

Assignment 2 CS9630a

Out: November 24th, 2015

In: December 15th, 2015

QUESTION (1)(25%): This question concerns **Image Restoration**. Compute and display the Y image of the YIQ images of lena.jpg (use **ntsc2rgb** to compute this). Title this all and all other images appropriately using fontsize 16. On the course webpage, there are 3 images, called **mystery1.jpg**, **mystery2.jpg** and **mystery3.jpg**, which are Gaussian blurred images of the Y image (with unknown σ values). Use MatLab functions **weiner2** and **deconvwnr** to perform Weiner filtering. on the 3 mystery images:

1. For each of 8 filter sizes, $[n \ n]$, where $n=3, 5, 7, 9, 11, 13, 15$ and 17 perform weiner filtering on the 3 mystery images. Display appropriately titles 24 images in total (4 images per page). For each image, compute the average of the norm of the difference image between the Y image and the restored image. Present these results in tabular form such as:

n	image1	image2	image3
i	$result_{i,1}$	$result_{i,2}$	$result_{i,3}$

where $result_{i,1}$ is the average of the norm of the difference between the Y image and the weiner restored image for mystery image 1 using $[i \ i]$. Make sure that the Y image and the weiner filtered image are of the same type and appropriately scaled before you do this calculation. Do your quantitative results agree with your qualitative judgement about which restored image is the best and worst?

2. For each of 8 NSR values from 10^{-6} , 10^{-5} , 10^{-4} , 10^{-3} , 10^{-2} , 10^{-1} , 10^0 and 10^1 each of the 6 sigma values 1.5, 3.0, 4.5, 6.0, 7.5, 9.0 and generate

a Gaussian point spread function PSF using **fspecial** where the size of the Gaussian used is determined by the $6\sigma + 1$ rule (the next biggest odd integer to this number). Use **deconvwnr** to do weiner deconvolution for each of the 3 images for all the NSR and sigma values. You should get $3*8*6=144$ images, which should be displayed 4 images per page in your latex document. Again, compute quantitative results by computing the average of the norm of the difference between the Y image and the deconvwnr images. Such a table might have the structure:

NSR	sigma	image1	image2	image3
NSR_value	sigma_value	$diff_{NSR,sigma,1}$	$diff_{NSR,sigma,2}$	$diff_{NSR,sigma,3}$

Obviously, you should use a loop to do these calculations and you should generate filenames and titles using character concatenation using the appropriate parameter names and values. Answer the following questions:

1. For each of the 3 images, which is the best and worst qualitative images, for both the **weiner2** and the **deconvwnr** filters? What qualitative conclusions can you draw?
2. Do your quantitative results as computed above agree with y of the Y image and each denoised image. Why or why not?

ANSWER (1):

QUESTION (2) (25%): This question concerns **Histogram Equalization** and **Histogram Modification**. You are required to perform histogram equalization or histogram modification in number of different ways on the lena.jpg image.

Perform the 3 histogramming tasks:

1. Compute histogram equalization on the red, green and blue colour planes. Display the resulting colour image, appropriately titled. Compute the histograms, h_1 , h_2 and h_3 of the red, green and blue colour planes using **hist**. Plot these histograms, appropriately titled.
2. Perform histogram modification on the colour lena.jpg image by forward permutating the colour histograms. Use histogram h_2 on the red image, histogram h_3 on the green image and histogram h_1 on the blue image. Display this image, appropriately titled.
3. Perform histogram modification on the colour lena.jpg image by backwards permutating the colour histograms. Use histogram h_3 on the red image, histogram h_1 on the green image and histogram h_2 on the blue image. Display this image, appropriately titled.

What conclusions can you make about these images compared to the original lena.jpg image. A total of 4 images are required (including the original lena.jpg image). Perform the 4 histogramming tasks:

1. Compute $h_4 = (0.333 * h_1 + 0.333 * h_2 + 0.333 * h_3)$ and histogram modify each original colour plane using h_4 . h_4 simply averages each colour.
2. Compute $h_5 = 0.299 * h_1 + 0.587 * h_2 + 0.114 * h_3$ and histogram modify each original colour plane using h_5 . h_5 is a weighted average of the 3 histograms, where the weights are the YIQ averages for the intensity image (Y image).

3. Compute $h6 = 0.1 * h1 + 0.9 * h2 + 0.1 * h3$ and histogram modify each original colour plane using $h6$. $h6$ is a weighted average of the 3 histograms, where the weights favour the green histogram over the red and blue histograms. Display this image, appropriately titled.
4. Compute $h7 = 0.49 * h1 + 0.02 * h2 + 0.49 * h3$ and histogram modify each original colour plane using $h7$. $h7$ is a weighted average of the 3 histograms, where the weights favour the red and blue histograms (equally) over the green histogram.

