and the given probabilities.
- Different levels of noise, that is, 0.01, 0.05, 0.1, and 0.2 were applied to analyze the impact visually.

Analysis:

- At low noise levels, the image was almost unaffected, and as the noise level increased, the image quality degrades severely.
- Noise at 0.2 was the one that obscured the image content. The figure shows how higher probabilities of noise introduce random black and white pixels that distort the image.

Task 2: PSNR and MSE Analysis in Salt-and-Pepper Noise

This involved adding salt-and-pepper noise to an image and calculating PSNR and MSE for measuring the distortion.

Implementation :

Again applied salt-and-pepper noise with the help of same function used in Task 1
Calculated two metrics:

- **PSNR:** The more significant the PSNR, the more the noise level is small, and the image is crystal clear.
- **MSE:** The lesser number of difference between the source image and the noisy version, the smaller will be the value of MSE.

Variances of noise as 0.01, 0.05, 0.1, 0.2 were tested.

R.CHANDU BADRINATH MANIKANTA
AP24122060018
MTech Data Science

Analysis

- As the noise level increased, PSNR decreased and MSE increased, meaning the more distortion in the image.
- At 0.01 noise level, the PSNR was high, and minimal degradation occurred, but at 0.2 it lowered to a much lower PSNR and MSE increased, which points to the poor quality of the image.
- From the above explanation, it is ensured that the outcome of the same comes out to be identical with the visual results, and this quantitative result has established that PSNR and MSE are well-suited noise impact evaluation tools.

Task 3: Applying 3x3 Convolution Kernels

This exercise described convolution as an operation most image processing algorithms use to apply filters to an image using a kernel matrix. A program was written to apply predefined and user-defined 3x3 kernels on an image, which can be viewed in order to see the effect of different filters.

Implementation:

- The function `apply_convolution()` was called to apply a 3x3 kernel using OpenCV's `filter2D` function.
- Predefined kernels, such as Identity, Blur, and Sharpen, are already well demonstrated to simply illustrate the different effects of various filters.
- Identity Kernel preserved the original image.
- Blur Kernel "blurred the image using pixel averaging of neighboring values."
- Sharpen Kernel showed the brightening of the edges within an image with enhanced details.
- \- Users can also specify their own kernel, which enables different filtering effects.

Discussion:

The **Identity Kernel** - provided no change in the image and was used as a kind of control
The **Blur Kernel** - smoothed the image, reducing sharpness and details, making this effect useful for noise removal
The **Sharpen Kernel** - enhanced the edges, increasing the contrast, therefore, giving a much clearer and more defined appearance to the image.
Custom kernels allowed users to experiment with different transformations that further extended the definition of convolution beyond expectation.

Takeaways

1. Salt-and-Pepper Noise:

This noise models random pixel disturbances, and it helps understand how images degenerate as a result of transmission or acquisition.
Visual inspection showed the fact that even low-noise noises degrade an image quality while the larger noises produce extreme distortion.

2. PSNR and MSE:
PSNR and **MSE** as a metrics with solid ground for noise-induced image degradation.

- PSNR diminishes and MSE grows with the amount of noise incurred. It therefore provides an accurate numerical value to rate the quality of the image.
- They are essential to the topic of image processing in that it must be applied in the assessment of noise removal algorithms.

Convolution and Kernel Filters:

- Convolution allows us to apply a number of filters to the image where the resulting manipulation can be blurring, sharpening, or extracting edges, to name a few.
- Predefined kernels helped illustrate normal filters, whereas custom kernels allowed for experimenting with individualized effects.
- Understanding convolution is the first step to image processing because it is used from simple filtering into a complex level of image recognition.

4. Applications of Image Processing Techniques:
These techniques are widely applied in real-word applications: in the field of medical imaging, photography, computer vision, improvement of quality, noise reduction, and edge detection.

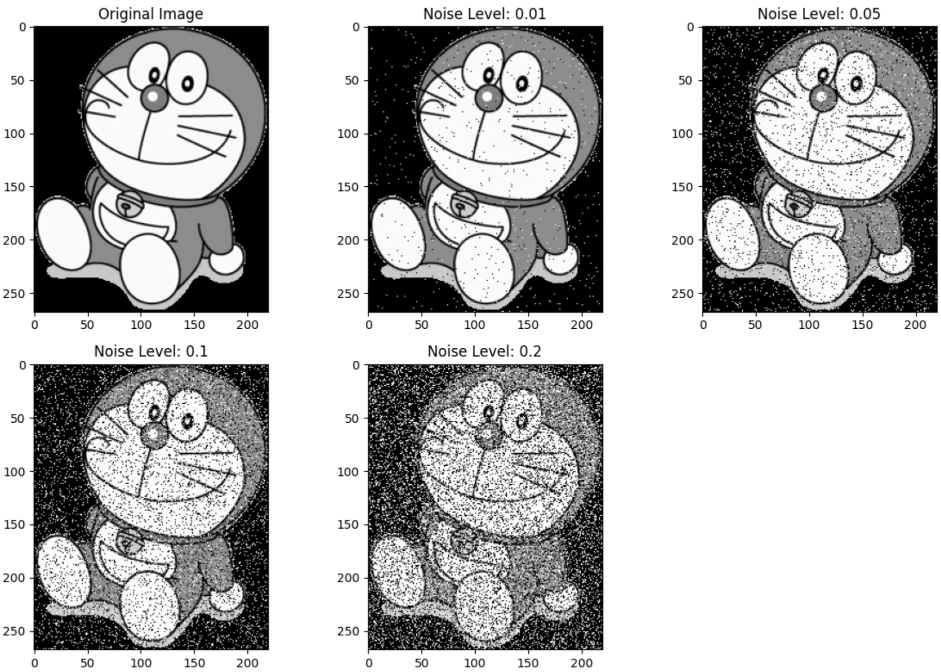
Conclusion:

