Challenge 2

Student: Xabier Oyanguren Asua
NIU: 1456628 Date: 28/04/2022

1 Objectives

We wish to study the amount of noise that a single image registers in front of the meaningful signal of the image.

2 Data Acquisition

Two videos were acquired using the camera of a Xiaomi Redmi 5 Pro (with a resolution of 1280x720 pixels per frame in each video). The same scene was recorded with a static camera with bad lighting conditions and with proper lighting conditions.

A Python script was generated where using OpenCV's library for Python, cv2, 200 frames of each video were taken in numpy arrays.

3 Procedure

The first 100 frames of each array were used to generate a relatively low noise image (in good and bad light conditions). To do this, the 100 images were averaged, as a way to get rid of the independent Poissonian noise typical in photo-detection and imaging. In fact this works for any independent identically distributed (iid) random noise because the mean of a sum of N (in this case 100) iid random variables tends towards a zero variance with increasing N (meaning the mean random variable tends towards the mean -presumably the true intensity at that pixel- almost surely).

The 101-st frame of each video was taken as a sample single photo, we will call it the relatively small range photo (where photons were only captured for 1/25 seconds -the video was taken with 25 frames per second-).

The rest of 100 photos were added to generate the emulation of a relatively big range photo, where each pixel contains the information of photons arriving for as long as 4 seconds.

These images can be found in Figures 1 and 2.

Then, we can estimate the noise in the emulated low and wide range images relative to the "no-noise" references we built by averaging, in both lighting conditions with a simple absolute value pixel-wise subtraction. Computing the estimated Signal to Noise Ratio (SNR) is then done following the equation:

$$SNR = \frac{I_{max}}{2\sigma_\epsilon}$$

where I_{max} represents the maximum intensity value (which we will set to 255 for the image pixels are 8 bit unsigned integers), and σ_ϵ stands for the estimated standard deviation of the noise, which we estimate as the standard deviation of the absolute pixel-wise difference between the noisy and "non-noisy" reference.

We can also compute the peak signal to noise ratio (PSRN) in decibels (dB) typically employed in computer vision as:

$$PSNR = 10\ log_{10}\Big(\frac{I_{max}^2}{MSE}\Big)$$

where MSE stands for the mean square differences found between the reference and the noisy image.

It is clear that both quantities should increase for a less noisy image (as the standard deviation of errors decreases, the SNR tends towards infinity monotonously, and so does the PSNR as the mean square error, which is an unbiased estimator of the variance of the noise).

4 Results

The resulting estimated noise images can be found in Figure 3 for good light conditions and Figure 4 for bad lighting conditions. In all images, the color-map is a gray-scale with minimum 0 and maximum 255.

The resulting SNR and PSNR-s can be found in the headers of the figures, and in Table 1.

We effectively find that the low range image has a bigger noise (smaller SNR and PSNR) than the wide range image, which is clear for the wide range image, when normalized to be compared with the reference non-noisy image, we are effectively dividing it pixel-wise by the number of frames considered, leaving in each pixel as noise, the mean of 100 independent presumably Poissonian random variables. This makes the effective variance of the noise random variable decrease at about a factor 100 (the standard deviation by about 10). Consequently, the SNR should increase a factor 10 ideally. In our case, we find a factor 7.2 in the bad lighting case and a factor 4.65 in the good lighting case. Yet due to our other simplifications in the computation of the SNR, we consider this a successful check.

Table 1: Resulting SNR and PSNR measurements.

Image	**Light Condition**	**SNR**	**PSNR (dB)**
Small range	Good	58.919	38.678
Wide range	Good	269.121	51.213
Small range	Bad	159.689	47.565
Wide range	Bad	1143.586	64.508

5 Code

The employed code can be found fully commented in my Github repository:
https://github.com/Oiangu9/_Miscellaneous/blob/main/Challenge2_CV_Code.py

References

[1] Theory Lectures.

