DAIICT

Assignment 1

Advanced Image Processing

Kishan Vaishnani - 202011004
Dhyanil Mehta - 202011032
Date: 29-1-2021

Table of Contents

Task 1

Problem Statement

Up-sample and down-sample the image of Figure 1 by scale factor 4. Discuss the effect of changing sampling rate. Also observe the effect of different quantization levels (L = 2, 4, 8, 16, 32, 64, 128, 256) for this image. (Figure 1 image: fig1.jpg)

Library Used: PIL

Method

1. Up & Down Sampling
   PIL.Image.Image.resize((Width * Scaling factor, Height * Scaling factor))

   We have used the PIL library's resize function for sampling the image, which by default uses Bicubic Interpolation.

   For up-sampling, scaling factor = 4 and,

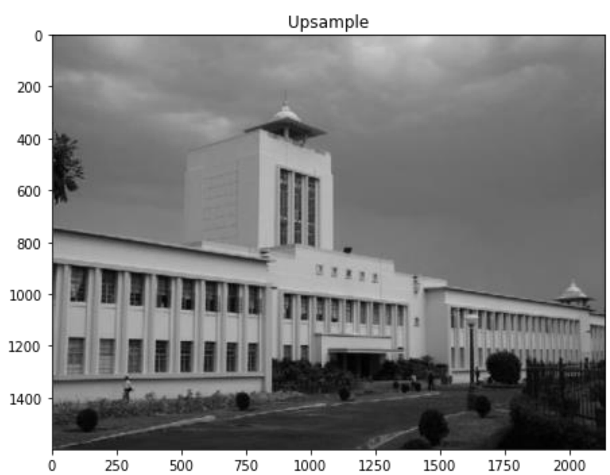
   For Down-sampling, scaling factor = ¼

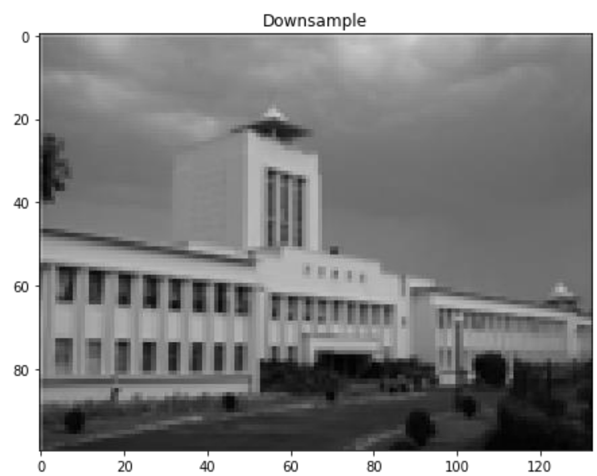
2. Quantization level
   PIL.Image.Image.quantize(L)

   Return new image with applied quantization level L where L = 2^k, k = Bits

Observation

1. Up-Sample & Down-Sample
   Up-sample size: (2136, 1600), Down-sample size: (133, 100)





2. Effect of different quantization levels

Observation: By increasing level we have observed image appears clearer than last level.

L=2

L=4

L=8

L=16

L=32	L=64
 L=128	 L=256

Conclusion

- Up-sampling the image increases its size by the sampling factor, and the image becomes zoomed, but the quality is reduced.
- Down-sampling decreases its size by the sampling factor and the image becomes more blurred, and quality is significantly reduced.

Task 2

Problem Statement

Compute the mean and variance of Figure 2. Compute the same parameters for Figure 1. Then add additive white Gaussian noise of mean 0 and standard deviation 20 to both the images and compute the means and variances. Explain your observations. Further, generate several such noisy images from Figure 1 and average all of them. Does the averaged image have a less noisy artifact? (Figure 2 image: baboon.jpg)

