Explanation of the use of the function randn() and poisson() of the subpacket numpy.random

Students Liu Cong, Leonardo Ulloa

Use of the function poisson()

The function poisson of the sub-packet numpy.random only has one required argument: the lambda of the distribution. This lambda can by a matrix, allowing you to obtain a new matrix, which has in each position a sample generated using as lambda the value that was in that position in the input matrix.

This is exactly what we did to simulate the detector noise for the image, to generate random values that follow the poission distribution using the intensity of each position as lambda.

Use of the function randn()

The function randn() of the sub-packet numpy.random generate samples following the Standard Normal Distribution, which is the normal distribution with mean 0 and 1 as standard deviation. This function allow you to specify the shape of the output.

For this simulation we want the variables to distribute with mean equals to the intensity (N) of each point and standard deviation equals to the root of this intensity ($\sqrt{N}$). Therefore, in order to use randn() to simulate the noise we need to change the parameters of the distribution, and for this aim we will use the following property: Let $X \sim N(\lambda_0, \sigma_0^2)$, let $Y = aX + b$, then $Y \sim N(\lambda_1, \sigma_1^2)$ where :

$$\lambda_1 = a\lambda_0 + b \tag{1}$$

$$\sigma_1^2 = a^2\sigma_0^2 \tag{2}$$

Then if we work in the previous expression to get the values of a and b that we need:

$$a = \frac{\sigma_1}{\sigma_0} \iff \sigma_1 = a\sigma_0 \Rightarrow \sigma_1^2 = a^2\sigma_0^2 \tag{3}$$

$$b = \lambda_1 - a\lambda_0 \tag{4}$$

Substituting the values of the mean and the variance of the Standard Normal Distribution we obtain:

$$a = \sigma_1 \quad b = \lambda_1 \tag{5}$$

Substituting the mean and the standard deviation we want:

$$a = \sqrt{N} \quad b = N \tag{6}$$

Where N is the intensity. Then the transformation that has to be made to the values generated by randn() is :

$$Y = \sqrt{N}X + N$$

Hence a way to do what we want is to generate random variable following the Standard Normal Distribution using as shape of the output the same of the image, then we can use the sqrt and multiplication ufunc to multiply each generated sample by the square root of the intensity in each position, and finally use the addition ufunc to sum the intensity in each position.

October 2024