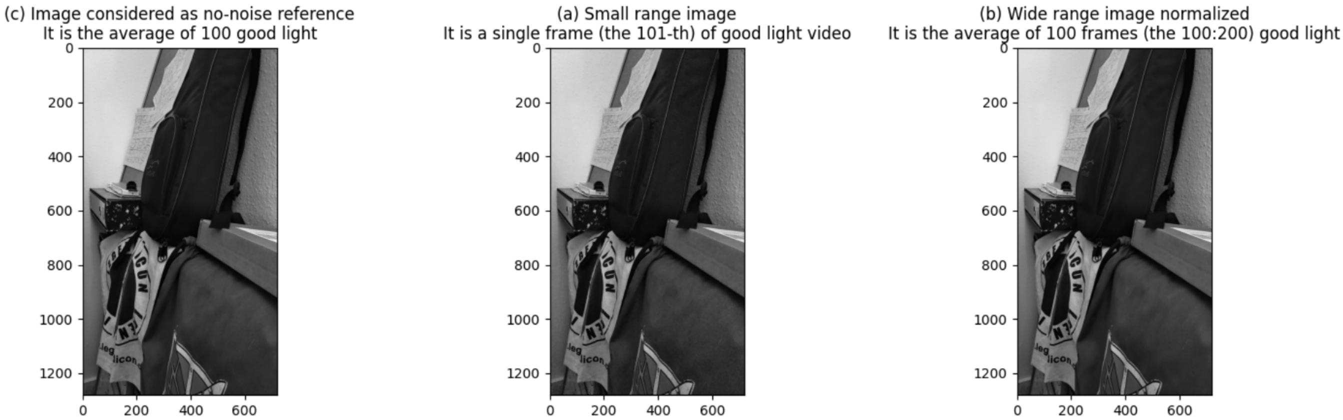

Figure 1: Good lighting condition employed images.

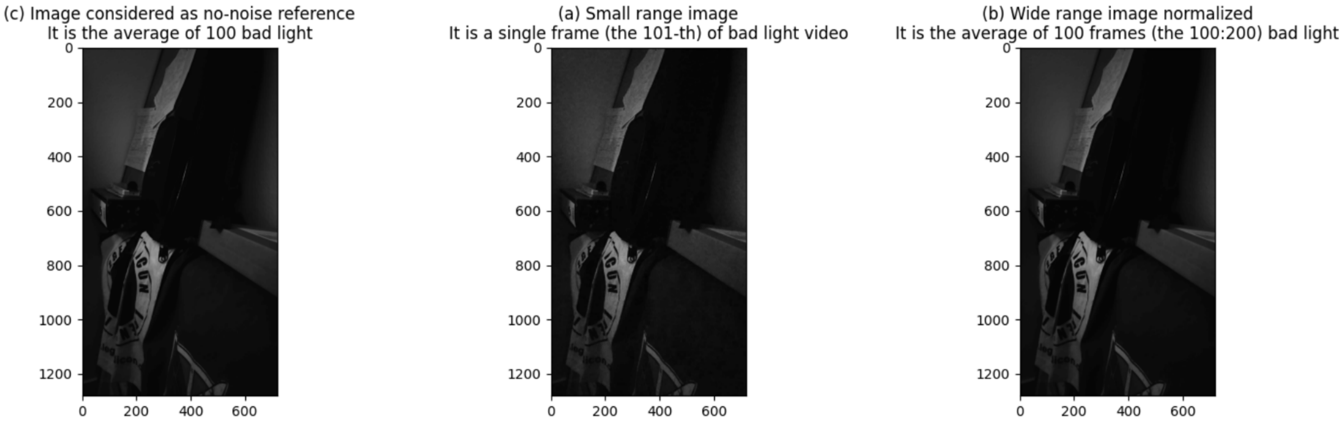

Figure 2: Bad lighting condition employed images.

