JHU 553.493/693, Spring 2020 – Homework #5 (and last!)

Applications of the Discrete Fourier Transform in 1D and 2D

Due on Blackboard by the end of the day on Friday, May 1

Instructions: Write, on top of the first page of your assignment: Name (LAST, First), the HW# (in this case, "Homework #5"), and acknowledge others with whom you may have worked (just write "*Worked with ...*"). For the computational problems where you are asked to write computer code you may choose the programming language that you prefer, such as *Python* or *Matlab*. You do *not* have to type up your homework. If you handwrite it, please scan it, making sure it is readable. You should submit *one* PDF file on Blackboard (or at most two, if you choose to submit your code separately).

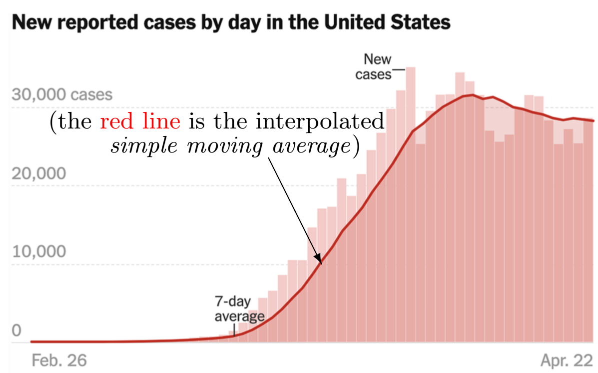
Problem 5.1 (moving average)**.** (The graph, which inspired this problem, appeared on the NYT.)

The *simple moving average* (SMA) is the unweighted mean of the previous N data points. That is, if we have a signal $s(t)$, $t \in I = \mathbb{Z} = Z(1)$, its N-point SMA is:

$$m(t) = \frac{s(t) + s(t-1) + \ldots + s(t-(N-1))}{N}, \quad t \in I.$$

(a) Find a convolution kernel $h(t)$, $t \in I = \mathbb{Z}$ such that we can write $m(t) = (h * s)(t)$, $t \in I$. This means we can interpret the SMA as a filter on $\mathbb{Z}$.

(b) The dual of $I = Z(1)/\mathbb{O}$ is $\widehat{I} = \mathbb{O}^\star/Z(1)^\star = \mathbb{R}/Z(1)$, so the frequency response $\mathcal{F}[h](f) = H(f)$, $f \in \widehat{I}$ of the filter is continuous and periodic. Show that:



$$H(f) = \frac{1}{N}\exp\big(-i\pi(N-1)f\big)\,\frac{\sin(\pi N f)}{\sin(\pi f)}, \quad f \in \widehat{I} = \mathbb{R}/\mathbb{Z}.$$ In particular, what is the value of $H(0)$?

(c) For $N = 8$, Plot the graph of the magnitude of the frequency response, i.e. of the real-valued function $|S(f)|$, just for the interval $f \in [0, 1]$. What type of filter is SMA (low-pass, high-pass, band-pass)? Justify your answer. *Note:* for a *periodic* Fourier transform, where are the low frequencies located?

Problem 5.2 (DFT and IDFT)**.** In class we defined the one-dimensional Discrete Fourier Transform (DFT) of a discrete- and finite-time signal $s[n]$, $n = 0, 1, \ldots, N-1$ as follows:

$$S[k] = \sum_{n=0}^{N-1} s[n]\, e^{-i2\pi kn/N} \qquad k = 0, 1, \ldots, N-1.$$

Show by direct computation that $s[n]$ can be recovered via the Inverse DFT: $s[n] = \dfrac{1}{N}\displaystyle\sum_{k=0}^{N-1} S[k]\, e^{+i2\pi kn/N}$.

Hint: You should first prove and then use the following *orthogonality* of discrete complex exponentials:

$$\sum_{k=0}^{N-1} e^{i2\pi km/N} e^{-i2\pi kn/N} = \begin{cases} N & \text{if } m = n \\ 0 & \text{otherwise.} \end{cases}$$

