

**HOMEWORK #6**  
**Due at 5 pm on Friday 11/21/14**

*Homework policy:* Homeworks can be turned in during class prior to the due date or to the TA's mailbox in the Graduate Student Lounge. Late homeworks will be marked down by 20% per day. If you know that you need to turn in a homework late because of an emergency or academic travel, please let the TA know ahead of time. Collaboration is encouraged on homework assignments, however, the homework that you submit should reflect your own understanding of the material. It is recommended that you make a copy of the homework for yourself (e.g. scan it in) before you turn it in.

**Required Readings:** Review sections 5.1-5.5 and 5.6.2. Read Section 5.7

**Problem 1**

Consider the object  $m(x, y) = \text{rect}(x + 2, y + 2) - \text{rect}(x - 2, y - 2)$ , which is similar (but not identical) to the object you looked at in Homework 3.

- (a) Determine the values of  $\Delta k_x$  and  $\Delta k_y$  needed to avoid aliasing.
- (b) Assume that the desired resolution in both the x and y directions is  $\frac{1}{4}$  cm. Determine  $k_{x,\max}$  and  $k_{y,\max}$  and also the number of voxels in the x and y directions. Assuming a spin-warp acquisition with one phase-encode line per repetition, what is the total scan-time for TR = 100 ms? (note: TR is the time between RF pulses).
- (c) Assume that the readout time (i.e. time to cover one line in k-space) is much smaller than TR = 100ms. Develop and describe a simple solution that takes advantage of the geometry of the object to reduce total scan time by roughly a factor of 2, while keeping the TR at 100 ms.

**Problem 2**

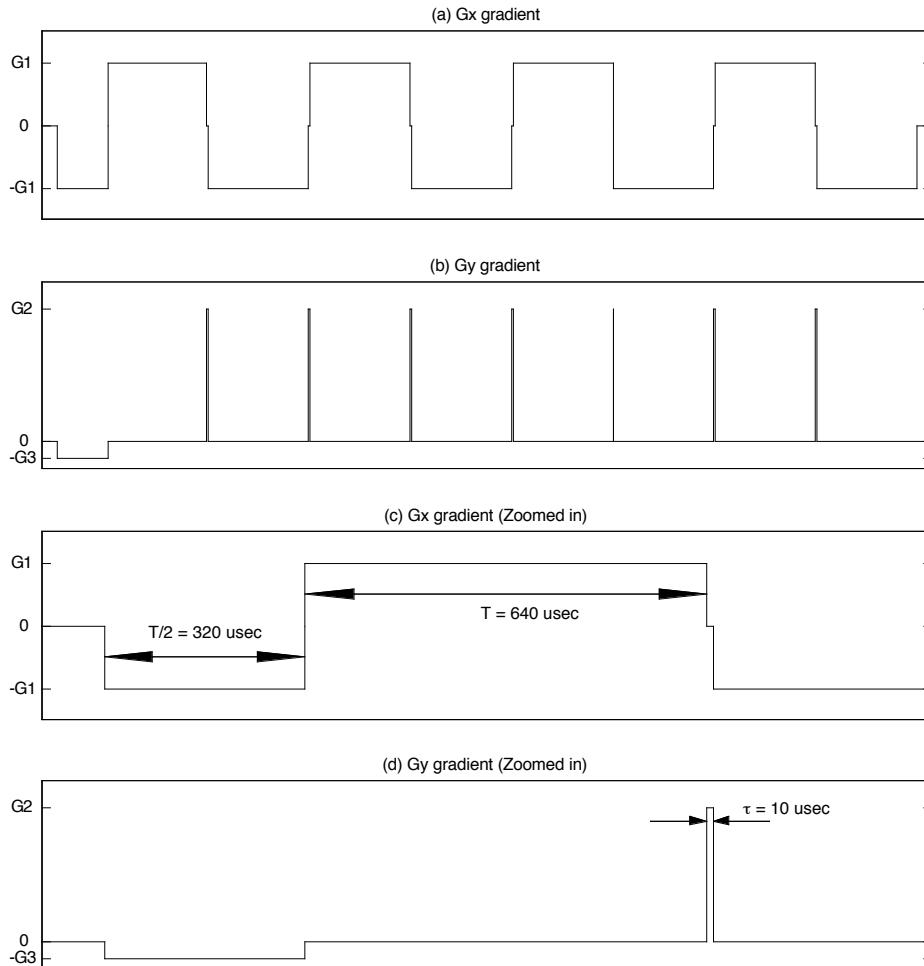
Consider the 2D object:  $f(x, y) = \cos(2\pi x - 2\pi\sqrt{3}y)\text{rect}(x/4)\text{rect}(\sqrt{3}y/4)$