Library Used: NumPy

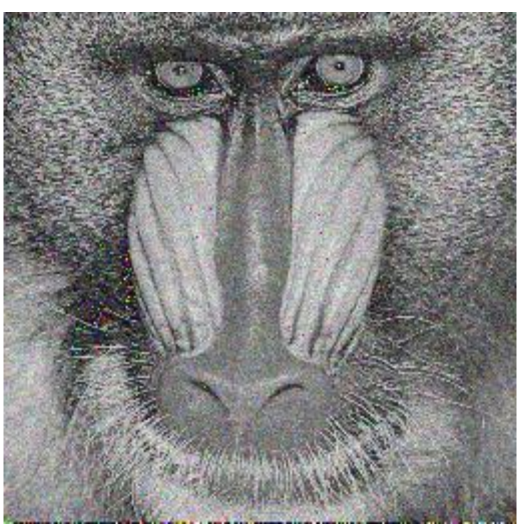
Method

1. Computing Mean
   numpy.mean(image_array)
2. Computing Variance
   numpy.var(image_array)

Here PIL Image object is converted to a NumPy array, and then the mean and variance are computed.

Observation

1. Effect of white gaussian noise of mean 0 and standard deviation 20
   Observation: By adding noise to an image, it appears with white and random colored spots and gets little bit unclear. Also, their mean and variance gets increased.



Without noise: Mean: 128.96, Variance: 1820.05 **With noise: Mean: 129.12, Variance: 2206.51**	**Without noise: Mean: 112.94, Variance: 2133.13** **With noise: Mean: 116.15, Variance: 2512.29**

2. Some noisy images of fig1 and average image of all of them
   Observation: Averaging more noisy images gives an image with even less noisy artifacts.

Average of noisy images

Conclusion

- As the quantization level increases, the image can use more values to display the image, and thus more and more fine details can be seen.
- The mean and variance of the image increases as the image becomes noisier as the noise are merely random values.
- The image quality is improved by averaging its noisy images together.

Task 3

Problem Statement

Find out the number of objects from Figure 3. Label distinct objects with distinct colors. [Use the algorithm of finding out connected components.] (Figure 3 image: cc.jpg)

Library used: OpenCV

Method

1. Converting a grayscale image into Binary Image

   cv2.threshold(a,b,c,d)

   a. Greyscale image to be convert

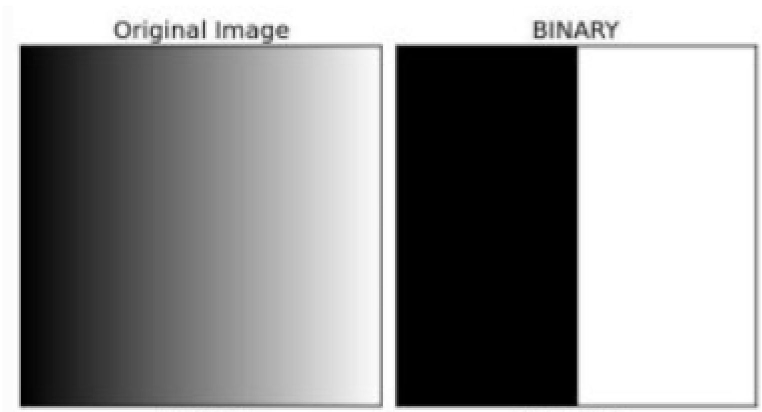
   b. A threshold value is used to classify the pixel values.

   c. maxValue will assign to a pixel if the pixel value is more/less than the threshold value.

   d. Different types of thresholding function.

   cv2.THRESH_BINARY

   This thresholding function will specify the threshold to be converted into pixel values.



Source: https://opencv-python-tutroals.readthedocs.io/

2. Finding Connected Components

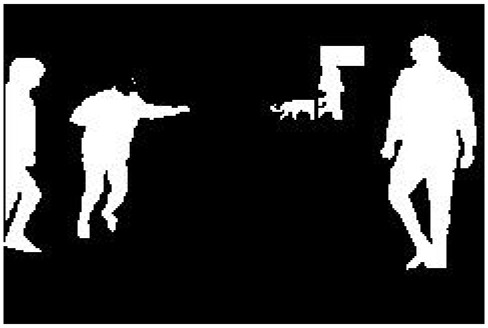
   cv2.connectedComponentsWithStats(BinaryImage, Connectivity)

   This method will return following Parameters.

   a. Total number of labels it had generated

   b. A labeled set similar size to Binary Image

Observation

	
Binary Image	Labeled Binary Image

Conclusion

- By converting the image to a <u>binary image</u> (i.e., L = 2), we can easily find connected components in the image.