Problem 5.3 (Eigenvalues of the one-dimensional DFT)**.** The DFT formulas may be written as follows:

$$S[k] = \sum_{n=0}^{N-1} s[n]\; W_N^{-nk}, \quad \text{for } k = 0, \ldots, N-1, \qquad s[n] = \frac{1}{N}\sum_{k=0}^{N-1} S[k]\; W_N^{nk}, \quad \text{for } n = 0, \ldots, N-1,$$

where $W_N = e^{i2\pi/N}$ is the N^{th} root of unity. We introduce the vectors $\mathbf{S} = \begin{bmatrix} S[0] \\ \vdots \\ S[N-1] \end{bmatrix}$ and $\mathbf{s} = \begin{bmatrix} s[0] \\ \vdots \\ s[N-1] \end{bmatrix}$

in $\mathbb{C}^N$, and we can rewrite the the above linear transformations may be rewritten in vector form $\mathbf{S} = D\mathbf{s}$ and $\mathbf{s} = D^{-1}\mathbf{S}$, where D is an $(N \times N)$ invertible complex matrix, called the DFT matrix.

(a) Let $N = 4$. Write explicitly the (4×4) matrix D.

(b) Now assume N is generic. It is the case that $\mathcal{F}^4[s] = s$, for any singal s (see Problem 4.7 in HW#4: this is a general property of Fourier transforms, that holds irrespective of the group where the signals are defined). What property of the matrix D follows immediately from this fact?

(c) Still assume that N is generic. Using the previous part of this problem, what are the possible eigenvalues of the matrix D? *Note:* You will find that one of the four possible eigenvalues is $\lambda = 1$: there are signals that are the DFT of themselves!

Remark: While we are only scratching the surface of the topic here, the DFT matrix has very interesting algebraic properties that can be used for computational purposes (see, for example, the two papers: J.H. McClellan and T.W. Parks, "Eigenvectors and Eigenvalues of the Discrete Fourier Transform", *IEEE Transactions on Audio and Electroacoustics*, 20(1):66-74, 1972; Berthold K.P. Horn, "Interesting Eigenvectors of the Fourier Transform", *Transactions of the Royal Society of South Africa*, 65(2):100–106, 2010).

Summary on the Two-Dimensional Discrete Fourier Transform (2D DFT). In class we defined the Fourier Transform for 2D discrete periodic signals, i.e. defined on the group: $I = Z(T_1) \times Z(T_2)/Z(MT_1) \times Z(NT_2)$, whose dual is the group

$$\widehat{I} = Z\big(\tfrac{1}{MT_1}\big) \times Z\big(\tfrac{1}{NT_2}\big)/Z\big(\tfrac{1}{T_1}\big) \times Z\big(\tfrac{1}{T_2}\big) = Z(F_1) \times Z(F_2)/Z(MF_1) \times Z(NF_2),$$

where $F_1 = \frac{1}{MT_1}$ and $F_2 = \frac{1}{NT_2}$. The formula for the Fourier transform and its inverse are given by:

$$S(k_1F_1, k_2F_2) = T_1T_2 \sum_{n_1=0}^{M-1}\sum_{n_2=0}^{N-1} s(n_1T_1, n_2T_2)\, e^{-i2\pi\left(\frac{k_1n_1}{M}+\frac{k_2n_2}{N}\right)}, \quad (k_1F_1, k_2F_2) \in \widehat{I}, \tag{1}$$

$$s(n_1T_1, n_2T_2) = F_1F_2 \sum_{k_1=0}^{M-1}\sum_{k_2=0}^{N-1} S(k_1F_1, k_2F_2)\, e^{i2\pi\left(\frac{k_1n_1}{M}+\frac{k_2n_2}{N}\right)}, \quad (n_1T_1, n_2T_2) \in I. \tag{2}$$

Note that the impulse in the dual group $\widehat{I}$ is:

$$\delta_{\widehat{I}}(k_1F_1, k_2F_2) = \begin{cases} \frac{1}{F_1F_2} = MN & \text{for } k_1 \in Z(M) \text{ and } k_2 \in Z(N), \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Assume $T_1 = T_2 = 1$; define $s[n_1,n_2] = s(n_1T_1, n_2T_2)$, $S[k_1,k_2] = S(k_1F_1, k_2F_2)$, and $\delta[k_1,k_2] = \delta_{\widehat{I}}(k_1F_1, k_2F_2)$. The formulas (1) and (2) become:

$$S[k_1,k_2] = \sum_{n_1=0}^{M-1}\sum_{n_2=0}^{N-1} s[n_1,n_2]\, e^{-i2\pi\left(\frac{k_1n_1}{M}+\frac{k_2n_2}{N}\right)}, \quad \text{and} \quad s[n_1,n_2] = \frac{1}{MN}\sum_{k_1=0}^{M-1}\sum_{k_2=0}^{N-1} S[k_1,k_2]\, e^{i2\pi\left(\frac{k_1n_1}{M}+\frac{k_2n_2}{N}\right)},$$

