

assigned: 5 October 2017

due: Friday 20 October 2017

reading assignment: Same as HW #3

Filtering using Frequency-Domain Analysis

For the procedures below, you can use the following MATLAB functions:

- $F = \text{fft2}(f, M, N)$: Compute the $M \times N$ DFT (FFT) of $M \times N$ image $f(x, y)$ to give $M \times N$ DFT $F(u, v)$.
- $f = \text{ifft2}(F, M, N)$: Compute $M \times N$ DFT $^{-1}$ (IFFT) of $M \times N$ DFT $F(u, v)$ to give $M \times N$ image $f(x, y)$.

When suitable P and Q are used, both `fft2` and `ifft2` are useful for zero-padded quantities.

- $H = \text{lpfilter}(\text{TYPE}, M, N, \text{sig})$ (from Gonzalez, Woods, and Eddins's MATLAB-based book): defines an $M \times N$ lowpass filter $H(u, v)$ of the given TYPE ('ideal' or 'gaussian') and cutoff frequency sig . The MATLAB .m file for this function is given on CANVAS along with the supplemental .m file for function `dftuv(M, N)`. Finally, file "mainproj3.m" demonstrates many of the ideas used below.

1. *Basic DFT-based Frequency Analysis*: Let f be the $N \times N$ "checker" image.
 - (a) Give figures for $|F(u, v)|$ and $\log(1 + |F(u, v)|)$. Next, modulate $f(x, y)$ by $(-1)^{x+y}$ to give an image $f_m(x, y)$, and give figures for $|F_m(u, v)|$ and $\log(1 + |F_m(u, v)|)$. Finally, set $F(0, 0)$ to zero and compute the inverse DFT to give a new image $g(x, y)$. Give observations on all images. In particular:
 - i) What do the log and modulation operations do?
 - ii) What is the observable and analytical difference between g and the original f ?
 - (b) Using only $N \times N$ operations (i.e., you do NOT zero pad anything!), apply a Gaussian lowpass filter $H(u, v)$ with $\text{sig} = 15$ to f and denote $g(x, y)$ as the lowpass-filtered output image.
 - i) Use the MATLAB function `mesh` in an appropriate way to make a 3-D plot of $|H(u, v)|$.
 - ii) Give figures for $|G(u, v)|$ and $g(x, y)$.
To display the Fourier-transform magnitude figures above, be sure to use appropriate modulation (i.e., multiply $g(x, y)$ by $(-1)^{x+y}$) and scaling (i.e., plot $\log(1 + |G(u, v)|)$ for $|G(u, v)|$).
 - (c) Repeat part (b), but now use appropriate zero padding and compute $P \times Q$ quantities.
 - (d) For the results of parts (b-c), what impact does a filter have on the output image? Do you observe wraparound error? Discuss the nature of the zero-padded results.
2. *Filtering a Corrupted Image*: Let $f(x, y)$ be the "lake" image and create the corrupted image $c(x, y)$:

$$c(x, y) = f(x, y) + 32 \cdot \cos\left(\frac{2\pi 32x}{N}\right)$$

Clearly, $c(x, y)$ will have pixels (x, y) having values outside the 8-bit $[0, 255]$ range. So, you must perform all processing with sufficient precision!

- (a) Design a suitable notch filter $H(u, v)$ that when applied to $c(x, y)$ gives an image $g(x, y)$ that nearly resembles the original image $f(x, y)$. A notch filter rejects (i.e., sets to 0) a few specific frequencies while passing all others. You must describe how you designed your filter H by giving analysis to back up your design. *Lecture notes L15-21 \rightarrow L15-24 are very helpful here!*
- (b) Give suitable pictures of:
 - i. $f(x, y)$, $|F(u, v)|$, $c(x, y)$, $|C(u, v)|$
 - ii. $|H(u, v)|$, $g(x, y)$, $|G(u, v)|$
 - iii. The image and Fourier-transform magnitude of the difference image $(f(x, y) - g(x, y))$.
You, of course, will need to do appropriate modulation and scaling for plotting the DFT magnitudes.
- (c) Is it possible to completely recover f from c ? Why or why not?