APPLIED MATHEMATICS : HONS AM 793 — Image processing

# ASSIGNMENT 4: RESTORATION (2022)

For this assignment you must write a short report in any word processor of your choice (MSWord, Latex, ....) where you explain your methods and show your results. All input and output images must be shown (preferably so that they can be compared on the same page). PLEASE print you images fairly large. The SunLearn grading system has no magnification option, and small images make it difficult to assess your work.

Add the code in an Appendix.

1 Take any grey-scale image of your own choice, and blur it with a gaussian mask $M$ of the form

$$M_{jk} = \exp\left(-\frac{(j-j_0)^2 + (k-k_0)^2}{s}\right)$$

with $s$ a chosen constant, and $(j_0, k_0)$ are the indices of the center pixel of the matrix. Scale $M$ so that its total sum of pixels is 1. (If it does not sum to 1, it will also brighten or darken the image, apart from blurring it.)

You may either do the blurring physically with periodic convolution, or in frequency space by multiplying its Fourier Transform with the Fourier Transform of the mask and performing the Inverse Fourier Transform afterwards. (The results ought to be equivalent.)

Then *deconvolve* the blurred image, using a *cut-off filter* in an attempt to reconstruct the image (i.e. *unblur* it). Let $A$ be the blurred image and let $M$ be the mask (both the same size), and let $B$ be the reconstructed image. The over-hat denotes the Fourier Transform. The cut-off filter is

$$\hat{H} = \min(1/\hat{M}, C),$$

where $C$ is a chosen constant. Deconvolution is performed by

$$\hat{B} = \hat{A} \times \hat{H}.$$

Here both / and $\times$ are *element-by-element* division or multiplication.

The reconstructed image $B$ is given by

$$B = \mathrm{IFT}(\hat{B}).$$

Play around with various choices of $C$, and show at least two examples in your report.

Recommendations: Do not take an image that is too large. An image of the approximate size 256×256 or smaller will be fine. If your image is $m \times n$, make sure that $m$ and $n$ are both even (discard a row or a column where necessary), and let $j_0 = m/2 + 1$ and $k_0 = n/2 + 1$. The gaussian spread constant, $s$, must not be too large — the larger you make it, the worse the blur, and the more difficult it is to unblur the image successfully. Therefore apply reasonable blur, but don't overdo it. The cut-off constant $C$ should be larger (about 10× or so) than the minimum value at the center of $1/|\hat{M}|$.

2 The image `avosblurred.tif` (call it $B$) was captured with an out-of-focus lens system (i.e as if convolved with a mask $M$). The image `avosblurnoise.tif` (call it $C$), is $B$ with noise added to it. Both images are colour images, but convert them to grey-scale images for the purpose of doing 2(a) and 2(b) below.

The point-spread function $M$ of the lens system (i.e. the mask) is given in the EXCEL file `convmask.xlsx`. (For those who use MATLAB, `M=readmatrix('convmask.xlsx');` will read in an EXCEL file.)

Construct a Wiener Filter $W$ with the given point-spread function. The formula is as follows:

If $H$ is the Fourier transform of $M$, then

$$W = \frac{1}{H}\left(\frac{H^2}{H^2 + K}\right)$$

Here $K$ is a constant that represents the noise-to-signal ratio. Application of the Wiener filter is as follows: The reconstructed image is $\mathrm{IFT}(\hat{C} \times W)$.

2(a) Apply the Wiener filter to the grey-scale version of `avosblurred.tif` to deblur it. Illustrate the result with various values of $K$.

2(b) Apply the Wiener filter to the grey scale version of `avosblurnoise.tif` to deblur it. Illustrate the result with various values of $K$, and try to find an optimal $K$ (Use your own judgement.) Then sample a uniform patch of the image and calculate the inverse signal-to-noise ratio. This means: take the standard deviation of the noise in the sampled patch and divide by the mean of the whole image. (This is only a course approximation of the inverse signal-to-noise ratio.) Compare this to your optimal $K$.

2(c) (Optional, will not earn marks.) Repeat 2(b) for the colour image `avosblurnoise.tif`. You need to Fourier Transform each of the colour planes, filter them each separately with the same Wiener filter, and recombine.