which are reminiscent of those in the previous problem. Note that $k_1 = 0,1,\ldots,M-1$, $k_2 = 0,1,\ldots,N-1$, $n_1 = 0,1,\ldots,M-1$, and $n_2 = 0,1,\ldots,N-1$. Introducing the M^{th} and N^{th} roots of unity, namely $W_M = e^{i2\pi/M}$ and $W_N = e^{i2\pi/N}$, we may rewrite the above expressions as follows:

$$\boxed{S[k_1,k_2] = \sum_{n_1=0}^{M-1}\sum_{n_2=0}^{N-1} s[n_1,n_2]\, W_M^{-k_1n_1} W_N^{-k_2n_2}, \quad \text{and:} \quad s[n_1,n_2] = \frac{1}{MN}\sum_{k_1=0}^{M-1}\sum_{k_2=0}^{N-1} S[k_1,k_2]\, W_M^{k_1n_1} W_N^{k_2n_2}}.$$

In fact, the above formulas are referred to as the 2D Discrete Fourier Transform (DFT) and the 2D Inverse Discrete Fourier Transform (IDFT). ∎

Problem 5.4. Consider the 2D signal:

$$s[n_1, n_2] = \sin\left(2\pi\frac{\ell_1 n_1}{M} + 2\pi\frac{\ell_2 n_2}{N}\right), \tag{4}$$

with $n_1 = 0, 1, 2, \ldots, M-1$ and $n_2 = 0, 1, 2, \ldots, N-1$; also,

$$\ell_1 \in \{0, 1, 2, \ldots, M-1\} \quad \text{and} \quad \ell_2 \in \{0, 1, 2, \ldots, N-1\}$$

are fixed constants. Show that the DFT of the above signal $s[n_1, n_2]$ is given by:

$$S[k_1, k_2] = \frac{i}{2}\big(\delta[k_1 + \ell_1, k_2 + \ell_2] - \delta[k_1 - \ell_1, k_2 - \ell_2]\big).$$

Hint: by equation (3), the delta function in the *dual* domain is equal to MN at the origin, not 1.

Problem 5.5 (Translating the frequencies of the DFT)**.** Consider a function (an $M \times N$ image, if you will) $s[n_1, n_2]$, with $n \in \mathbb{Z}\times\mathbb{Z}/Z(M)\times Z(N)$ (so s is M-periodic with respect to n_1 and N-periodic with respect to n_2), and its Discrete Fourier Transform $S[k_1, k_2]$ defined on the same domain. Assume, for simplicity, that both M and N are even. Define the new function $g[n_1, n_2] = (-1)^{n_1+n_2} s[n_1, n_2]$: that is, we are changing the sign of the pixels where $n_1 + n_2$ is *odd*, leaving the others unchanged. Show that the DFT of g is $G[k_1, k_2] = S[k_1 - M/2, k_2 - N/2]$. *Remark:* This means that the DFT G of g is just the DFT S of s, where the origin has been *shifted* to the center of the rectangle $\mathbb{D} = \{0, 1, \ldots, M-1\} \times \{0, 1, \ldots, N-1\}$, and the center of the rectangle to the corner in position (M, N). The low and high frequencies have been switched!

Problem 5.6 (DFT of the Laplace operator)**.** At the beginning of the course we introduced the numerical Laplace operator, namely for an image s we defined $\nabla^2 s = h_L * s$, where the convolution kernel h_L is:

$$h_L = \begin{bmatrix} h_L(-1,-1) & h_L(-1,0) & h_L(-1,1) \\ h_L(0,-1) & h_L(0,0) & h_L(0,1) \\ h_L(1,-1) & h_L(1,0) & h_L(1,1) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}.$$

We can create a *periodic* Laplacian operator $\ell[n_1, n_2]$, with $(n_1, n_2) \in \mathbb{Z}\times\mathbb{Z}/Z(M)\times Z(N)$, just by considering h_L as defined on all of $\mathbb{Z}^2$ and repeating it periodically (every M integers in the n_1 direction, and every N integers in the n_2 direction). In other words, $\ell[n_1, n_2]$ is defined as follows:

$$\ell[n_1, n_2] = \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} h_L(n_1 - mM, n_2 - nN)$$

