

Quantifying Noise and Enhancing Images

Aidan Sharpe

The goal of this lab is to understand how to work with noise in two-dimensional signals. In this case, our signals are greyscale images from the AT&T face database used in the Signals & Systems course project. This exercise may be completed in either MATLAB or Python.

Required Python Libraries

If you choose to use Python, the following libraries must be installed for the provided code to work properly.

- [NumPy](#)
- [Matplotlib](#)
- [Pillow](#)
- [SciPy](#)

Adding Noise

During this lab, we will experiment with two different types of noise, namely Gaussian noise and Salt & Pepper noise.

Load the first image from the AT&T face database `s1/1.pgm`. Show the clean image along with four increasing levels of Gaussian noise. Then, do the same for salt and pepper noise. Functions are provided for adding Gaussian and salt and pepper noise.

Quantifying Noise

We are able to change the amount of Gaussian noise by varying the standard deviation. The amount of salt and pepper noise is quantified by an “amount” parameter. We can quantify the amount of noise in general by using the signal to noise ratio. This is calculated as the power of the signal divided by the power of the noise.

1. Calculate the average SNR over the entire database for Gaussian noise with a standard deviation of 5.
2. Plot the average SNR for Gaussian noise with standard deviations ranging from 1 to 10.
3. Plot the average SNR for salt & pepper noise amounts ranging from 0.01 to 0.10.

Enrichment Question: It would be nice to have a generic noise amount that applies to both Gaussian noise and salt and pepper noise. Using the SNRs you calculated, what is the relationship between standard deviation for Gaussian noise and amount for salt and pepper noise?

With this relationship in mind, find the standard deviation for Gaussian noise and the amount of salt and pepper noise required to produce an average SNR of 20.

Removing Noise

Now we will attempt to remove the noise from the images. We will use two techniques: median filtering and Gaussian smoothing. These image processing algorithms are available in the `scipy.ndimage` module.

1. Over the entire database, produce images with an SNR of 20.
2. Apply the median filter to each image.
3. Compare the SNR before and after filtering.
4. Using ANOVA, determine if the median filter is significantly better for one type of noise.
5. Do the same for Gaussian smoothing.