This was a comprehensive report on the various image processing techniques covered in this report: noise addition, quality evaluation, and convolution. It illustrates such both visual and quantitative effects of these operations. Emphasis is therefore given on their use in enhancement and analysis of images. With application of these techniques, the user acquires a greater understanding of how images may be manipulated and evaluated in terms of quality, thus providing a basis for more sophisticated use in image processing and computer vision.

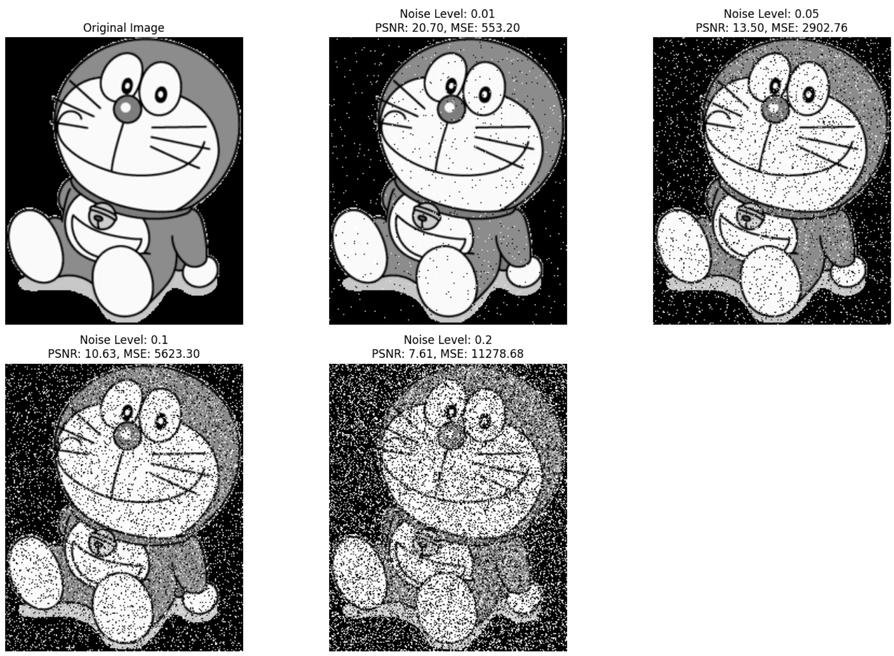
R.CHANDU BADRINATH MANIKANTA
AP24122060018
MTech Data Science

OUTPUT

Task1:



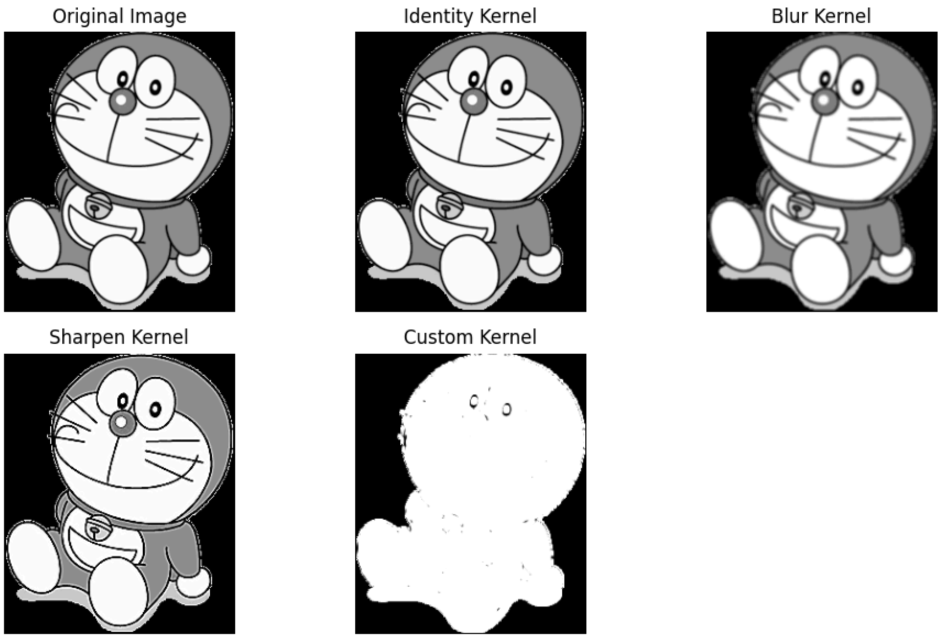
Task 2:




```
Noise Level      PSNR    MSE
-----------------------------------
0.01             20.70   553.20
0.05             13.50   2902.76
0.10             10.63   5623.30
0.20             7.61    11278.68
```


R.CHANDU BADRINATH MANIKANTA
AP24122060018
MTech Data Science

Task 3:



R.CHANDU BADRINATH MANIKANTA
AP24122060018
MTech Data Science