Note that ℓ is completely determied by the MN values on $\mathbb{D} = \{0, 1, \ldots, M-1\} \times \{0, 1, \ldots, N-1\}$:

$$\ell[n_1, n_2] = \begin{cases} -4 & \text{for } (n_1, n_2) = (0, 0), \\ 1 & \text{for } (n_1, n_2) = (1, 0),\ (M-1, 0),\ (0, 1),\ \text{and } (0, N-1), \\ 0 & \text{otherwise} \end{cases}$$

Note that, by periodicity, the value at $(M-1, 0)$ is determined by the one at $(-1, 0)$, and the value at $(0, N-1)$ is determined by the one at $(0, -1)$.

(a) Compute (by hand) the DFT $L[k_1, k_2]$ of $\ell[n_1, n_2]$. Since $\ell[n_1, n_2]$ is real and symmetric, its DFT is real and symmetric too: so make sure to write it in a form that does not contain complex functions. *Hint:* You should be able to obtain an expression where only cosines and constants make their appearance.

(b) For $M = N = 128$, plot $z = L[k_1, k_2]$, with $(k_1, k_2) \in \mathbb{D}$, as the graph of a function (i.e. as a surface). By interpreting L as the frequency response of a filter, how would you classify such filter? I.e. is it low-pass, high-pass, or band-pass? Justify your answer.

In the problems that follow, we shall employ the following notation:

$b = \#$ of bits per pixel (typically $b = 8$), $L = 2^b = \#$ of gray levels, $\mathbb{P} = \{0, 1, 2, \ldots, L-1\}$,
$\mathbb{D} = \{0, 1, \ldots, M-1\} \times \{0, 1, \ldots, N-1\}$, where: $M = \#$ rows, $N = \#$ columns.

***Matlab* commands for the 2-dimensional Discrete Fourier Transform.** In this assignment you are going to work with the image `image-chest-xrays.png`. It shows the x-ray of a chest with *deterministic* additive noise, namely $g = s + n$, where the noise is of the type seen in a previous homework: $n[n_1, n_2] = \sin(2\pi\ell_1 n_1/M + 2\pi\ell_2 n_2/N)$. You have already shown that its Fourier transform is a pair of delta functions in the frequency domain (note that in real applications the additive noise may not be *exactly* sinusoidal of the type (4), but its Fourier transform is still *approximately* a pair of δ functions.) You can import the image as follows:

```
I0 = imread('image-chest-xrays.png');
g = I0(:,:,1);
Rg = imref2d(size(g));
```

The second command chooses one of the three color components of the `png` image (the image is in black and white, so they are all the same); the third one allows us to display the coordinates of the image:

```
imshow(g,Rg,'InitialMagnification',300);
```

At this point, we compute its DFT: $G[k_1, k_2] = \sum_{n_1=0}^{M-1} \sum_{n_2=0}^{N-1} g[n_1, n_2]\, W_M^{-k_1 n_1} W_N^{-k_2 n_2}$, as follows:

```
G = fft2(g);
```

We have that $G[k_1, k_2]$ is a *complex* function, mostly comprised of low frequencies, with very large peaks. In order to display its *magnitude*, we first mitigate its peaks with a logarithm, then rescale it from 0 to 255.

```
AG1 = log(1+abs(G));
MaxVal = max(max(AG1));
AG2 = uint8(255*(AG1/MaxVal));
```

At this point, we can display the magnitude of the Fourier transform—more precisely, of the function $\log\bigl(1 + |G[k_1, k_2]|\bigr)$:

```
SAG2 = fftshift(AG2);
imshow(SAG2,Rg,'InitialMagnification',300);                    (*)
```

Note that this displays $\log\bigl(1+|G[k_1, k_2]|\bigr)$ in the domain $\mathbb{D} = \{0, 1, \ldots, M-1\} \times \{0, 1, \ldots, N-1\}$. As discussed in class, *the low frequescies are at the corners of the shown image*. We can shift the entire Fourier transform by $(M/2, N/2)$ (that is, we can move the origin to the center of the image, so to display the low frequencies in the middle of the image), by typing, instead of $(*)$:

```
imshow(fftshift(AG2),RI,'InitialMagnification',300);
```

At this point it is possible to perform mathematical operations directly on the Fourier transform of the image, e.g. by my multiplying it by a desired frequency response of a filter: $U[k_1, k_2] = H[k_1, k_2]G[k_1, k_2]$. We can see the effects of the operation in the spatial domain by computing the *inverse* Discrete Fourier Transform (2D IDFT) of the result:

```
u=uint8(ifft2(U));
imshow(u,'InitialMagnification',300);
```

Problem 5.7 (Low-Pass Filters: Ideal vs. Butterworth)**.** Download the 400×400 image `image-Dante.png`, call it $g[n_1, n_2]$, with $(n_1, n_2) \in \mathbb{D} = \{0, 1, \dots, M-1\} \times \{0, 1, \dots, N-1\}$, where $M = N = 400$.