- a) Sketch the object, labeling key features
- b) Compute and sketch the Fourier Transform of the object, labeling key features.
- c) Determine the values of  $\Delta k_x$  and  $\Delta k_y$  needed to avoid aliasing.
- d) Assume that the desired resolution in the x and y directions is given by  $\frac{1}{2}$  cm and  $\frac{1}{2\sqrt{3}}$  cm. Determine  $k_{x,\max}$  and  $k_{y,\max}$  and also the number of voxels in the x and y directions.
- e) Draw and label gradient waveforms that will provide sufficient coverage of k-space, according to the specifications in parts c and d. Assume a spin-warp trajectory, a maximum available gradient of 6 G/cm, and a sampling rate of 10 micro-seconds. Where possible, minimize the duration of the gradients (i.e. minimize the time of the initial readout dephaser necessary to get to the starting point in k-space).

### Problem 3

Consider the gradient waveforms shown in the figure. The full waveforms are shown in panels (a) and (b), and zoomed-in views are shown in (c) and (d). The analog-to-digital converter (ADC) is turned on during the flat parts of the readout ( $G_x$ ) gradients with a sampling rate of  $\Delta t$ .

- (a) Determine the sequence parameters ( $G_1$ ,  $G_2$  and  $G_3$ , and  $\Delta t$ ) to achieve the following image specifications:  $\text{FOV}_x = 256\text{mm}$ ;  $\text{FOV}_y = 192\text{ mm}$ ,  $\delta_x = 4\text{ mm}$  and  $\delta_y = 24\text{ mm}$ .
- (b) Draw the k-space trajectory (make sure to label units correctly).



## Problem 4 – MATLAB exercise

In this exercise we will examine the Fourier transforms of some simple test objects.

**Part 1.** Create a script with the following code

```
%Define a test object
span = -16:15;
nvox = length(span);
[x,y] = meshgrid(span,span);
obj = zeros(nvox,nvox);
obj([1:8 17:24],:) = 1;

%Define k-space coverage
dk = 1/32;kmax = 0.5;
kspan = -kmax:dk:(kmax-dk);
Nk = length(kspan);
[kx,ky] = meshgrid(kspan,kspan);

%Brute-force computation of the Fourier Transform
% This is not the best way to compute it, but is helpful for showing the
% process
j = sqrt(-1);

for ix = 1:Nk
    for iy = 1:Nk;
        g = exp(-j*2*pi*(kx(ix,iy)*x + ky(ix,iy)*y));
        f = sum(sum(g.*obj)); % Fourier Transform
        fmat(ix,iy) = f; %Store the Fourier transform values in fmat
    end;
end
```

**Part 2.** Use imagesc to plot images of the object and its Fourier transform.

**Part 3.** Find the 3 highest absolute values in the Fourier transform.

- a) Where in k-space do these occur? (i.e. provide the values of kx and ky).
- b) Draw the quiver diagrams showing orientation of phasors corresponding to each of these points in k-space.
- c) Create a second set of quiver diagrams that takes into account the knowledge of the object (i.e. where the object is equal to zero, the product of the object and the phasor is zero). Based on the orientation of the phasors in the quiver diagrams and the knowledge of the object, provide an explanation of why the 3 highest values are observed at these points in k-space.
- d) Compute and plot the vector sum of the phasors in the quiver diagrams from part c for each of the 3 points in k-space. Verify that the vector sum is equal to the value of the Fourier transform at the corresponding points in k-space.
- e) Using the code loop above as a starting point, write MATLAB code to do a brute force computation of the inverse 2D Fourier transform and verify that the transform generates the original object. (HINTS: You may need to take the real part of your answer to account for numerical precision effects. You will also want to divide the sum by the total number of points (which is 1024 for this object)).

**Part 4.**

- a) Now design an object where the highest absolute value occurs at the center of k-space ( $k_x = 0$ ,  $k_y = 0$ ) and the next 4 highest absolute values occur at the following points in k-space  $(-1/16, -1/16)$ ,  $(-1/16, 1/16)$ ,  $(1/16, -1/16)$ , and  $(1/16, 1/16)$ .
- b) For each of the points in k-space: (1) plot the quiver diagrams showing orientation of phasors; (2) plot a second set of quiver diagrams that also take into account knowledge of the object; (3) compute and plot

the vector sum of the phasors from the second set of quiver diagrams and verify that it is equal to the Fourier transform at the corresponding k-space location.

- c) Use your code to compute the inverse transform of the object and verify that you obtain the original object.