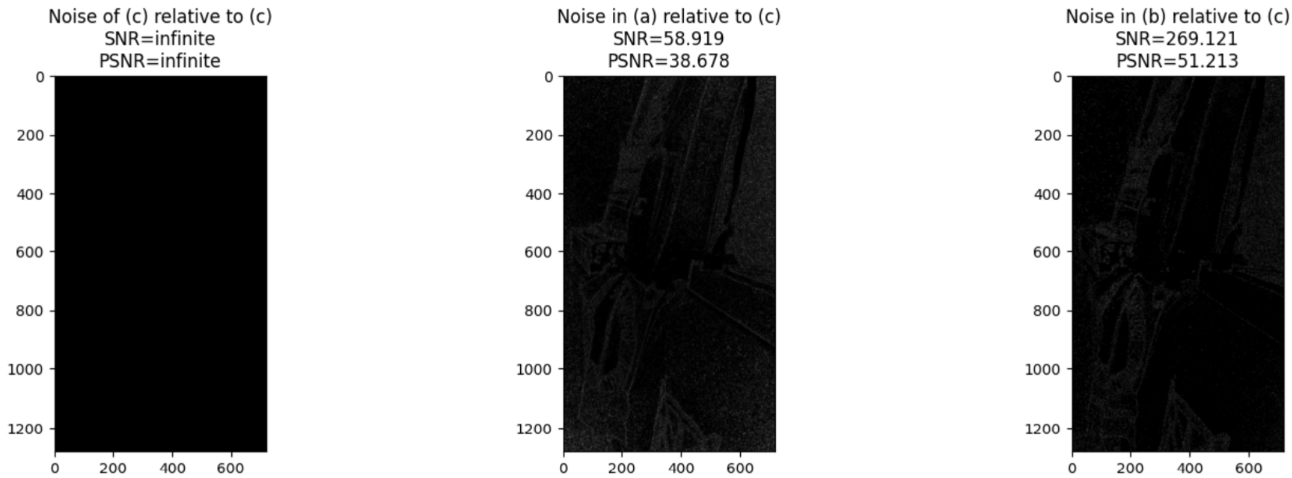

Figure 3: Noise images computed for images in good light conditions in Figure 1. Grayscale color-map ranges from 0 (black) to 255 (white).

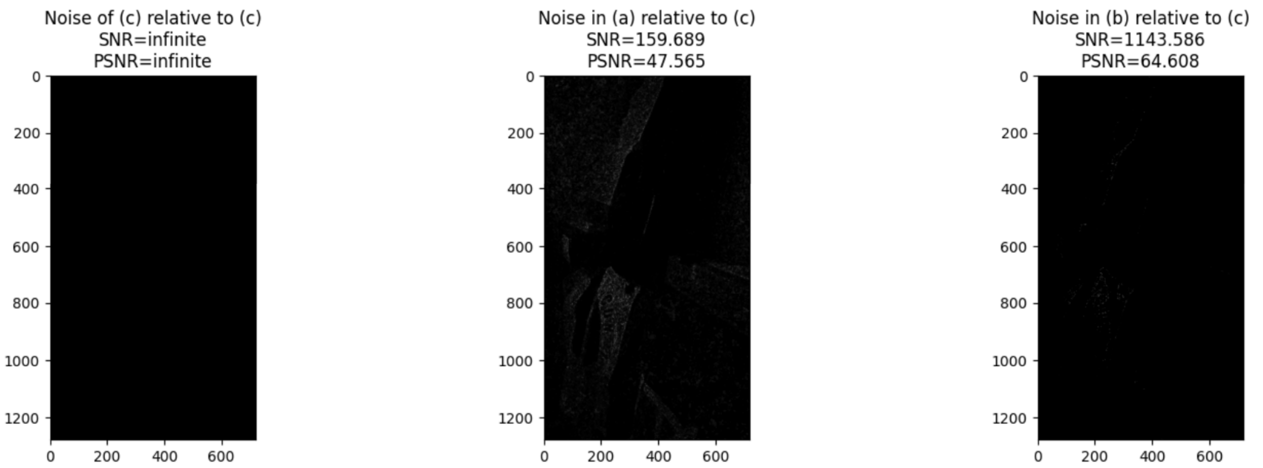

Figure 4: Noise images computed for images in bad light conditions in Figure 2. Grayscale color-map ranges from 0 (black) to 255 (white).