Display each image appropriately titled. Plot the histograms $h4$, $h5$, $h6$ and $h7$. Given these histogram and the 4 results, explain your results. Is this what you expected? For each pair of histogram modified colour images compute the average of the norm of the difference between the original colour image and the histogram modified images. You should get a 4×4 table of results such as:

	EQ4	EQ5	EQ6	EQ7
EQ4	$diff_{44}$	$diff_{45}$	$diff_{46}$	$diff_{47}$
EQ5	$diff_{54}$	$diff_{55}$	$diff_{56}$	$diff_{57}$
EQ6	$diff_{64}$	$diff_{65}$	$diff_{66}$	$diff_{67}$
EQ7	$diff_{74}$	$diff_{75}$	$diff_{76}$	$diff_{77}$

where $diff_{ij}$ is the average norm of the differences images for colour histogram modified images i and j . Naturally $diff_{44}$, $diff_{55}$, $diff_{66}$ and $diff_{77}$ are all zero. As well, note that $diff_{ij}$ and $diff_{ji}$ should be identical. Do these quantitative results agree with your qualitative judgements? Discuss your results.

ANSWER (2):

QUESTION (3) (50%): This question concerns Edge Detection using 1st order and 2nd order derivatives computed from the Fourier transform of the of the Y image of the YIQ colour image of the lena.jpg image (made using MatLab function **ntsc2rgb**). Perform the following tasks:

1. Compute the centered Fourier Transform (FT) of the Y image, $F(u, v)$, and display the log spectrum of this image, appropriately titled.
2. Compute frequencies u and v , Compute matrices U and V using **meshgrid**. Then compute the $Fx(u, v)$ and $Fy(u, v)$ (the Fourier transforms of the 1st order derivatives) as $iu2\pi F(u, v)$ and $iv2\pi F(u, v)$ respectively and compute $Fxx(u, v)$ and $Fyy(u, v)$ (the Fourier transforms of the 2nd order derivatives) as $-u^2 4\pi^2 F(u, v)$ and $-v^2 4\pi^2 F(u, v)$ respectively (or equivalently as $iu2\pi Fx(u, v)$ and $iv2\pi Fy(u, v)$ respectively). Finally, compute the Laplacian as $DelF(u, v) = Fxx(u, v) + Fyy(u, v)$. Display the log spectrum images of Fx , Fy , Fxx , Fyy and $DelF$, appropriately titled. Do these Fourier transform results make sense. Discuss the results.
3. Compute the inverse Fourier transform, FT^{-1} , of $Fx(u, v)$, $Fy(u, v)$, $Fxx(u, v)$ and $Fyy(u, v)$ to get $fx(x, y)$, $fy(x, y)$, $fx(x, y)$ and $fy(x, y)$. Compute the Laplacian in the spatial domain in one of 2 ways: as $del f(x, y) = fxx(x, y) + fyy(x, y)$ and as $nabla_f(x, y) = real(ift2(DelF(u, v)))$. In theory $del f$ and $nabla_f$ should be the same. Display all 6 derivative images, fx , fy , fx , fy , $del f$ and $nabla_f$, appropriately titled.
4. Compute the following statistics for $del f$: the minimum, maximum, mean, standard deviation, and median values. Compute 2 thresholds, $thresha$ and $threshb$, where $thresha$ is the median value of $del f$ while $threshb$ is 0.5 of $thresha$. Compute 3 edge maps (black for edgels, white for the background pixels) as:

- (a) Compute an edge map by looking for zero crossings. No thresholding is used. Compute the number of edgels using variable *num_edgels0*.
- (b) Compute an edge map by looking for zero crossings with thresholding *thresha* on the magnitude of *delf*. Compute the number of edgels using variable *num_edgelsa*.
- (c) Compute an edge map by looking for zero crossings with thresholding *threshb* on the magnitude of *delf*. Compute the number of edgels using variable *num_edgelsb*.

Display 3 appropriately titled edge map images. The title should contain the number of edgels and the threshold values used.

5. Lastly, we compute the edge maps directly from the 1st order derivatives, *fx* and *fy*. Compute $grad1 = \sqrt{fx^2 + fy^2}$ and $grad2 = abs(fx) + abs(fy)$. Print the minimum, maximum, mean, standard deviation and median values for *grad1* and *grad2*. Compute thresholds, *thresh1* and *thresh2* as the median values of *grad1* and *grad2* respectively. Compute edge maps by determining if a pixel is an edgel or not using $abs(grad1(a, b)) > thresh1$ or $abs(grad2(a, b)) > thresh1$. Again, edgels are coloured black while background pixels are coloured white. Use variables *num_edges1* and *num_edges2* to count the number of edgels found. Display these 2 edge maps as images with appropriate titles. Make sure each title includes the method used to compute the edges (the equations), the values of the thresholds and the edgel counts.

Give a discussion of your results. Which edges do you think are the best and why? Are 2nd order edges better than 1st order edges or vice versa?

ANSWER (3):