EE6310 Image and Video Processing, Spring 2024

Indian Institute of Technology Hyderabad
Homework 4, Assigned 07.03.2024, Due **11:59 pm on 17.03.2024**

Persistent questioning and healthy inquisitiveness are the first requisite for acquiring learning of any kind.
–Mahatma Gandhi

Instructions:

- For Q1, use images from University of Southern California's image database at `http://sipi.usc.edu/database/database.php?volume=misc`.
- Please use the *lighthouse.png* and the template (*template.png*) uploaded along with this assignment for Q2 and Q3.
- Please turn in Python Notebooks with the following file name convention: `your-roll-number`-hw4.ipynb.
- Do not turn in images. Please use the same names for images in your code as in the original.

1 Image Quality Assessment

In this problem you will compare the two full reference image quality assessment techniques discussed in class – PSNR (MSE) and the SSIM index. This problem is aimed at understanding the shortcomings of PSNR (MSE) in measuring image quality.

1. Write a Python function that accepts a pair of images of identical dimensions and computes the MSE and PSNR between them. (2)
2. Write a Python function that accepts a pair of images of identical dimensions and computes the SSIM index between them. The function should return the mean SSIM (MSSIM) index as well as the SSIM map. Use the definition of the SSIM index from the slides and use a Gaussian window for the weighted mean, variance and covariance computations. (8)

Write a program to generate the following distortions:

1. Mean shift by 16. (1)
2. Additive white gaussian noise (AWGN) with standard deviation $\sigma = 16$. (1)
3. JPEG compression at quality level 10. Use any open source Python library for this. (1)
4. Gaussian blur using a 5×5 window with $\sigma = 2$. (1)

Work with 3 images of your choice from the set mentioned in the instructions. Subject the images to the distortions mentioned above. Compute the MSE, PSNR and SSIM index for each distortion. Also, rate the images yourself (subjectively) using a continuous scale between 0 and 1. Compute the correlation between your scores and PSNR, your scores and SSIM. What can you conclude about the two metrics? (1)

2 Edge Detection

In this problem, you will write a program to compute the edge map of an image. See instructions for image.

2.1 Gradient Edge Detectors (10)

1. Use the following gradient operators:
   (a) Centered 2-D differencing.
   (b) Roberts operator.
   (c) Prewitt operator.
   (d) Sobel operator.
2. Estimate gradient magnitude using the following definitions:
   (a) $M(i,j) = \sqrt{\Delta_x^2(i,j) + \Delta_y^2(i,j)}$
   (b) $M(i,j) = |\Delta_x(i,j)| + |\Delta_y(i,j)|$
   (c) $M(i,j) = \max\{|\Delta_x(i,j)|, |\Delta_y(i,j)|\}$
3. Threshold the magnitude map using an empirical threshold τ to find the edge map E.

2.2 Laplacian Edge Detectors (5)

1. Compute the Laplacian using the convolution template

$$\begin{bmatrix} 0 & +1 & 0 \\ +1 & -4 & +1 \\ 0 & +1 & 0 \end{bmatrix}$$

2. Compute the edge map E as the output of a zero crossing detector.

2.3 Laplacian of Gaussian (LoG) (5)

LoG was motivated by the sensitivity of gradient and Laplacian edge detectors noise. In the problem, you will implement an edge detector using the LoG operator. Work with the *lighthouse.png* image that is corrupted with AWGN whose $\sigma_n = 10$. Experiment with different values of σ for the Gaussian pre-filter and compute the edge map E after zero crossing detection. Compare your result with gradient based techniques implemented above and verify the robustness of LoG.

3 Template Matching (10)

In this problem, you will write a program to implement template matching using the normalized cross-correlation metric defined in class. Use the definition of $\hat{C}_{\mathbf{B_T} \diamond \mathbf{I}(i,j), \mathbf{T}}$ from slides. See instructions for image and template.

1. Find the normalized cross correlation image.
2. What is a good threshold to detect a match?
3. Now add Gaussian noise to the template with $\sigma = 1, 3, 5, 10$. Repeat steps (1) and (2) at each noise level. How does the threshold change as noise increases?
4. Rotate the image at the following angles $\theta = 5^o, 10^o, 15^o$. Repeat steps (1) and (2) at each angle. How does the threshold change as rotation increases? Implement rotation using the rotation matrix

$$\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

   and a suitable interpolation technique.