(a) First compute its 2D DFT $G[k_1, k_2]$ with $k_1, k_2 \in \{0, \dots, 399\}$, and display its magnitude via the function $\log\bigl(1 + |G[k_1, k_2]|\bigr)$, in a way that the low frequencies are shown in the *middle* of the image.

(b) Consider an *ideal* low pass filter, i.e. with frequency response $L[k_1, k_2] = \begin{cases} 1 & \text{for } \sqrt{k_1^2 + k_2^2} \le D_0 \\ 0 & \text{otherwise} \end{cases}$

(since the image has $M = N = 400$, you may choose, for example, $D_0 = 50$) and apply it to $G[k_1, k_2]$. Note that if you apply the filter to the version of the Fourier transform that has been shifted by $(M/2, N/2)$, you will also have to shift the frequency response L of the filter before multiplying it by $G[k_1, k_2]$ (otherwise, you'd have to define the ideal low-pass filter at the 'four corners' of $\mathbb{D}$): in fact, the easiest way to proceed in practice is to (1) apply fftshift to $G[k_1, k_2]$; (2) multiply by $L[k_1 - M/2, k_2 - N/2]$, and (3) apply `fftshift` to the resulting matrix to actually get the product $L[k_1 - M/2, k_2 - N/2]G[k_1 - M/2, k_2 - N/2]$. Finally, compute the IDFT of $G[k_1, k_2]L[k_1, k_2]$ and display it: this will be a smooth (blurry) image, but note the 'wavy' artifacts (due to the Gibbs phenomenon) near the boundaries.

(c) The latter can be avoided if, instead of an ideal low-pass filter, one uses a *Butterworth* low-pass filter of order n, namely:

$$B_n[k_1, k_2] = \frac{1}{1 + \left(\frac{\sqrt{k_1^2 + k_2^2}}{D_0}\right)^{2n}} \tag{5}$$

Choose the same value for D_0 as you did in the previous part, and a low value of n (e.g $n = 4$; note that for large n we have $B_n[k_1, k_2] \simeq L[k_1, k_2]$, and our efforts would be in vain). Compute the IDFT $m[n_1, n_2] = \mathcal{F}^{-1}\bigl[G \cdot B_n\bigr][n_1, n_2]$, with $(n_1, n_2) \in \mathbb{D}$ and verify that this yields a smooth Dante image *without* the Gibbs-type artifacts.

(d) Now that you have a smooth version of Dante, you can actually *sharpen* the original picture by adding to it its sharp component, thus boosting its high frequencies (this is the principle behind Unsharp Masking): compute $g_2 = g + 3(g - m)$, where $g - m$ is the sharp component of g, and display it. Compare sharpness of g and g_2 using the Total Variation (TV) norm (see Problem 2.4 in HW#2).

Problem 5.8 (Notch filter)**.** Download the image `image-chest-xrays.png`: it is the x-ray of a chest corrupted by additive deterministic sinusoidal noise, that makes it almost impossible to visualize the ribe, spine, heart, and other organs. Use *Matlab* to compute its 2D DFT $G[k_1, k_2]$ and visualize its magnitude, as explained above. You will note that there are two "peaks", one at the north-west and the other at the south-east of the origin—roughly identify their coordinates just by looking a the image $\log\bigl(1 + |G[k_1, k_2]|\bigr)$. These correspond to the additive sinusoidal noise (remember, from Problem **5.4**, that the DFT of a sinusoidal function is the difference of two purely imaginary delta functions). Eliminate such peaks with a *notch filter*, i.e. compute $U[k_1, k_2] = H[k_1, k_2]G[k_1, k_2]$, where the frequency response $H[k_1, k_2]$ of the filter is equal to zero in a neighborhood of each of those peaks, and equal to one everywhere else. Compute the IDFT of $U[k_1, k_2]$ and display the resulting image.